

Pengembangan Sistem Pelaporan Pemeliharaan Stasiun ONLIMO (Online Monitoring) Milik BPPT (Badan Pengkajian dan Penerapan Teknologi) Berbasis Android

Ferlie Deanada Effendi¹, Herman Tolle², Komang Candra Brata³

Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹ferlieffendi@gmail.com, ²emang@ub.ac.id, ³k.candra.brata@ub.ac.id

Abstrak

Sebagai bentuk pertanggung jawaban dari segala kegiatan yang dilakukan oleh operator stasiun dibutuhkan sebuah pelaporan kepada *admin data center*. Salah satu kegiatan yang wajib dilaporkan adalah kegiatan perawatan stasiun ONLIMO. Namun, sistem pelaporan yang masih diterapkan masih menggunakan *group chatting room*, hal tersebut menyulitkan *admin data center* untuk melakukan pendataan setiap laporan yang ada. Selain itu, dirasakan pula kesulitan dalam hal pencatatan penggantian *sparepart* sensor. Sehingga, diperlukan sebuah sistem yang mampu mempermudah proses pelaporan di Badan Pengkajian dan Penerapan Teknologi (BPPT). Sistem diimplementasikan pada *mobile application* dikarenakan banyaknya operator stasiun yang menggunakan perangkat *android* dan pada sisi server menggunakan *platform website*. Pada pengembangannya digunakan metode *agile* karena sesuai dengan proses pengevaluasian penelitian di BPPT. Hasil pengujian fungsional didapatkan bahwa keseluruhan fitur telah valid dan sesuai dengan kebutuhan. Sedangkan untuk pengujian non-fungsional, didapatkan nilai *usability* sebesar 75 yang menyatakan bahwa sistem sudah baik namun diperlukan perbaikan lagi, sistem bergantung dengan koneksi internet untuk aspek *reliability*, dan sistem dapat menyeleksi pengguna dengan baik untuk aspek *security*. Untuk tingkat efektifitas didapatkan 100% dimana seluruh pengguna dapat menggunakan sistem secara efektif, dan didapatkan nilai 0,0103 *goals/sec* untuk efisiensi sistem yang menandakan sistem pelaporan ONLIMO sudah efektif dan efisien.

Kata kunci: *pelaporan, agile, usability, mobile application, android*

Abstract

For the responsibility of all activities who's been done by station operators, a report to the admin data center is needed. One activities that must be reported is ONLIMO station maintenance activities. However, the current reporting system still using group chat rooms, which makes the admin difficult to get all the data. In addition, there were also difficulties in recording sensor sparepart replacement. So, a system is needed to simplify reporting process at Badan Pengkajian and Penerapan Teknologi (BPPT). The system is implemented on mobile application because a lot of station operators using Android device and on server side using the website platform. For development, agile method is used because it suitable with the process of evaluating research at BPPT. The functional test results found that all feature was valid. While for non-functional testing, we found usability value is 75 which shows that the system is good but needed repairs, the system needs internet connection for reliability aspects, and the system can selecting users well for security aspects. For the effectiveness level obtained 100% where all users can use the system effectively, and for system efficiency is 0.0103 goals/sec which indicates the ONLIMO reporting system already effective and efficient.

Keywords: *reporting, agile, usability, mobile application, android*

1. PENDAHULUAN

Pelaporan merupakan salah satu bentuk bentuk pertanggung jawaban atas hal yang telah dikerjakan (Siagina, 2003). Kegiatan pelaporan

ini diterapkan dalam kegiatan pemeliharaan stasiun pada ONLIMO (*Online Monitoring*) yang dikembangkan oleh Badan Pengkajian Dan Penerapan Teknologi (BPPT) yang bekerja sama dengan Kementerian Lingkungan Hidup dan

Kehutanan (KLHK). ONLIMO merupakan proyek untuk memonitor kualitas air yang ada di Indonesia dengan menggunakan sensor yang dicelupkan ke dalam air. Sensor dan data logger yang berada pada stasiun menghasilkan nilai kualitas air yang dikirimkan ke *server*. Stasiun dijaga oleh operator stasiun yang berkewajiban merawat dan melaporkan setiap kegiatan kepada pihak pusat.

