

## **ANALISIS KERANGKA KERJA PENGUJIAN PERANGKAT LUNAK PADA PT. JAVAN CIPTA SOLUSI**

**ALYA JELITA NURRAHMANIA<sup>1</sup>, ANDHIK BUDI CAHYONO<sup>2</sup>**

<sup>1,2</sup>Program Studi Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia

Jalan Kaliurang Km. 14,5, Sleman, Daerah Istimewa Yogyakarta 55584

Email: [18523153@students.uii.ac.id](mailto:18523153@students.uii.ac.id)<sup>1</sup>, [andhikbudi@uii.ac.id](mailto:andhikbudi@uii.ac.id)<sup>2</sup>

### **ABSTRAK**

Kerangka kerja pengujian digunakan sebagai salah satu cara mendefinisikan tahapan-tahapan dalam melakukan pengujian. Kerangka kerja pengujian yang digunakan oleh PT. Javan Cipta Solusi merupakan salah satu strategi yang digunakan untuk melakukan pengujian dan melalui beberapa tahapan sebelum akhirnya perangkat lunak digunakan oleh pengguna. Implementasi kerangka kerja tersebut digunakan pada salah satu proyek pengembangan fitur perangkat lunak. Makalah ini membahas tentang analisis implementasi kerangka kerja pengujian pada PT. Javan Cipta Solusi terhadap proyek pengembangan perangkat lunak. Hasil dalam analisis kerangka kerja pengujian ini mendefinisikan tahapan pengujian yang digunakan oleh PT. Javan Cipta Solusi dalam menguji perangkat lunak untuk salah satu proyek pengembangannya.

**Kata Kunci:** Pengujian, Kerangka kerja, Perangkat Lunak

### **I. PENDAHULUAN**

Perangkat lunak dalam pengembangannya di dalam sebuah proyek melewati beberapa proses sebelum akhirnya perangkat lunak digunakan oleh pengguna. Tahapan pengujian merupakan salah satu tahapan yang berperan penting dalam pengembangan perangkat lunak. Pengujian perangkat lunak merupakan proses validasi dan verifikasi bahwa produk yang sedang dikembangkan memenuhi kebutuhan yang mendasari perancangan dan pengembangan perangkat lunak (Nidhra, 2012). Pengujian juga dilakukan untuk mencari kesalahan yang terjadi pada sebuah perangkat lunak pengujian perangkat lunak juga bertujuan untuk mengevaluasi kualitas dari perangkat lunak tersebut (Jin dan Xue 2011), Untuk mendapatkan kesalahan pada proses pengujian diperlukannya skenario pengujian yang bertujuan agar semua kemungkinan dalam proses pengujian fitur di dalam perangkat lunak terdefinisi semua. Pengujian perangkat lunak pada umumnya dapat dilakukan oleh beberapa cara. Pengujian perangkat lunak dapat berupa pengujian

secara manual atau secara otomatis. Pengujian secara manual biasanya menguji langsung perangkat lunak tersebut tanpa menggunakan bantuan dari tools pengujian yang lain. Sebaliknya pengujian secara otomatis menggunakan tools lain untuk menguji perangkat lunak, pengujian otomatis digunakan ketika perangkat lunak memiliki banyak kasus pengujian untuk diuji.

