

SCENARIO TEST

KIRI BANG !

Versi 1.0

Prototype 1

9 April 2016

Disusun oleh :

Anneke

Hendry Gustin

Junior

Mahesa

DAFTAR ISI

1. Pengantar	1
1.1 Tujuan	1
1.2 Gambaran Proses Pengujian	1
2. Skenario dan Hasil Tes	1
2.1 Unit Testing	2
2.1.1 Fungsi getFromLocation	2
2.1.2 Fungsi getToLocation	4
2.1.3 Fungsi countDistance	7
2.1.4 Fungsi countFare	10
2.2.5 Fungsi generateNavigation	14
2.2 Integration Testing	17
2.2.1 Urutan Pengujian	17
2.2.2 Metode Pengujian	18
2.2.3 Jendela Splash	19
2.2.4 Jendela Main Menu	19
2.2.5 Jendela Daftar Angkot	22
2.2.6 Jendela Navigasi	24
2.2.7 Jendela Info Angkot	26
2.2.8 Jendela Help	27
2.2.9 Jendela About	28
2.3 Function Testing	30
2.3.1 Melihat daftar dan informasi angkot	30
2.3.2 Melihat trayek/rute setiap angkot	32
2.3.3 Memasukkan tempat asal	33
2.3.4 Memasukkan tempat tujuan	34
2.3.5 Mendapatkan prakiraan tarif, jarak, dan navigasi dari tempat asal ke tujuan	36
2.3.6 Melihat petunjuk penggunaan aplikasi (Help)	37
2.3.7 Melihat <i>About</i>	38

PENGANTAR

Dokumen ini meliputi semua obyektif, fokus, dan juga strategi yang digunakan dalam pengujian kami. Selain itu, struktur dan peralatan yang digunakan untuk pengujian White Box dan pengujian tingkat fungsional juga dibahas. Hasil dan perencanaan yang terlibat dalam pengujian kami dikerjakan secara manual.

1.1 TUJUAN

Tujuan utama untuk pengujian kami adalah untuk mengenal proses setiap kasus pengujian dan menghasilkan data uji. Sementara tujuan lain dari pengujian ini tentunya untuk mencari dan menemukan bugs, error, dan kelemahan dari aplikasi, mengetahui sejauh mana kualitas dari aplikasi, serta memastikan bahwa aplikasi sudah siap untuk digunakan.

1.2 GAMBARAN PROSES PENGUJIAN

Tim kami akan melakukan beberapa pengujian:

1. Unit Testing
Pengujian ini akan dilakukan dengan kotak hitam dan pengujian kotak putih, yang akan menggunakan *Branch Testing*.
2. Integration Testing
Pengujian ini memungkinkan pengujian dari semua unit yang diuji secara individual untuk diujikan secara keseluruhan.
3. Function Testing
Pengujian ini akan memastikan bahwa semua *use case* telah dilaksanakan dengan benar dengan memverifikasi jika mereka hadir di dalam aplikasi.

SKENARIO DAN HASIL TES

Bagian ini akan menggambarkan strategi yang direkomendasikan untuk merancang dan menerapkan tes yang diperlukan. Kami akan menjelaskan rincian tes yang perlu dilakukan untuk setiap item tes yang telah diidentifikasi sebelumnya. Tes yang akan dilakukan sebagai berikut :

- Unit Testing
- Integration Testing
- Function Testing

Selain itu, untuk masing-masing tes ini, setiap *test case* akan dijelaskan secara rinci. Untuk setiap *test case*, kami akan memberikan deskripsi, input atau langkah-langkah untuk melakukan *test case* tersebut, dan output atau hasil yang diharapkan.

2.1 UNIT TESTING

Unit testing akan menguji komponen beserta dengan fungsinya secara individu dengan fungsi. Ini pengujian tingkat rendah yang mencakup pengujian Black Box dan pengujian White Box. Dalam pengujian kotak hitam, batas-batas fungsi ini akan diuji untuk melihat apakah terjadi kesalahan di sana. pengujian kotak putih akan memverifikasi bahwa semua jalur dalam fungsi yang benar melalui basis Branch Testing.

2.1.1 FUNGSI GETFROMLOCATION

Tes akan dilakukan pada fungsi untuk mengambil lokasi asal berdasarkan pilihan user pada *combo box* yang telah tersedia. Lokasi tersebut akan digunakan lebih lanjut untuk menghitung jarak, tarif dan pencarian navigasi.