Adapun kegiatan utama yang dilakukan operator stasiun yakni membersihkan sensor kemudian melaporkannya kepada pusat. Data pelaporan yang didapatkan inilah yang ditunjukkan di *website* ONLIMO dan dapat memberikan informasi kepada pihak yang melihat data tersebut. Menurut Bapak Heru sebagai koordinator utama dalam proyek ONLIMO, pelaporan dilakukan dengan menggunakan *group chatting room*. Kendala yang dihadapi adalah pesan yang berisi laporan tertumpuk dengan pesan lain didalam *group chatting room*. Tertumpuknya pesan menyebabkan informasi menjadi tidak akurat.

Kesulitan lainnya adalah mendata pelaporan yang telah dilakukan oleh para operator stasiun dan pencatatan penggantian *sparepart* sensor. Kemudian dengan adanya penambahan titik peletakan sensor, membuat kesulitan yang dialami beliau akan lebih terasa. Sehingga kegiatan pelaporan dapat menjadi kurang efisien dan efektif dalam pelaksanaannya.

Dari permasalahan tersebut, maka diperlukan adanya sebuah sistem yang dapat menampung serta mempermudah pelaporan yang dilakukan oleh para operator stasiun. Mengacu pada proses pelaporan yang dilakukan operator stasiun yakni mengirimkan gambar atau video, maka perangkat *mobile* berbasis Android tepat digunakan untuk menjadi solusi dari kendala yang dihadapi. Lalu untuk bagian sisi server atau administrator yang berada di *platform website*, dapat dilakukan pembuatan rekap pelaporan resmi dari hasil pelaporan serta penggantian *sparepart* yang telah dilakukan.

Berdasarkan uraian di atas, dilakukan pengembangan sistem pelaporan untuk pemeliharaan stasiun ONLIMO yang dikembangkan oleh BPPT. Dalam pengembangannya digunakan metode *Agile* yang cocok dengan cara evaluasi penelitian di BPPT, yaitu evaluasi dilakukan setelah produk selesai dan dalam evaluasi tersebut akan terdapat perubahan kebutuhan dan dengan *Agile*, klien mudah diajak berdiskusi. Diskusi digunakan

untuk mendapatkan *feedback* sehingga sistem diharapkan benar-benar sesuai dengan kebutuhan klien. Dengan adanya diskusi ini, para pengembang dapat meminimalkan dokumentasi. (Sommerville, 2011).

2. METODE PENELITIAN



Gambar 1. Diagram alir metode penelitian

Dalam pengembangan sistem ini, diterapkan metode *Agile*. Dimana metode *Agile* merupakan salah satu metode pengembangan perangkat lunak yang menawarkan proses pengembangan perangkat lunak yang lebih cepat dan memiliki keringanan di sisi bisnis (Abrahamsson, et al., 2002). Metode *Agile* juga merupakan salah satu metode pengembangan software yang didasarkan pada pengembangan berulang dan bertahap (Kumar, G. & Bhatia, P.K., 2012). Sehingga dapat disimpulkan bahwa metode *Agile* adalah metode pengembangan perangkat lunak yang memiliki proses pengembangan yang berulang dan bertahap, sehingga pengembangan yang dilakukan lebih cepat. Penerapan metode *Agile* pada pengembangan sistem ini dapat dilihat dalam Gambar 1.

