

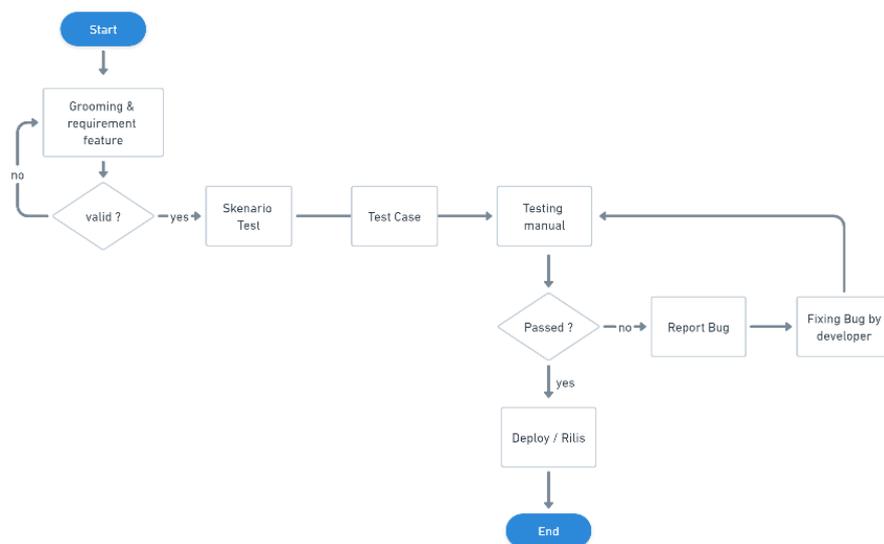
BAB III ANALISIS DAN PERANCANGAN

3.1 Analisis

3.1.1 Identifikasi Masalah

Pengujian perangkat lunak penting untuk menunjukkan kualitas perangkat lunak dan berpengaruh pada kepuasan pelanggan. Pada *Departement General Affair*, aplikasi *GA System 2 United Tractors* untuk membantu dalam *request* kebutuhan *user* ke *general affair*, dan *project management* untuk proyek konstruksi.

Saat ini pengujian perangkat lunak yang sedang berjalan pada Unit aplikasi *GA System 2 United Tractors* dilaksanakan secara manual pengujian. Pengujian tersebut dilakukan oleh seorang *QA (Quality Assurance)*. Proses pengujian dilakukandengan menyusun scenario pengujian berdasarkan persyaratan fitur yang telah disepakati. Dalam proses pengujian ini, banyak langkah yang dilakukan *QA*, termasuk menguji alur fitur, fungsionalitas fitur, dan integrasi data secara berulang. Ketika menemukan sebuah *bug* dari fitur seorang *QA* membuat laporan reporting dalam bentuk *excel* dan melaporkan kepada *engineer developer*. Berikut gambaran langkah langkah pengujian *manual test* dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Langkah-langkah pengujian manual test

Hal tersebut sangat membutuhkan ketelitian, waktu panjang dan lama untuk melakukan proses pengujian perangkat lunak secara manual. Pengujian tersebut hanya bisa membantu untuk pengujian perangkat lunak secara berkala bukan untuk pengujian berulang. Pengujian berulang apabila dilakukan secara manual akan memakan banyak waktu dan kurang efisien, serta tidak menjamin kecacatan perangkat lunak akan berkurang. Kecacatan perangkat lunak dapat segera di atasi dengan adanya bantuan *reporting bug*.

Dengan demikian, adapun permasalahan yang dapat diidentifikasi terhadap kualitas perangkat lunak pada aplikasi *GA System 2 United Tractors* adalah sebagai berikut:

1. *QA* sering kali mengalami ketidak efisien waktu dalam melakukan pengujian secara manual secara berulang.
2. *Reporting* yang dilakukan *QA* seringkali tidak dipahami oleh *Engineer developer*.
3. Kecacatan / *bug* pada *project* masih sering ditemukan ketika akan *rilis user*.
4. *Project* hanya dilakukan secara manual *testing* sehingga perlu adanya implementasi *automation testing*.
5. Kurangnya kolaborasi dan komunikasi antara seluruh *stakeholders* – tim bisnis maupun teknikal yang terlibat dalam pengembangan.