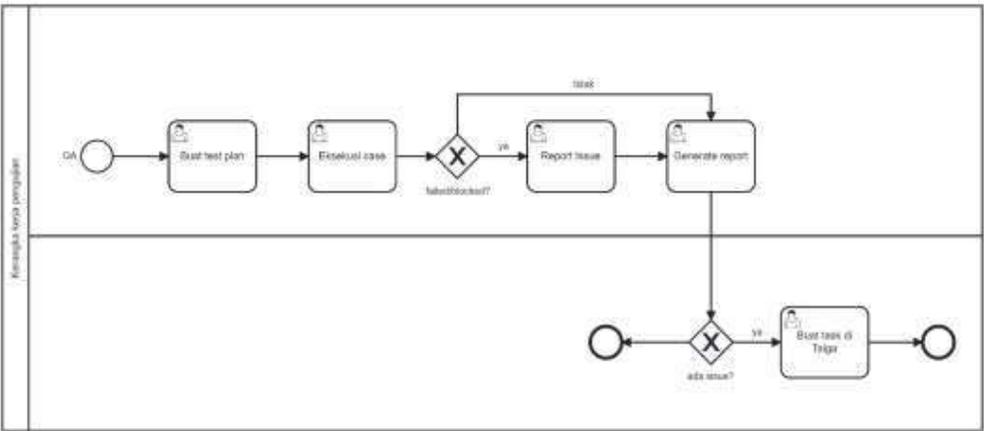
PT. Javan Cipta Solusi sebagai sebuah *software developer* yang melayani permintaan pengembangan perangkat lunak, dalam proses pengujian perangkat lunak memiliki sebuah kerangka kerja pengujian yang digunakan oleh *Quality Assurance* sebagai acuan dalam pengujian perangkat lunak. Sebagian besar pengujian yang dilakukan oleh PT. Javan Cipta Solusi menggunakan pengujian secara manual. Pengujian dilakukan dengan menguji langsung pada perangkat lunak tersebut. Kerangka kerja pengujian oleh PT. Javan Cipta Solusi merupakan kerangka kerja yang digunakan secara internal oleh tim proyek di PT. Javan Cipta Solusi. Proses pengujian ini dilakukan oleh PT. Javan Cipta Solusi dimulai dari merancang test plan atau pendefinisian kasus pengujian dari perangkat lunak tersebut.

Tujuan dari pembahasan kerangka kerja pengujian oleh PT. Javan Cipta Solusi untuk menganalisis proses implementasi dari kerangka kerja pengujian yang dimiliki oleh PT. Javan Cipta Solusi. Analisis dari kerangka kerja pengujian ini juga bertujuan untuk mendefinisikan bagaimana pengujian perangkat lunak dilakukan oleh PT. Javan Cipta Solusi, sebelum akhirnya perangkat lunak akan diserahkan kepada klien untuk digunakan oleh pengguna perangkat lunak. Kerangka kerja pengujian ini diimplementasikan terhadap sebuah proyek pengembangan fitur perangkat lunak yang dimiliki oleh sebuah Lembaga Pemerintah.

## II.METODE PENELITIAN

Kerangka kerja pengujian perangkat lunak yang dilakukan oleh PT. Javan Cipta Solusi menggunakan kerangka kerja yang menekankan pengujian terhadap siklus pengembangan perangkat lunak, pengujian dilakukan secara berulang untuk memastikan perangkat lunak tidak terdapat perubahan dan hasil yang diharapkan sesuai. Pengujian yang digunakan oleh PT. Javan Cipta Solusi untuk memudahkan

pengujian perangkat lunak dan memudahkan klien untuk mengidentifikasi issue yang terdapat pada perangkat lunak. Sebagai tahap awal, seorang *Quality Assurance* atau *Tester* dalam menguji perangkat lunak dimulai dari pemahaman terhadap perangkat lunak dari proyek yang akan dikerjakan. *Tester* akan diberikan panduan terkait perangkat lunak untuk memudahkan menganalisis kebutuhan pengujian. Setelah memahami alur dari perangkat lunak, seorang *Quality Assurance* akan langsung masuk ke dalam proyek untuk melakukan pengujian. Kerangka kerja pengujian oleh PT. Javan Cipta Solusi dapat dilihat Gambar 1. Kerangka Kerja Pengujian PT. Javan Cipta Solusi.



Gambar 1. Kerangka Kerja Pengujian PT. Javan Cipta Solusi.

Aktivitas pengujian perangkat lunak dimulai dari membuat *test plan* yang ditulis ke dalam sebuah dokumen yaitu *User Acceptance Testing (UAT)*. Dokumen *User Acceptance Testing* berisikan semua kasus pengujian fitur yang akan diuji. Kasus pengujian akan dibagi ke dalam beberapa modul fitur yang sedang dikembangkan yang sebelumnya telah dilakukan pemahaman terhadap alur fitur perangkat lunak. Kasus pengujian tersebut akan dituliskan ke dalam sebuah tabel untuk memudahkan seorang *Quality Assurance* melakukan pengujian. *User Acceptance Testing* berisikan skenario pengujian dari fitur-fitur yang sedang dikembangkan dan memuat hal-hal lain terkait proyek sedang dilakukan. Contoh tabel dari dokumen *User Acceptance Testing* yang berisikan test case dapat dilihat pada Gambar 2. *Template Dokumen User Acceptance Testing*.