Studi pustaka merupakan langkah awal dalam metode pengembangan sistem. Pengembangan aplikasi meliputi 5 tahapan pengembangan yang terus berulang bila *feedback* dari klien belum memenuhi kriteria yaitu semua kebutuhan dari klien telah terpenuhi dan dilakukan pengujian *usability* sebagai data pendukung wawancara. Dimulai dari analisis kebutuhan, kemudian lanjut ke tahap perancangan sistem, implementasi dan terakhir dilakukan pengujian. Apabila pengujian selesai, maka akan dilakukan wawancara kebutuhan

serta sistem akan dicoba oleh klien untuk mendapatkan *feedback* apakah sistem telah sesuai atau belum. Klien juga akan diberikan kuisisioner SUS (*Software Usability Scale*) untuk mengetahui tingkat usability system dan mendukung pernyataan dalam wawancara.

Apabila klien masih menginginkan adanya perubahan pada sistem, maka akan dilakukan kembali tahapan analisis kebutuhan hingga pengujian berdasarkan perubahan yang diinginkan dan akan dicoba kembali oleh klien hingga tidak ada lagi perubahan yang diinginkan. Kemudian pada setiap iterasi dilakukan pengujian usability untuk mendukung data wawancara kebutuhan.

Pada pengujian *usability*, jumlah pengguna yang diuji sebanyak 5 orang karena jumlah tersebut adalah hasil terbaik ketika menguji *usability* (Nielsen, J., 2000). Kelima orang tersebut terdiri dari 3 orang operator stasiun, 1 orang kalibrator, dan 1 orang *admin data center*.

Selain pengujian *usability*, dilakukan juga pengujian *security* (pengujian terhadap autentikasi pengguna sistem) dan *reliability* (menguji apakah sistem dapat berjalan dengan baik dengan internet).

Pada pengujian *usability* dilakukan dua tipe pengujian yaitu dengan kuisisioner dan dengan penghitungan metrik. Pengujian metrik digunakan untuk menghitung dua aspek *usability* yaitu efektifitas dan efisiensi. Pada pengujian aspek efektifitas akan digunakan metrik *completion-rate* yang mengukur tingkat keberhasilan pengguna menyelesaikan tugas yang diberikan (Mifsud, J., 2015). Persamaan metrik *completion-rate* dapat dilihat pada persamaan (1).

$$\text{efektifitas} = \frac{\text{jumlah tugas yang selesai}}{\text{total tugas yang diberikan}} \times 100\% \quad (1)$$

Kemudian pada aspek efisiensi akan digunakan metrik *time-based efficiency* untuk mengukur seberapa cepat pengguna menggunakan sistem (Designer, UI, 2018). Metrik ini meminta masukan berupa tugas yang berhasil dilakukan dan waktu yang digunakan. Untuk menentukan keberhasilan tugas, dilakukan penghitungan waktu penyelesaian tugas terhadap lima responden acak yang berumur antara 20 hingga 50an tahun. Waktu yang telah dihitung, akan dirata rata per tugasnya. Rata-rata inilah yang dibuat sebagai acuan kesuksesan pengguna dalam

menyelesaikan tugas. Kemudian, pengguna sesungguhnya akan diuji, dan hasilnya akan dihitung dengan metrik *time-based efficiency* pada persamaan (2).

$$\text{time - based efficiency} = \frac{\sum_{j=1}^R \sum_{i=1}^N \frac{n_{ij}}{t_{ij}}}{NR} \quad (2)$$

N = total tugas yang diberikan

R = jumlah pengguna

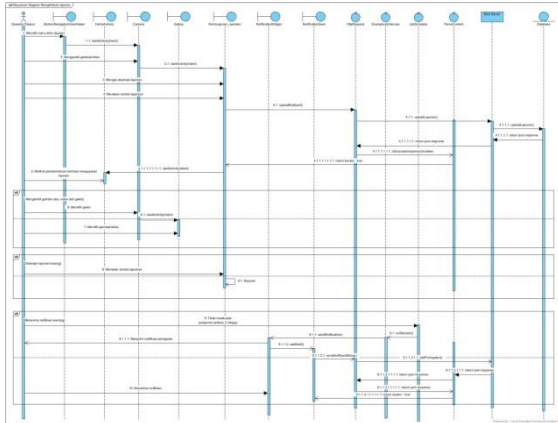
n_{ij} = hasil penyelesaian tugas i yang dilakukan pengguna j
 t_{ij} = waktu yang diperlukan pengguna j untuk menyelesaikan tugas i .