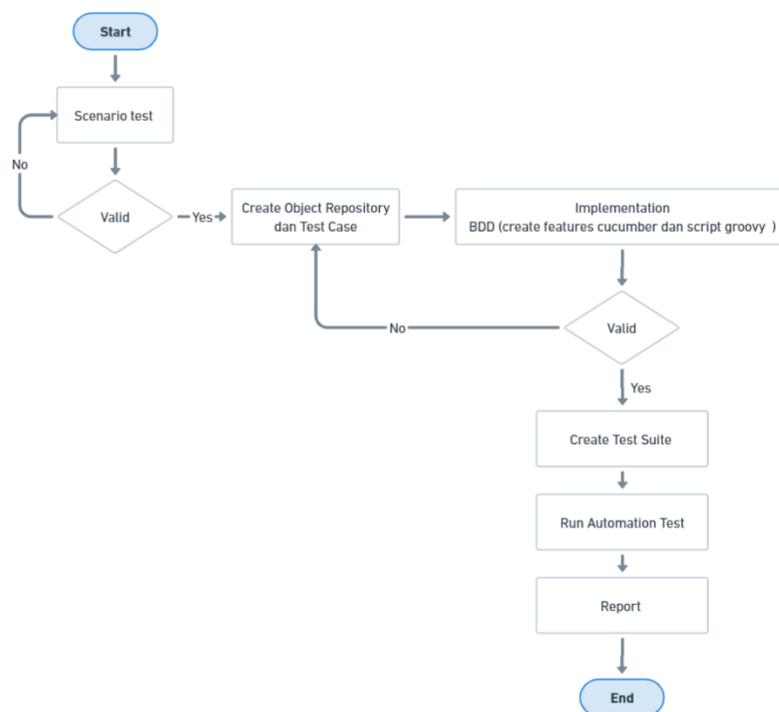
3.1.2 Pemecahan Masalah

Berdasarkan permasalahan yang sudah disebutkan di atas, maka dibuat implementasi *automation testing* dengan metode *behavior driven development (bdd)* dengan *tools katalon studio* dari sistem pengujian aplikasi *GA System 2 United Tractors*.

Dalam proses pengembangan sebuah sistem Aplikasi *GA System 2 United Tractors*, berdasarkan tingkat urgensi dalam mengatasi beberapa identifikasi permasalahan diatas, metode *behavior driven development* dapat menjadi solusi yang tepat karena BDD bertujuan untuk peningkatan komunikasi dan kolaborasi

antara para *stakeholder* - baik tim bisnis maupun teknis - yang terlibat dalam pengembangan sistem. Pendekatan BDD ini fokus pada kebutuhan pengguna dalam skenario yang telah ditentukan oleh semua pihak terkait. BDD juga menghasilkan *test report* yang mudah dipahami, sesuai dengan skenario yang tercantum dalam *feature files*. Dengan adanya BDD, tim dapat dengan cepat beradaptasi dengan perubahan dan memungkinkan insinyur untuk merespons perubahan dengan cepat, sehingga mengurangi pemborosan waktu dan pengulangan yang berlebihan.

Dengan demikian pada tahap ini, penulis juga akan mengidentifikasi cara kerja pemecahan masalah atau langkah langkah pengujian secara *automation test* dengan menggunakan metode *behavior driven development*. Berikut gambaran langkah langkah pengujian *automation test* dapat di lihat pada gambar 3.2.



Gambar 3. 2 Langkah-langkah pengujian *automation test* teknik *BDD*

Detail penjelasan langkah – langkah pemecahan masalah dengan *automation test* menggunakan metode *bdd (behavior driven development)* sebagai berikut :

1. *Scenario test*

Pada tahap ini, penulis akan mendesain *scenario test* dari fitur aplikasi yang telah penulis pilih. *Test case* berupa *positif test case* dan *negative test case*.

2. *Create object repository dan test case*

Dalam sebuah fitur terdapat sebuah *component*, dalam implementasi *automation test*, *component* tersebut akan di ubah menjadi *object repository*, kemudian akan di rancang kembali berupa *test case*.

3. Implementasi BDD (*Create feature cucumber dan script groovy*)

Pada tahap ini metode *BDD (Behavior Driven Development)* dibuat, penulis membuat *feature cucumber* pada fitur dan membuat *script groovy* berdasarkan *scenario test* dan *test case* yang sebelumnya sudah dibuat.

4. *Create test suite*

Pada tahap *create test suite*, proses *running automation* terbuat. Apabila implementasi BDD sudah sesuai, pada *test suite* ini digunakan untuk menjalankan *automation test*.

5. *Run automation test*

Pada tahap ini, *automation test* dapat dijalankan sesuai dengan *test suite* yang telah dibuat.