User Acceptance Document					
Project Name:					
Project Code:					
Client Name					
Total		Passed		Retest	
Untested		Failed		Blocked	
Sub Module	Test Case Description	Test Steps	Expected Results	Actual Results	Status

Gambar 2. Template Dokumen User Acceptance Testing

**Keterangan:**

*Sub-Module* : Bagian dari modul fitur utama yang akan diuji.

*Test Case Description* : Deskripsi dari kasus pengujian perangkat lunak yang akan diuji.

*Test Steps* : Langkah-langkah dalam melakukan pengujian.

*Expected Results* : Hasil yang diharapkan dari pengujian.

*Actual Results* : Hasil yang sebenarnya diperoleh dari pengujian.

*Status* : Hasil dari pengujian yang diperoleh, status diisi oleh beberapa nilai seperti “*Passed*” apabila pengujian dari *expected results* dan *actual results* sesuai. “*Failed*” apabila pengujian tidak sesuai dengan *expected results* dan *actual results*.

*Notes* : Catatan yang digunakan oleh klien ketika terjadi perubahan kecil.

Dalam dokumen *User Acceptance Testing* akan tercatat berapa kasus pengujian yang akan diuji oleh *tester* dan status kasus pengujian tersebut, hal ini memudahkan *Quality Assurance* untuk melihat sejauh mana perkembangan dari pengujian perangkat lunak yang sedang dikembangkan. Setelah test plan selesai dideskripsikan ke dalam dokumen *User Acceptance Testing*, tahapan selanjutnya adalah mengeksekusi kasus pengujian yang sebelumnya telah ditulis ke dalam dokumen *User Acceptance Testing*. Seorang *Quality Assurance* bertanggung jawab atas semua kasus pengujian yang telah didefinisikan. Semua kasus pengujian diuji terlebih dahulu oleh *Quality Assurance*.

Pengujian dilakukan secara keseluruhan kasus pengujian dan memastikan semua fitur yang didefinisikan sesuai dengan apa yang diharapkan. Kasus pengujian yang tidak berhasil selanjutnya akan diberikan catatan bahwa kasus pengujian tersebut tidak lolos pengujian. *Tester* akan melaporkan kasus-kasus pengujian mana

saja yang tidak berhasil dalam pengujian yang selanjutnya issue tersebut akan ditindaklanjuti oleh tim proyek dan diperbaiki oleh pengembang perangkat lunak.

## **I. HASIL DAN PEMBAHASAN**

Kerangka pengujian ini dilakukan terhadap pengembangan perangkat lunak fitur Jabatan Fungsional yang dimiliki oleh salah satu Lembaga Pemerintah. Pengembangan fitur Jabatan Fungsional bertujuan untuk membantu pegawai negeri sipil untuk mengajukan Jabatan Fungsional. Salah satu modul yang dikembangkan pada proyek ini adalah modul yang menampilkan sebaran pegawai negeri sipil yang melakukan Jabatan Fungsional Pelaksanaan Pengadaan Barang dan Jasa (JF PPBJ) dari seluruh Lembaga Pemerintah di Indonesia. Pada modul Sebaran JF PPBJ kasus pengujian mendefinisikan bagaimana Sebaran JF PPBJ tersebut dapat diakses umum. Modul Sebaran JF PPBJ didefinisikan ke dalam test case yang dapat dilihat pada Tabel 1. Skenario Pengujian.