Lalu pengujian *usability* dengan kuisisioner dilakukan setelah pengguna menggunakan sistem. Kuisisioner yang digunakan adalah kuisisioner SUS (*Software Usability Scale*). SUS (*Software Usability Scale*) merupakan kuisisioner berisi 10 pernyataan terhadap penilaian *usability* yang menggunakan skala likert sebagai acuan penilaiannya (Thomas, N., 2015). SUS dihitung berdasarkan pernyataan positif dan negatif. Pada pernyataan positif yakni nomor 1,3,5,7 dan 9, nilai skala likert akan dikurangi dengan 1. Sedangkan pernyataan negatif yaitu nomor 2, 4, 6, 8 dan 10, angka 5 dikurangi dengan nilai skala likert. Total dari penghitungan tersebut akan dikalikan dengan 2,5.

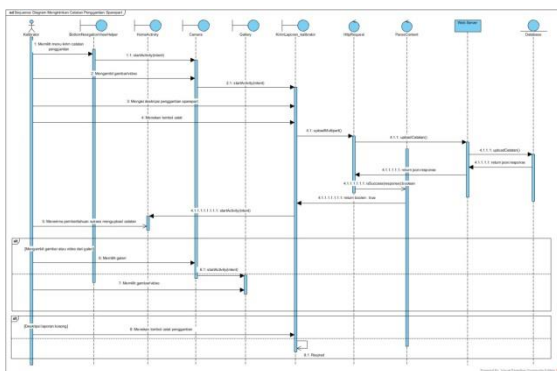
Setelah pengujian *usability*, terdapat pengujian *security* dan *reliability*. Pengujian *security* memastikan bahwa akses data yang sah berada di orang yang benar. Sehingga tidak ada penyalahgunaan data karena hak akses yang tidak sah (Help ,Software Testing., 2018). Sedangkan pengujian *reliability* adalah pengujian yang ditujukan untuk mengukur kinerja keseluruhan sistem dalam beberapa periode waktu (Performance, Testing., 2018). Dimana pada penelitian ini dilakukan pengujian kinerja sistem ketika menggunakan internet dan tidak menggunakan internet.

3. HASIL

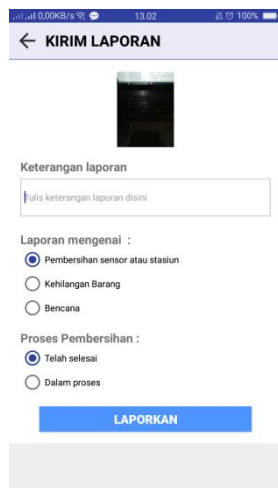
Dalam pengembangan sistem pelaporan ONLIMO ini terdapat dua fitur utama, yaitu fitur untuk mengirim laporan bagi operator stasiun dan fitur untuk mengirim catatan penggantian *sparepart* bagi kalibrator. Perancangan fitur ini terdapat dalam Gambar 2 dan Gambar 3. Kedua diagram ini dijadikan acuan untuk membuat sistem. Hasil implementasi dari rancangan tersebut terdapat dalam Gambar 4 dan Gambar 5. Dimana pada gambar tersebut memperlihatkan halaman kirim laporan.