2.1.1.1 BLACK BOX TESTING

Setiap test case akan diujikan ketika combo box dalam keadaan tertutup.

Test Case 1: Memilih salah satu pilihan combo box ketika pilihan awal masih kosong.

Test Case 2: Memilih salah satu pilihan combo box ketika pilihan awal sudah ada.

Nama penguji:	Anneke
Tipe tes:	Unit
Nomor <i>test case</i>:	1
Nama <i>test case</i>:	getFromLocation_empty

Deskripsi <i>test case</i> :		Memilih salah satu pilihan combo box ketika pilihan awal masih kosong.	
Item yang diujikan			
1	Melihat apakah hasil yang didapatkan sesuai dengan pilihan combo box.		
Spesifikasi			
Input	Expected Output	Actual Output	Bug found?
Stasiun Bogor	Stasiun Bogor	Stasiun Bogor	No
Langkah pengujian			
1	Pilih menu “Navigasi” pada jendela Main Menu.		
2	Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.		

Nama penguji:	Anneke
Tipe tes:	Unit
Nomor test case:	2
Nama test case:	getFromLocation_unempty

Deskripsi <i>test case</i> :		Memilih salah satu pilihan combo box ketika pilihan awal sudah ada.	
Item yang diujikan			
1	Melihat apakah hasil yang didapatkan sesuai dengan pilihan combo box.		
Spesifikasi			
Input	Expected Output	Actual Output	Bug found?
Botani Square	Botani Square	Botani Square	No
Langkah pengujian			
1	Pilih menu “Navigasi” pada jendela Main Menu.		
2	Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.		
3	Pilih kembali combo box tempat asal, lalu pilih lagi salah satu tempat yang berbeda dengan pilihan pada langkah 2.		

2.1.2 FUNGSI GETTOLOCATION

Tes akan dilakukan pada fungsi untuk mengambil lokasi tujuan berdasarkan pilihan user pada *combo box* yang telah tersedia. Lokasi tersebut akan digunakan lebih lanjut untuk menghitung jarak, tarif dan pencarian navigasi.

2.1.2.1 BLACK BOX TESTING

Setiap test case akan diujikan ketika combo box dalam keadaan tertutup.

Test Case 1: Memilih salah satu pilihan combo box ketika pilihan awal masih kosong.

Test Case 2: Memilih salah satu pilihan combo box ketika pilihan awal sudah ada.

Nama penguji:		Anneke	
Tipe tes:		Unit	
Nomor <i>test case</i> :		1	
Nama <i>test case</i> :		getToLocation_empty	
Deskripsi <i>test case</i> :		Memilih salah satu pilihan combo box ketika pilihan awal masih kosong.	
Item yang diujikan			
1	Melihat apakah hasil yang didapatkan sesuai dengan pilihan combo box.		
Spesifikasi			
Input	Expected Output	Actual Output	Bug found?
Stasiun Bogor	Stasiun Bogor	Stasiun Bogor	No
Langkah pengujian			

1	Pilih menu “Navigasi” pada jendela Main Menu.
2	Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.
3	Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar.

Nama penguji:		Anneke	
Tipe tes:		Unit	
Nomor <i>test case</i> :		2	
Nama <i>test case</i> :		getToLocation_unempty	
Deskripsi <i>test case</i> :		Memilih salah satu pilihan combo box ketika pilihan awal sudah ada.	
Item yang diujikan			
1	Melihat apakah hasil yang didapatkan sesuai dengan pilihan combo box.		
Spesifikasi			
Input	Expected Output	Actual Output	Bug found?

Botani Square	Botani Square	Botani Square	No
Langkah pengujian			
1	Pilih menu “Navigasi” pada jendela Main Menu.		
2	Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.		
3	Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar.		
4	Pilih kembali combo box tempat tujuan, lalu pilih salah satu tempat yang berbeda dengan pilihan sebelumnya.		

2.1.3 FUNGSI COUNTDISTANCE

Tes akan dilakukan pada fungsi untuk melakukan perhitungan jarak berdasarkan kedua input lokasi yang telah diterima sebelumnya. Lokasi tersebut adalah lokasi awal dan lokasi tujuan berupa variabel string.