Tabel 1. Skenario Pengujian

<i>Sub-Module</i>	<i>Test Case Description</i>	<i>Test Steps</i>	<i>Expected Results</i>	<i>Actual Results</i>	<i>Status</i>	<i>Notes</i>
<i>Dashboard JF PPBJ – Data LKPP</i>	Menampilkan data Jumlah PPBJ (Aktif) berdasarkan data LKPP	<ol style="list-style-type: none"> <li>1. Buka <i>website</i> PPSDM</li> <li>2. Arahkan kursor pada Menu Statistik dan Data</li> <li>3. Pilih Sebaran JF PPBJ</li> <li>4. Klik pilihan Dashboard JF PPBJ</li> <li>5. Pilih Data LKPP</li> </ol>	Dapat menampilkan data Jumlah JF PPBJ (orang) data berdasarkan data LKPP	Terdapat data pada kolom Jumlah JF PPBJ Aktif (orang) berdasarkan data LKPP	<i>Passed</i>	
<i>Dashboard JF PPBJ – Data SAPK BKN</i>	Menampilkan data Jumlah PPBJ (Aktif) berdasarkan data SAPK BKN	<ol style="list-style-type: none"> <li>1. Buka <i>website</i> PPSDM</li> <li>2. Arahkan kursor pada Menu Statistik dan Data</li> <li>3. Pilih Sebaran JF PPBJ</li> <li>4. Klik pilihan Dashboard JF PPBJ</li> <li>5. Pilih Data SAPK BKN</li> </ol>	Dapat menampilkan data Jumlah JF PPBJ (orang) data berdasarkan data SAPK BKN	Terdapat data pada kolom Jumlah JF PPBJ Aktif (orang) berdasarkan data SAPK BKN	<i>Passed</i>	
<i>Dashboard JF PPBJ – Data SAPK BKN</i>	Menampilkan data Instansi JF PPBJ berdasarkan data SAPK BKN	<ol style="list-style-type: none"> <li>1. Buka <i>website</i> PPSDM</li> <li>2. Arahkan kursor pada Menu Statistik dan Data</li> <li>3. Pilih Sebaran JF PPBJ</li> <li>4. Klik pilihan Dashboard JF PPBJ</li> <li>5. Pilih Data SAPK BKN</li> </ol>	Dapat menampilkan data Rekomendasi Kebutuhan JF PPBJ (orang) menurut data SAPK BKN	Terdapat data pada kolom Instansi JF PPBJ menurut data SAPK BKN	<i>Passed</i>	

<i>Dashboard</i> PPBJ – Data SAPK BKN	Menampilkan data Rekomendasi kebutuhan JF PPBJ (orang) berdasarkan data SAPK BKN	1. Buka <i>website</i> PPSDM 2. Arahkan kursor pada Menu Statistik dan Data 3. Pilih Sebaran JF PPBJ 4. Klik pilihan Dashboard PPBJ 5. Pilih Data SAPK BKN	Dapat menampilkan data Rekomendasi Kebutuhan JF PPBJ (orang) menurut data SAPK BKN	Terdapat data pada kolom Telah mendapatkan Rekomendasi Kebutuhan(Instansi) JF PPBJ menurut data SAPK BKN	<i>Passed</i>
<i>Dashboard</i> PPBJ – Data LKPP	Menampilkan data Instansi JF PPBJ berdasarkan data LKPP	1. Buka <i>website</i> PPSDM 2. Arahkan Kursor pada Menu Statistik dan Data 3. Pilih Sebaran JF PPBJ 4. Klik pilihan Dashboard PPBJ 5. Pilih Data LKPP	Dapat menampilkan data Rekomendasi Kebutuhan JF PPBJ (orang) menurut data LKPP	Terdapat data pada kolom Instansi JF PPBJ menurut data LKPP	<i>Passed</i>
<i>Dashboard</i> PPBJ – Data LKPP	Menampilkan data Rekomendasi kebutuhan JF PPBJ (orang) berdasarkan data SAPK BKN	1. Buka <i>website</i> PPSDM 2. Arahkan Kursor pada Menu Statistik dan Data 3. Pilih Sebaran JF PPBJ 4. Klik pilihan Dashboard PPBJ 5. Pilih Data LKPP	Dapat menampilkan data Rekomendasi Kebutuhan JF PPBJ (orang) menurut data LKPP	Terdapat data pada kolom Telah mendapatkan Rekomendasi Kebutuhan(Instansi) JF PPBJ menurut data LKPP	<i>Passed</i>

Sebaran JF PPBJ dapat diakses oleh umum di situs web Lembaga Pemerintah tersebut. Tampilan dari sebaran JF PPBJ dapat dilihat pada Gambar 3. Tampilan Sebaran JF PPBJ.