Gambar 2. Diagram sequence mengirim laporan



Gambar 3. Diagram sequence mengirim catatan penggantian sparepart

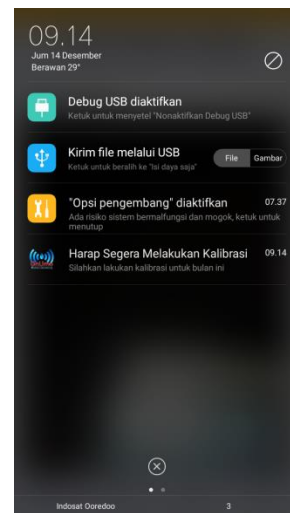


Gambar 4. Hasil implementasi mengirim laporan



Gambar 5. Hasil implementasi mengirim catatan penggantian sparepart

Setelah sistem selesai, dilakukan wawancara dengan klien untuk menggali kebutuhan yang ingin ditambahkan serta melakukan pengujian *usability*. Pada penelitian ini, terdapat penambahan kebutuhan satu kali yaitu untuk mengirim notifikasi otomatis kepada kalibrator. Hasil implementasi kebutuhan tambahan tersebut dapat dilihat dalam Gambar 6. Sedangkan untuk hasil pengujian *usability*, didapatkan nilai 80. Nilai 80 pada pengujian *usability* yang menggunakan SUS merupakan nilai yang sangat baik (Sauro, J., 2011).



Gambar 6. Hasil implementasi menerima notifikasi kalibrasi bagi kalibrator

Setelah kebutuhan tambahan selesai, dilakukan kembali wawancara dan pengujian *usability*. Dari hasil wawancara tidak ada lagi tambahan kebutuhan yang diinginkan, sehingga iterasi berhenti. Kemudian hasil pengujian *usability* yang didapatkan mendapatkan nilai 85, dimana nilai tersebut cukup tinggi untuk hasil pengujian *usability* dengan SUS.

Setelah iterasi selesai, dilakukan beberapa pengujian akhir. Pengujian yang dilakukan adalah pengujian fungsional dan non fungsional. Pengujian fungsional dilakukan dengan metode *black-box*. Hasil dari pengujian ini adalah seluruh fitur dalam sistem telah tervalidasi. Hal ini dikarenakan antara hasil yang diharapkan dan hasil yang didapatkan telah sesuai tanpa adanya kesalahan. Fitur atau kebutuhan yang telah dirancang juga telah sesuai dengan yang ada pada sistem. Kemudian pada iterasi kedua ditambahkan satu kebutuhan bagi kalibrator yaitu mengirim notifikasi bagi kalibrator. Fitur tambahan ini juga telah berhasil divalidasi. Sehingga dapat dikatakan bahwa semua fitur atau kebutuhan sistem telah tervalidasi dan terpenuhi.

Pada pengujian non fungsional, terdapat tiga pengujian yang dilakukan yaitu pengujian *usability*, pengujian *reliability*, dan pengujian *security*. Pengujian *usability* dilakukan dengan menggunakan kuisioner SUS dan menghitung metrik *usability* untuk aspek efektifitas dan efisiensi. Untuk mendapatkan hasil dari aspek efektifitas dan efisiensi dilakukan pendataan dengan memperhatikan kesuksesan pengguna dan waktu yang digunakan dalam melakukan tugas yang diberikan. Hasil yang didapatkan dari aspek efektifitas sebesar 100%, hal ini menunjukkan bahwa seluruh tugas yang diberikan berhasil diselesaikan oleh pengguna. Sedangkan hasil dari aspek efektifitas, didapatkan nilai 0,0103 *goals/sec*. Hal ini menunjukkan bahwa setiap pengguna dapat menyelesaikan tugas sebanyak 0,0103 setiap detiknya dengan total waktu yang diperlukan oleh seluruh pengguna sebanyak 466,37 detik.

Hasil yang didapatkan dari pengujian *usability* dengan kuisioner SUS dapat dilihat pada Tabel 1. Dari hasil yang telah dijabarkan, dihitung rata rata nilai *usability* dan didapatkan nilai 75. Nilai ini termasuk bagus, akan tetapi masih butuh perbaikan (Sauro, J., 2011).