6. *Report*

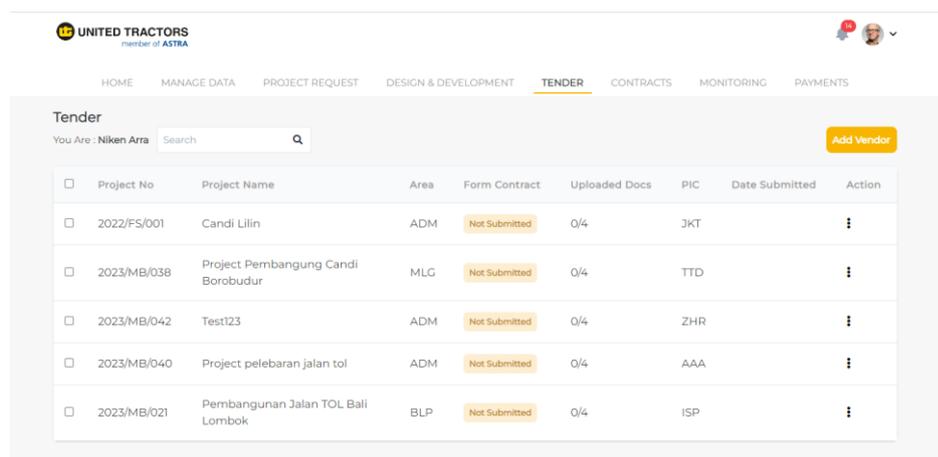
Pada tahap *report*, semua *test case* yang telah dibuat dan telah dijalankan akan menampilkan hasil testing. Baik berupa *failed* atau *passed test*.

3.2 Rencana Skenario Pengujian

Pada tahap rencana skenario pengujian dapat dijabarkan tentang gambaran dari sistem yang akan dilakukan pengujian. Skenario pengujian akan menjadi pondasi utama dalam melakukan pengujian *automation test*. Sehingga proses pengujian akan dapat lebih efektif dan efisien.

3.2.1 Rencana fitur yang akan di uji

Fitur yang akan di uji pada *automation test* dengan metode bdd (behavior driven development) pada Aplikasi *GA System 2 United Tractors* yaitu fitur tender.



Gambar 3.3 UI Menu Tender

Fitur *tender* merupakan salah satu fitur pada aplikasi *GA System 2 United Tractors* yang digunakan untuk menambah *vendor tender* dan mengatur kebutuhan *vendor tender*. Fitur tender memungkinkan pengguna untuk mengeloladan mengkoordinasikan proses pengajuan dan pelaksanaan tender. Untuk pengujian secara fungsionalitas dari fitur *tender*, pengguna dapat melakukan aktivitas seperti tambah *vendor*, *edit vendor*, *approve vendor*, *upload document vendor*, *add request contract* dan *cancel vendor*. Fitur *tender* dapat dikatakan berfungsi apabila pengguna dapat menggunakan fitur ini dengan baik. Selain itu pada fitur *tender* ini terdapat *history form contract* untuk melihat masing-masing *progress contract vendor tender*.

3.2.2 Skenario Pengujian

Pada rancangan pengujian menggunakan *automation testing*. Skenario pengujian terdiri dari beberapa *test case* yang akan diimplementasikan pada

automation testing. Adapun *test case* yang akan diujikan meliputi *test case positif* maupun *test case negative*.

Test Case Fitur Tender

Fitur *tender* memiliki 2 *test case*, yaitu 1 *test case positif* (*tender* berhasil) dan 1 *test case negative* (*tender* gagal).

No	Skenario	Data test	Step to test	Expected Result
1	Menambahkan data <i>vendor</i>	<ul style="list-style-type: none"> • <i>Name</i> : PT Tirta Jaya Vendor • <i>PIC</i> : Niken • <i>Email</i> : niken@gmail.com • <i>Vendor Adress</i> : Nganjuk • <i>Phone</i> : 085156922642 • Nama <i>Vendor</i> yang TTD Kontrak : Tirta A • <i>Jabatan Vendor</i> yang TTD Kontrak : CEO 	<ol style="list-style-type: none"> 1. Masuk ke <i>menu tender</i> 2. <i>Klik button add vendor</i> 3. Isi data <i>vendor</i> 4. <i>Klik button simpan</i> 	Sistem menerima penyimpanan data <i>vendor</i> kemudian menampilkan pada halaman <i>tender</i>
2	Membatalkan tambah <i>vendor</i>	-	<ol style="list-style-type: none"> 1. Masuk ke <i>menu tender</i> 2. <i>Klik button add vendor</i> 	Sistem tidak menyimpan data <i>vendor</i> dan kembali ke