2.1.3.1 BLACK BOX TESTING

Pengujian akan dilakukan ketika dua buah input variabel lokasi telah diterima, lalu fungsi countDistance akan melakukan perhitungan jarak antar kedua variabel tersebut menggunakan data yang diambil dari database dengan perhitungan matematika sederhana.

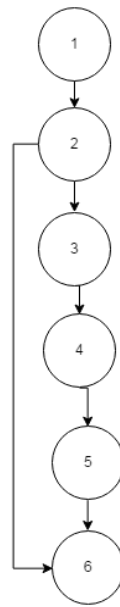
Test Case 1: Mendapatkan dua buah variabel lokasi yang berada di dalam satu jalur angkot yang sama.

Nama penguji:		Anneke	
Tipe tes:		Unit	
Nomor test case:		1	
Nama test case:		countDistance_samePath	
Deskripsi test case:		Mendapatkan dua buah variabel lokasi yang berada di dalam satu jalur angkot yang sama.	
Item yang diujikan			
1	Melihat apakah hasil perhitungan jarak yang didapatkan sesuai atau mendekati dengan perhitungan jarak sesungguhnya.		
Spesifikasi			
Input	Expected Output	Actual Output	Bug found?
Stasiun Bogor, Botani Square	2743 meter	2743 meter	No
Langkah pengujian			
1	Pilih menu “Navigasi” pada jendela Main Menu.		
2	Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.		
3	Pilih combo box tempat tujuan, lalu pilih salah satu		

	tempat yang tersedia dalam daftar.
4	Tekan tombol “OK”.

2.1.3.2 WHITE BOX TESTING

Basis Path Testing



Path ID	Path
1.1	1 - 2 - 3 - 4 - 5 - 6
1.2	1 - 2 - 6

2.1.4 FUNGSI COUNTFARE

Tes akan dilakukan pada fungsi untuk melakukan perhitungan tarif berdasarkan variabel jarak yang telah diterima sebelumnya. Perhitungan tarif yang dilakukan cukup sederhana dengan cara membagi tarif berdasarkan kriteria jarak tertentu.

2.1.4.1 BLACK BOX TESTING

Pengujian akan dilakukan ketika input variabel jarak telah diterima, lalu fungsi countFare akan melakukan perhitungan sederhana untuk menentukan perkiraan tarif yang harus dibayarkan untuk mencapai tempat tujuan.

Test Case 1: Mendapatkan variabel jarak = 0.

Test Case 2: Mendapatkan variabel jarak < 7km.

Test Case 3: Mendapatkan variabel jarak > 10 km.

Nama penguji:		Anneke
Tipe tes:		Unit
Nomor <i>test case</i>:		3
Nama <i>test case</i>:		countFare_zero
Deskripsi <i>test case</i>:		Mendapatkan variabel jarak = 0.
Item yang diujikan		
1	Melihat apakah hasil perkiraan tarif yang didapatkan sesuai	

	atau mendekati dengan tarif sesungguhnya.		
Spesifikasi			
Input	Expected Output	Actual Output	Bug found?
okm	Rp 3500	Rp 3500	No
Langkah pengujian			
1	Pilih menu “Navigasi” pada jendela Main Menu.		
2	Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.		
3	Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar.		
4	Tekan tombol “OK”.		

Nama Penguji:	Anneke
Tipe tes:	Unit
Nomor <i>test case</i>:	2
Nama <i>test case</i>:	countFare_low
Deskripsi <i>test case</i>:	Mendapatkan variabel jarak > 7km.

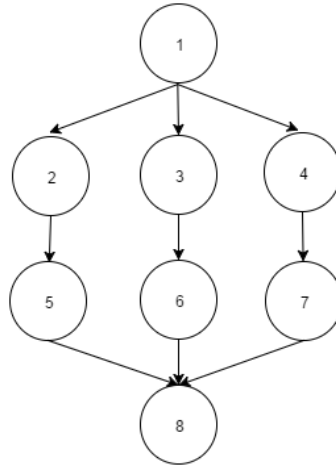
Item yang diujikan			
1	Melihat apakah hasil perkiraan tarif yang didapatkan sesuai atau mendekati dengan tarif sesungguhnya.		
Spesifikasi			
Input	Expected Output	Actual Output	Bug found?
8km	Rp 4000	Rp 4000	No
Langkah pengujian			
1	Pilih menu “Navigasi” pada jendela Main Menu.		
2	Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.		
3	Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar.		
4	Tekan tombol “OK”.		