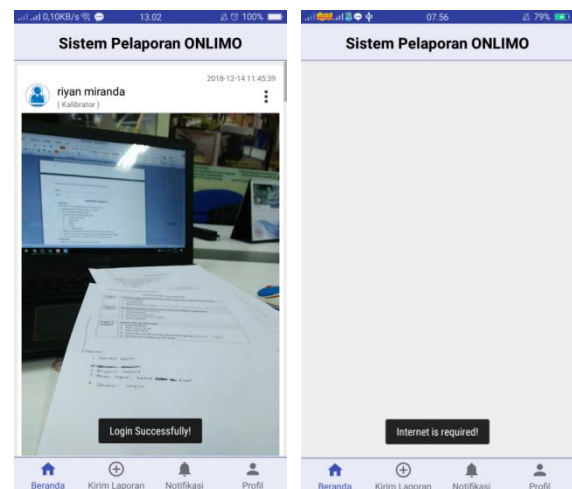
Tabel 1. Hasil pengujian *usability*

End-user	Nilai	Total nilai	Hasil (Total x 2,5)
Operator Stasiun Masjid Istiqlal	4+3+4+1+4+2+3+3+4+1	29	72,5
Operator Stasiun Pintu Air Manggarai	3+3+3+1+3+3+3+3+1	26	65

Operator Stasiun Bendungan Pasar Baru	4+4+4+3+4+3+4+3+4+1	34	85
Kalibrator	3+3+4+3+4+3+4+3+4+0	31	77,5
Admin Data Center	3+4+4+1+3+3+4+4+3+1	30	75

Kemudian untuk hasil dari pengujian *reliability* didapatkan dari menguji ketergantungan sistem dengan koneksi internet. Hasil pengujian membuktikan bahwa sistem harus berjalan dengan menggunakan koneksi internet untuk dapat mengakses data yang dibutuhkan. Dalam Gambar 7 terdapat perbedaan kinerja sistem ketika terkoneksi dan tidak terkoneksi.

Pengujian terakhir adalah pengujian *security*. Teknik yang digunakan dalam pengujian ini adalah perbedaan hak akses. Penggunaan teknik ini dilakukan menguji hak akses dengan melakukan autentifikasi dan otorisasi setiap tipe pengguna (Help ,Software Testing., 2018). Apakah akses yang dimiliki benar atau tidak. Hasil pengujian pun menunjukkan bahwa sistem dapat membedakan hak akses setiap pengguna. Sehingga bisa dikatakan sistem dapat menjaga keamanan hak akses dari berbagai pengguna.



Gambar 7. Diagram sequence mengirim catatan penggantian *sparepart*

4. KESIMPULAN

Berdasarkan penelitian yang dilakukan, didapatkan beberapa kesimpulan. Pertama adalah dari hasil proses analisis kebutuhan yang terdiri dari spesifikasi sistem, kegunaan sistem, jenis pengguna dan juga kebutuhan fungsional

serta non fungsional. Penggambaran persyaratan dilakukan dengan menggunakan *use case diagram* dan dijelaskan secara detail dengan *use case scenario*. Hasil yang telah didapatkan kemudian dijadikan acuan untuk membuat rancangan sistem. Rancangan sistem digambarkan dengan menggunakan beberapa diagram. Salah satu diagram yang digunakan adalah diagram *sequence* yang menunjukkan jalannya sistem dari segi kode program. Keseluruhan rancangan ini dibuat berdasarkan kebutuhan yang telah didefinisikan sebelumnya. Rancangan yang telah dibuat dijadikan dasar untuk implementasi sistem. Diakhir pengembangan, klien menginginkan adanya penambahan kebutuhan yaitu kebutuhan notifikasi untuk kalibrator. Selain dengan penambahan, dilakukan pula *usability testing* menggunakan SUS kuisisioner yang berfungsi sebagai data pendukung dari hasil wawancara penambahan kebutuhan. Nilai SUS yang didapatkan pada iterasi pertama sebesar 80 dan pada iterasi kedua didapatkan nilai 85. Hal ini mendukung data wawancara yang telah merasa cukup puas dengan sistem yang telah dibuat.