Gambar 3. Tampilan Sebaran JF PPBJ.

Tampilan sebaran JF PPBJ terdiri dari beberapa bagian visualisasi data seperti tabel, *pie chart* dan *bar chart* yang memudahkan pengguna dalam melihat data jabatan fungsional. Dalam pengembangannya, Sebaran JF PPBJ terdapat beberapa issue yang dilaporkan, Issue tersebut meliputi ketidaksesuaian data yang asli dengan data yang ditampilkan atau perubahan terkait tampilan perangkat lunak. Tester akan melaporkan fitur yang tidak sesuai tersebut ke dalam sebuah *task* pada *management tools* yang digunakan oleh PT. Javan Cipta Solusi yaitu Taiga. Sebuah *task issue* yang dilaporkan harus memenuhi beberapa ketentuan, seperti kondisi saat ini dari *issue* dan bagaimana *acceptance criteria* dari issue tersebut.

Setelah memastikan *issue* tersebut sudah diperbaiki maka tester akan menambahkan komentar pada *task* tersebut bahwa *issue* tersebut sudah tidak terdapat lagi pada perangkat lunak. Tahapan terakhir sebelum pengujian perangkat lunak dapat digunakan oleh pengguna adalah memastikan semua kasus pengujian tidak terdapat lagi *issue* atau *bug*. Kasus pengujian yang terdapat *issue* akan diuji kembali. Selanjutnya dokumen UAT akan diserahkan kepada manajer proyek

sebagai laporan bahwa perangkat lunak tersebut dapat diserahkan kepada klien sebelum nantinya akan digunakan oleh pengguna perangkat lunak tersebut.

#### IV. KESIMPULAN

Berdasarkan pengujian yang telah dilakukan, kerangka kerja pengujian perangkat lunak oleh PT. Javan Cipta Solusi dalam praktiknya untuk menguji kelayakan perangkat lunak sebelum diserahkan kepada klien dinilai cukup untuk memenuhi kebutuhan kelayakan perangkat lunak. Pada proyek pengembangan fitur Jabatan Fungsional menggunakan kerangka kerja ini juga dinilai cukup baik untuk mendefinisikan kebutuhan dari klien. Hasil dari kerangka kerja tersebut mampu untuk memenuhi kebutuhan dari pengguna perangkat lunak.

Analisis dan pembuatan skenario pengujian dengan bantuan Dokumen *User Acceptance Testing* memudahkan dalam mendokumentasikan dan melakukan pengujian. Selain itu hasil dari pengujian perangkat lunak tersebut dapat digunakan oleh pengguna atau karyawan dari Lembaga Kementerian tersebut untuk mengajukan Jabatan Fungsional.

#### DAFTAR PUSTAKA

- Nakagawa, E.Y., and J.S. Maldonado, 2011, Contributions and perspectives in architectures of software testing environments, in IEEE Proceeding of 25th Brazilian Symposium on Software Engineering, pp. 66-71. DOI 10.1109/SBES.2011.42
- Nidhra, S. (2012). Black Box and White Box Testing Techniques - A Literature Review. *International Journal of Embedded Systems and Applications*, 2(2), 29–50. <https://doi.org/10.5121/ijesa.2012.2204>
- J. Jin and F. Xue, "Rethinking Software Testing Based on Software Architecture," *2011 Seventh International Conference on Semantics, Knowledge and Grids*, 2011, pp. 148-151, doi: 10.1109/SKG.2011.32.
- Crispin, L., & Gregory, J. (2009). *Agile testing: A practical guide for testers and agile teams*. Pearson Education
- Reza, H., and S. Lande, 2010, Model based testing using software architecture, in IEEE Proceeding of 7th International Conference on Information Technology, pp. 188-192. DOI 10.1109/ITNG.2010