	Anneke
Tipe tes:	Unit
Nomor test case:	3

Nama <i>test case</i> :		countFare_high	
Deskripsi <i>test case</i> :		Mendapatkan variabel jarak > 10km.	
Item yang diujikan			
1	Melihat apakah hasil perkiraan tarif yang didapatkan sesuai atau mendekati dengan tarif sesungguhnya.		
Spesifikasi			
Input	Expected Output	Actual Output	Bug found?
12km	Rp 5000	Rp 5000	No
Langkah pengujian			
1	Pilih menu “Navigasi” pada jendela Main Menu.		
2	Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.		
3	Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar.		
4	Tekan tombol “OK”.		

2.1.4.2 WHITE BOX TESTING

Basis Path Testing



Path ID	Path
2.1	1 - 2 - 5 - 8
2.2	1 - 3 - 6 - 8
2.3	1 - 4 - 7 - 8

2.2.5 FUNGSI GENERATENAVIGATION

Tes akan dilakukan pada fungsi untuk melakukan perhitungan jarak berdasarkan kedua input lokasi yang telah diterima sebelumnya. Lokasi tersebut adalah lokasi awal dan lokasi tujuan berupa variabel string.

2.1.3.1 BLACK BOX TESTING

Pengujian akan dilakukan ketika dua buah input variabel lokasi telah diterima, lalu fungsi generateNavigation akan melakukan pencarian navigasi tersebut menggunakan data yang diambil dari database dengan algoritma sederhana.

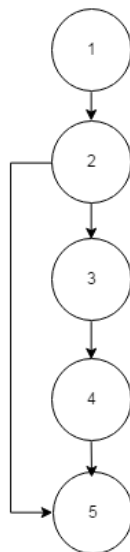
Test Case 1: Mendapatkan dua buah variabel lokasi yang berada di dalam satu jalur angkot yang sama.

Nama penguji:		Anne	
Tipe tes:		Unit	
Nomor test case:		1	
Nama test case:		generateNavigation_samePath	
Deskripsi test case:		Mendapatkan dua buah variabel lokasi yang berada di dalam satu jalur angkot yang sama.	
Item yang diujikan			
1	Melihat apakah hasil navigasi yang didapatkan sesuai dengan yang seharusnya ditempuh.		
Spesifikasi			
Input	Expected Output	Actual Output	Bug found?
Stasiun Bogor, Botani Square	02	02	No
Langkah pengujian			
1	Pilih menu “Navigasi” pada jendela Main Menu.		
2	Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.		

3	Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar.
4	Tekan tombol “OK”.

2.1.5.2 WHITE BOX TESTING

Basis Path Testing



Path ID	Path
3.1	1 - 2 - 3 - 4 - 5
3.2	1 - 2 - 5

2.2 INTEGRATION TESTING

Integration Testing adalah jenis pengujian dimana perangkat lunak dikombinasikan dan diuji untuk melihat apakah mereka berinteraksi sesuai dengan fungsi dan kebutuhan mereka. Pengujian ini dapat dilakukan secara progresif sampai seluruh sistem telah terintegrasi.

2.2.1 URUTAN PENGUJIAN

Integration Testing dimulai setelah setiap unit telah diuji secara individual. Untuk menghemat waktu, kami melakukan pengujian ini dengan urutan sebagai berikut:

1. Pertama, kita membagi integrasi menjadi beberapa bagian, lalu masing-masing akan diujikan.
2. Kedua, untuk setiap tes, beberapa *test case* akan dirancang. Dalam setiap *test case*, harus ada tepatnya satu komponen baru yang dianalisis.
3. Ketiga, untuk setiap komponen yang berkaitan kami akan mengujinya terlebih dahulu.

2.2.1.1 URUTAN KOMPONEN YANG DIUJIKAN

Dengan mempertimbangkan Integration dan Unit Testing, maka kami membagi pengujian ini ke dalam 7 komponen berikut:

- Jendela Splash
- Jendela Main Menu
- Jendela Daftar Angkot
- Jendela Navigasi
- Jendela Info Angkot
- Jendela Help
- Jendela About

2.2.1.2 PRASYARAT UNIT TESTING UNTUK SETIAP KOMPONEN

Untuk setiap komponen Integration Testing, kami telah membuat daftar prasyarat Unit Testing yang harus dilakukan terlebih dahulu, yaitu:

1. Jendela Splash
Untuk jendela Splash hanya konstruktor yang perlu diujikan.
2. Jendela Main Menu

- Button Daftar Angkot, Navigasi, Help, About.