Setelah sistem dibuat, dilakukan beberapa pengujian sebelum sistem digunakan. Terdapat dua pengujian yang dilakukan, yaitu pengujian fungsional dan pengujian non fungsional. Hasil dari pengujian fungsional dapat disimpulkan bahwa seluruh fitur telah valid dan sesuai dengan kebutuhan. Kemudian untuk pengujian non fungsional, terdapat tiga pengujian yaitu pengujian *usability*, pengujian *reliability*, dan pengujian *security*. Dari hasil pengujian *usability* didapatkan total nilai 75, dimana nilai ini sudah baik tapi masih diperlukan perbaikan. Sedangkan untuk pengujian *reliability* didapatkan hasil bahwa sistem bergantung dengan koneksi internet. Pengujian terakhir adalah *security*, hasil pengujian menunjukkan bahwa sistem dapat menyeleksi pengguna dengan baik.

Kemudian untuk tingkat efektifitas dan efisiensi yang dimiliki sistem didapatkan hasil untuk efektifitas adalah 100% dimana seluruh pengguna dapat menggunakan sistem secara efektif. Sedangkan untuk efisiensi sistem, didapati nilai 0,0103 *goals/sec*. Sehingga dapat disimpulkan sistem pelaporan ONLIMO ini sudah efektif dan efisien.

Saran bagi sistem ini diantaranya perbaikan *user experience* dari sistem. Hal ini didasari dengan nilai *usability* yang hanya mencapai nilai 75. Selain itu dibutuhkan adanya sistem *website*

pada sisi klien untuk lebih mempermudah akses sistem bagi pengguna. Kemudian diperlukan adanya fitur tambahan untuk tempat berdiskusi permasalahan yang ditemui oleh operator stasiun ataupun kalibrator di stasiun pemantauan kepada *admin data center*.

5. DAFTAR PUSTAKA

- Abrahamsson, P., Salo, O., Ronkainen, J. & Warsta, J. 2002. *Agile Software Development Methods: Review and Analysis*, VTT publication 478, Espoo, 107p.
- Designer, UI. 2018. *Efficiency*. [online] Tersedia di <<http://ui-designer.net/usability/efficiency.htm>> [Diakses 16 Juli 2018]
- Help ,Software Testing., 2018. *Security Testing (A Complete Guide)*. [online] Tersedia di <<https://www.softwaretestinghelp.com/how-to-test-application-security-web-and-desktop-application-security-testing-techniques/>> [Diakses 16 Juli 2018]
- Kumar, G. dan Bhatia, P.K., 2012. *Impact of Agile Methodology on Software Development Process*. International Journal of Computer Technology and Electronics Engineering (IJCTEE), 2(4).
- Lewis, James R. dan Sauro, J. 2009. *The Factor Structure of the System Usability Scale*. [e-journal]. Researchgate. Tersedia di <https://www.researchgate.net/publication/221099434_The_Factor_Structure_of_the_System_Usability_Scale> [Diakses 01 Desember 2018]
- Mifsud, J. 2015. *Usability Metrics – A Guide To Quantify The Usability Of Any System*. [online] Tersedia di <<https://usabilitygeek.com/usability-metrics-a-guide-to-quantify-system-usability/>> [Diakses 15 Juli 2018]
- Nielsen, J. 2012. *Usability 101: Introduction to Usability*. [online] Tersedia di <<https://www.nngroup.com/articles/usability-101-introduction-to-usability/>> [Diakses 10 Juli 2018]
- Performance, Testing. 2018. *What is Reliability Testing?*. [online] Tersedia di <<http://www.testingperformance.org/definitions/what-is-reliability-testing>>

[Diakses 16 Juli 2018]

- Sauro, J., 2011. *Measuring Usability With The System Usability Scale (SUS)*. [online] Tersedia di <
<https://measuringu.com/sus/>> [Diakses 17 Desember 2018]
- Siagina, S. 2003. *Filsafat Administrasi Edisi Revisi*. Jakarta : PT Bumi Aksara.
- Sommerville, Ian. 2011. *Software Engineering. 9th Edition*. [e-book] United State of America : Pearson Education ,Inc.