			3. Isi data <i>vendor</i> 4. <i>Klik button cancel</i>	halaman <i>tender</i>
3	<i>Details project tender</i>	<ul style="list-style-type: none"> • <i>Name</i> : PT Tirta Jaya Vendor • <i>PIC</i> : Niken • <i>Email</i> : niken@gmail.com • <i>Vendor Adress</i> : Nganjuk • <i>Phone</i> : 085156922642 • <i>Nama Vendor</i> yang TTD Kontrak : Tirta A • <i>Jabatan Vendor</i> yang TTD Kontrak : CEO 	<ol style="list-style-type: none"> 1. Masuk ke <i>menu tender</i> 2. Pilih salah satu <i>project tender</i> 3. <i>Klik button option</i> 4. Pilih <i>details</i> 5. <i>Download</i> salah satu document 	Sistem akan menampilkan <i>details project</i> dan <i>file</i> akan terunduh

Tabel 3.1 *Test Case* Tender Berhasil

No	Skenario	Data test	Step to test	Expected Result
4	Menambahkan data <i>vendor</i> dengan data tidak di isi	-	<ol style="list-style-type: none"> 1. Masuk ke <i>menu tender</i> 2. <i>Klik button add vendor</i> 	Sistem akan menolak dan menampilkan pesan “Harap

			<p>3. Kosongkan Isi data <i>vendor</i></p> <p>4. <i>Klik button</i> <i>simpan</i></p>	di isi”
5	Gagal Membatalkan project <i>vendor</i>	-	<p>1. Masuk ke <i>menu tender</i></p> <p>2. Pilih salah satu <i>project tender</i></p> <p>3. <i>Klik button option</i></p> <p>4. Pilih <i>cancel project</i></p>	Sistem akan membatalkan project <i>vendor</i>
6	Menambahkan data <i>vendor</i> dengan data yang telah terdaftar	<ul style="list-style-type: none"> • <i>Name</i> : PT Tirta Jaya Vendor • <i>PIC</i> : Niken • <i>Email</i> : niken@gmail.com • <i>Vendor Adress</i> : Nganjuk • <i>Phone</i> : 085156922642 • Nama <i>Vendor</i> yang TTD Kontrak : Tirta A Jabatan <i>Vendor</i> yang TTD 	<p>1. Masuk ke <i>menu tender</i></p> <p>2. <i>Klik button add vendor</i></p> <p>3. Isi data <i>vendor</i> dengan data terdaftar</p> <p>4. <i>Klik button</i> <i>simpan</i></p>	Sistem akan menolak dan menampilkan pesan “Data sudah terdaftar”

		Kontrak : CEO		
--	--	---------------	--	--

Tabel 3.2 *Test Case* Tender Gagal

3.2.3 Kriteria Perbandingan Pengujian *Automation* dan Manual

Pemilihan kriteria untuk membandingkan antara pengujian *automation* dan manual test akan di implementasikan pada *Aplikasi GA System 2 United Tractor*. Dengan terfokus pada indikator permasalahan – permasalahan pada aplikasi tersebut. Adapun pemilihan kriteria perbandingan diantara lain :

1. Cakupan pengujian (*coverage*)

Pada *Aplikasi GA System 2 United Tractor*, *Automation testing* dapat mencakup fitur dan aspek perancangan secara konsisten dan berulang – ulang. Sedangkan manual *testing* mencakup berbagai fitur dan aspek perancangan secara fleksibilitas dan dapat terpengaruh oleh faktor bantuan manusia.

2. *Efisiensi* dan kecepatan

Pada *Aplikasi GA System 2 United Tractor*, *Automation testing* akan di jalankan secara berulang – ulang dengan cepat dan konsisten, hasil dari kecepatan dapat mengeksekusi tes dalam hitungan detik atau menit. Sedangkan pada manual testing, akan dijalankan dan dinilai oleh QA secara manual. Pengujian setiap fitur dan aspek secara manual membatasi jumlah pengujian dalam periode waktu tertentu.

3. Kualitas pelaporan

Pada *Aplikasi GA System 2 United Tractor*, *Automation testing* hasil pengujian akan dihasilkan secara otomatis oleh *Katalon Studio*. Laporan berisi informasi tentang tes mana yang berhasil dan gagal dengan detail waktu eksekusi dan nilai yang diharapkan sebenarnya. Selain itu, laporan membantu tim pengembangan memahami secara cepat dan efisien pada setiap aspek perancangan dan mendapatkan pandangan lebih jelas mengenai pengujian dan potensi terjadinya *bug*. Sedangkan pada manual *testing*, laporan hasil pengujian dibuat secara manual oleh QA setelah menyelesaikan setiap pengujian dan catatan tertulis secara manual dalam bentuk file *excel*.