Komponen berikut harus diujikan terlebih dahulu sebelum melakukan pengujian Jendela Main Menu.

3. Jendela Daftar Angkot

- List Daftar Angkot.

Komponen berikut harus diujikan terlebih dahulu sebelum melakukan pengujian Jendela Daftar Angkot.

4. Jendela Navigasi

- Combo Box Tempat Asal, Tempat Tujuan, Button OK, Label Informasi Navigasi.

Komponen berikut harus diujikan terlebih dahulu sebelum melakukan pengujian Jendela Navigasi.

5. Jendela Info Angkot

- Label Informasi Angkot, Gambar Peta Trayek.

Komponen berikut harus diujikan terlebih dahulu sebelum melakukan pengujian Jendela Info Angkot.

6. Jendela Help

- Label Informasi Help, Button Exit.

Komponen berikut harus diujikan terlebih dahulu sebelum melakukan pengujian Jendela Help.

7. Jendela About

- Label Informasi About, Button Exit.

Komponen berikut harus diujikan terlebih dahulu sebelum melakukan pengujian Jendela About.

2.2.2 METODE PENGUJIAN

Integration Testing akan dilakukan dengan menggunakan metode Sandwich karena metode ini merupakan kombinasi antara pengujian bottom-up dan top-down.

Kita bisa mulai integrasi sedini mungkin dalam tahap pengembangan perangkat lunak. Untuk Integration Testing, satu-satunya cara untuk menguji integrasi setiap *test case* adalah untuk menambahkan satu komponen dan mengujinya untuk melihat apakah Komponen tersebut bekerja dengan komponen lainnya.

2.2.3 JENDELA SPLASH

Hanya terdapat satu *test case* untuk jendela Splash.

Nama penguji:	Junior Lie
Test Case 1	Memanggil Jendela Splash
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat dipanggil, dengan cara memilih ikon Kiri Bang! pada smartphone.
Hasil test case:	Ok.

2.2.4 JENDELA MAIN MENU

Ada beberapa *test case* untuk jendela Main Menu, yaitu untuk melakukan penambahan, pengujian beberapa button yang ada pada jendela ini, serta untuk keluar dari jendela ini.

Nama penguji:	Junior Lie
Test Case 1	Memanggil Jendela Main Menu
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat dipanggil, dengan cara memilih ikon Kiri Bang! pada smartphone, lalu menunggu jendela Splash hingga berakhir.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
----------------------	------------

Test Case 2	Menambahkan Button Daftar Angkot
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Main Menu dipanggil.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 3	Menambahkan Button Navigasi
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Main Menu dipanggil.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 4	Menambahkan Button Help
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Main Menu dipanggil.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 5	Menambahkan Button About

Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Main Menu dipanggil.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 6	Memanggil Jendela Daftar Angkot
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat memanggil jendela Daftar Angkot, dengan cara memilih button Daftar Angkot.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 7	Memanggil Jendela Navigasi
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat memanggil jendela Navigasi, dengan cara memilih button Navigasi.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 8	Memanggil Jendela Help
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat memanggil

	jendela Help, dengan cara memilih button Help.
Hasil <i>test case</i>:	Ok.

Nama penguji:	Junior Lie
Test Case 9	Memanggil Jendela About
Deskripsi <i>test case</i>:	Melakukan pengujian apakah jendela ini dapat memanggil jendela About, dengan cara memilih button About.
Hasil <i>test case</i>:	Ok.

Nama penguji:	Junior Lie
Test Case 10	Menutup Jendela Main Menu
Deskripsi <i>test case</i>:	Melakukan pengujian apakah jendela ini dapat ditutup, dengan cara memilih button “Back” pada smartphone.
Hasil <i>test case</i>:	Ok.

2.2.5 JENDELA DAFTAR ANGKOT

Ada beberapa *test case* untuk jendela Daftar Angkot, yaitu untuk melakukan penambahan, pengujian list yang ada pada jendela ini, serta untuk keluar dari jendela ini.

Nama penguji:	Junior Lie
Test Case 1	Memanggil Jendela Daftar Angkot
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat dipanggil, dengan cara memilih button Daftar Angkot.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 2	Menambahkan List Daftar Angkot
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Daftar Angkot dipanggil.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 3	Memanggil Jendela Informasi Angkot
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat memanggil Jendela Informasi Angkot, dengan cara memilih salah satu dari list angkot yang ada.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
----------------------	------------

Test Case 4	Menutup Daftar Angkot
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat ditutup, dengan cara memilih button “Back” pada smartphone.
Hasil test case:	Ok.

2.2.6 JENDELA NAVIGASI

Ada beberapa *test case* untuk jendela Navigasi, yaitu untuk melakukan penambahan, pengujian beberapa combo box dan button yang ada pada jendela ini, serta untuk keluar dari jendela ini.

Nama penguji:	Junior Lie
Test Case 1	Memanggil Jendela Navigasi
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat dipanggil, dengan cara memilih button Navigasi.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 2	Menambahkan Combo Box Tempat Asal
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Navigasi dipanggil.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 3	Menambahkan Combo Box Tempat Tujuan
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Navigasi dipanggil.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 4	Menambahkan Button OK
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Navigasi dipanggil.
Hasil test case:	Ok.

Nama penguji:	Junior Lie
Test Case 5	Menutup Jendela Navigasi
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat ditutup, dengan cara memilih button “Back” pada smartphone.
Hasil test case:	Ok.

2.2.7 JENDELA INFO ANGKOT

Ada beberapa *test case* untuk jendela Info Angkot, yaitu untuk melakukan penambahan, pengujian beberapa label dan gambar yang ada pada jendela ini, serta untuk keluar dari jendela ini.

Nama penguji:	Hendry
Test Case 1	Memanggil Jendela Info Angkot
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat dipanggil, dengan cara memilih salah satu angkot dari list yang ada pada jendela Daftar Angkot.
Hasil test case:	Ok.

Nama penguji:	Hendry
Test Case 2	Menambahkan Label Informasi Angkot
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Info Angkot dipanggil.
Hasil test case:	Ok.

Nama penguji:	Hendry
Test Case 3	Menambahkan Gambar Trayek Angkot
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Info Angkot dipanggil.

Hasil test case:	Ok.
-------------------------	-----

Nama penguji:	Hendry
Test Case 4	Menutup Jendela Info Angkot
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat ditutup, dengan cara memilih button “Back” pada smartphone.
Hasil test case:	Ok.

2.2.8 JENDELA HELP

Ada beberapa *test case* untuk jendela Help, yaitu untuk melakukan penambahan, pengujian label dan button yang ada pada jendela ini, serta untuk keluar dari jendela ini.

Nama penguji:	Hendry
Test Case 1	Memanggil Jendela Help
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat dipanggil, dengan cara memilih button Help.
Hasil test case:	Ok.

Nama penguji:	Hendry
Test Case 2	Menambahkan Label Help

Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Help dipanggil.
Hasil test case:	Ok.

Nama penguji:	Hendry
Test Case 3	Menambahkan Button Exit
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela Help dipanggil.
Hasil test case:	Ok.

Nama penguji:	Hendry
Test Case 4	Menutup Jendela Help
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat ditutup, dengan cara memilih button Exit.
Hasil test case:	Ok.

2.2.9 JENDELA ABOUT

Ada beberapa *test case* untuk jendela About, yaitu untuk melakukan penambahan, pengujian label dan button yang ada pada jendela ini, serta untuk keluar dari jendela ini.

Nama penguji:	Hendry
Test Case 1	Memanggil Jendela About
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat dipanggil, dengan cara memilih button About.
Hasil test case:	Ok.

Nama penguji:	Hendry
Test Case 2	Menambahkan Label About
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela About dipanggil.
Hasil test case:	Ok.

Nama penguji:	Hendry
Test Case 3	Menambahkan Button Exit
Deskripsi test case:	Pengujian ini akan dijalankan secara otomatis ketika jendela About dipanggil.
Hasil test case:	Ok.

Nama penguji:	Hendry
----------------------	--------

Test Case 4	Menutup Jendela About
Deskripsi test case:	Melakukan pengujian apakah jendela ini dapat ditutup, dengan cara memilih button Exit.
Hasil test case:	Ok.

2.3 FUNCTION TESTING

Bagian ini akan memastikan bahwa perangkat lunak yang diujikan memenuhi persyaratan dan diterima dengan baik oleh pengguna. Setiap persyaratan akan dikaitkan dengan beberapa *test case*, menggunakan beberapa dengan data yang valid, dan beberapa dengan data yang tidak valid. Meskipun waktu yang ada sangat terbatas, kami telah mencantumkan semua fungsi utama serta *test case* untuk setiap fungsi tersebut. Salah satu manfaat dari pengujian ini adalah tim implementasi dapat memastikan bahwa setiap fungsi telah diimplementasikan dengan benar dan bahwa perangkat lunak telah bekerja baik dalam keadaan normal atau tidak normal.

2.3.1 MELIHAT DAFTAR DAN INFORMASI ANGKOT

Nama penguji:	Mahesa
Test Case	Melihat Daftar Angkot
Deskripsi test case:	Melihat Daftar Angkot yang tersedia di Bogor.
Input (Langkah Pengujian)	1. Pilih menu “Daftar Angkot” pada jendela Main Menu.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> Menampilkan daftar angkot yang tersedia di Bogor.

Hasil test case:	Ok.
-------------------------	-----

Nama penguji:	Mahesa
Test Case	Melihat Informasi Angkot
Deskripsi test case:	Melihat Informasi Angkot yang dipilih oleh pengguna.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Daftar Angkot” pada jendela Main Menu. 2. Pilih salah satu angkot yang ada pada daftar.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan informasi angkot yang dipilih pengguna.
Hasil test case:	Ok.

Nama penguji:	Mahesa
Test Case	Kembali ke Daftar Angkot
Deskripsi test case:	Kembali ke Daftar Angkot dari jendela Informasi Angkot.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Daftar Angkot” pada jendela Main Menu. 2. Pilih salah satu angkot yang ada pada daftar. 3. Tekan tombol “Back” pada smartphone.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan kembali daftar angkot yang tersedia di Bogor.
Hasil test case:	Ok.

Nama penguji:	Mahesa
Test Case	Kembali ke Main Menu
Deskripsi test case:	Kembali ke Main Menu dari jendela Daftar Angkot.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Daftar Angkot” pada jendela Main Menu. 2. Pilih salah satu angkot yang ada pada daftar. 3. Tekan tombol “Back” sebanyak 2 kali pada smartphone.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan kembali Main Menu dari Kiri Bang!.
Hasil test case:	Ok.

2.3.2 MELIHAT TRAYEK/RUTE SETIAP ANGKOT

Nama penguji:	Mahesa
Test Case	Melihat Peta Trayek Angkot
Deskripsi test case:	Melihat Peta Trayek Angkot yang dipilih oleh pengguna.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Daftar Angkot” pada jendela Main Menu. 2. Pilih salah satu angkot yang ada pada daftar.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan peta trayek angkot yang dipilih pengguna.
Hasil test case:	Ok.

2.3.3 MEMASUKKAN TEMPAT ASAL

Nama penguji:	Mahesa
Test Case	Memilih Tempat Asal
Deskripsi test case:	Melihat dan memilih Tempat Asal untuk input navigasi angkot.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Navigasi” pada jendela Main Menu. 2. Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan daftar tempat asal yang ada di Bogor. • Pilihan pengguna tersimpan sebagai input.
Hasil test case:	Ok.

Nama penguji:	Mahesa
Test Case	Mengganti Tempat Asal
Deskripsi test case:	Melihat dan mengganti Tempat Asal yang sudah terisi sebelumnya.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Navigasi” pada jendela Main Menu. 2. Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar. 3. Pilih kembali combo box tempat asal, lalu pilih lagi salah satu tempat yang berbeda dengan pilihan pada langkah 2.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Pilihan pengguna yang baru tersimpan sebagai input.

Hasil test case:	Ok.
-------------------------	-----

Nama penguji:	Mahesa
Test Case	Kembali ke Main Menu
Deskripsi test case:	Kembali ke Main Menu dari jendela Navigasi.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Navigasi” pada jendela Main Menu. 2. Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar. 3. Tekan tombol “Back” pada smartphone.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan kembali Main Menu dari Kiri Bang!.
Hasil test case:	Ok.

2.3.4 MEMASUKKAN TEMPAT TUJUAN

Nama penguji:	Mahesa
Test Case	Memilih Tempat Tujuan
Deskripsi test case:	Melihat dan memilih Tempat Tujuan untuk input navigasi angkot.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Navigasi” pada jendela Main Menu. 2. Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar. 3. Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar.

Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan daftar tempat tujuan yang ada di Bogor. • Pilihan pengguna tersimpan sebagai input.
Hasil test case:	Ok.

Nama penguji:	Maheza
Test Case	Mengganti Tempat Tujuan
Deskripsi test case:	Melihat dan mengganti Tempat Tujuan yang sudah terisi sebelumnya.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Navigasi” pada jendela Main Menu. 2. Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar. 3. Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar. 4. Pilih kembali combo box tempat tujuan, lalu pilih salah satu tempat yang berbeda dengan pilihan sebelumnya.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Pilihan pengguna yang baru tersimpan sebagai input.
Hasil test case:	Ok.

Nama penguji:	Maheza
Test Case	Kembali ke Main Menu
Deskripsi test case:	Kembali ke Main Menu dari jendela Navigasi.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Navigasi” pada jendela Main Menu. 2. Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar.

	<ol style="list-style-type: none"> Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar. Tekan tombol “Back” pada smartphone.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> Menampilkan kembali Main Menu dari Kiri Bang!.
Hasil test case:	Ok.

2.3.5 MENDAPATKAN PRAKIRAAN TARIF, JARAK, DAN NAVIGASI DARI TEMPAT ASAL KE TUJUAN

Nama penguji:	Mahesa
Test Case	Mendapatkan prakiraan tarif, jarak dan navigasi
Deskripsi test case:	Mendapatkan prakiraan tarif, jarak dan navigasi berdasarkan tempat asal dan tempat tujuan yang dipilih pada jendela Navigasi
Input (Langkah Pengujian)	<ol style="list-style-type: none"> Pilih menu “Navigasi” pada jendela Main Menu. Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar. Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar. Tekan tombol “OK”.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> Menampilkan prakiraan tarif, jarak dan navigasi untuk pengguna.
Hasil test case:	Ok.

Nama penguji:	Mahesa
----------------------	--------

Test Case	Kembali ke Main Menu
Deskripsi test case:	Kembali ke Main Menu dari jendela Navigasi.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih menu “Navigasi” pada jendela Main Menu. 2. Pilih combo box tempat asal, lalu pilih salah satu tempat yang tersedia dalam daftar. 3. Pilih combo box tempat tujuan, lalu pilih salah satu tempat yang tersedia dalam daftar. 4. Tekan tombol “OK”. 5. Tekan tombol “Back” pada smartphone.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan kembali Main Menu dari Kiri Bang!.
Hasil test case:	Ok.

2.3.6 MELIHAT PETUNJUK PENGUNAAN APLIKASI (HELP)

Nama penguji:	Mahesa
Test Case	Melihat Jendela Help
Deskripsi test case:	Melihat petunjuk penggunaan aplikasi Kiri Bang! pada Jendela Help
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih button “Help” pada jendela Main Menu.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan jendela Help.
Hasil test case:	Ok.

Nama penguji:	Mahesa
Test Case	Kembali ke Main Menu
Deskripsi test case:	Kembali ke Main Menu dari jendela Help.
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih button “Help” pada jendela Main Menu. 2. Tekan tombol “Back” pada smartphone.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan kembali Main Menu dari Kiri Bang!.
Hasil test case:	Ok.

2.3.7 MELIHAT ABOUT

Nama penguji:	Mahesa
Test Case	Melihat Jendela About
Deskripsi test case:	Melihat informasi mengenai pengembang aplikasi Kiri Bang! pada Jendela About
Input (Langkah Pengujian)	<ol style="list-style-type: none"> 1. Pilih button “About” pada jendela Main Menu.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none"> • Menampilkan jendela About.
Hasil test case:	Ok.

Nama penguji:	Mahesa
Test Case	Kembali ke Main Menu
Deskripsi <i>test case</i>:	Kembali ke Main Menu dari jendela About.
Input (Langkah Pengujian)	<ol style="list-style-type: none">1. Pilih button “About” pada jendela Main Menu.2. Tekan tombol “Back” pada smartphone.
Output (Hasil yang diharapkan)	<ul style="list-style-type: none">• Menampilkan kembali Main Menu dari Kiri Bang!.
Hasil <i>test case</i>:	Ok.