

BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

Tahap implementasi adalah tahap untuk merepresentasikan atau menuangkan rancangan yang telah dibuat ke dalam bentuk program dengan mengikuti dan menerapkan semua yang dirancang pada tahap perancangan. Sedangkan tahap pengujian adalah tahap pengetesan terhadap aplikasi yang dibuat. Bab ini merupakan hasil dari proses fase konstruksi.

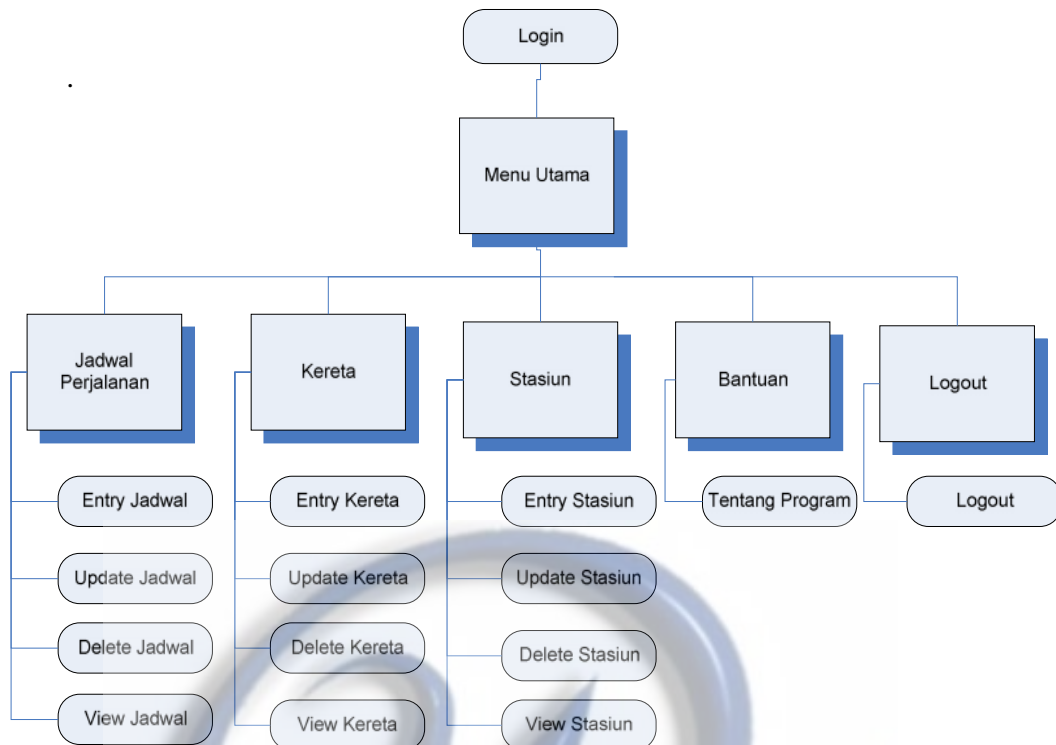
5.1 Prototype

Prototype program dibuat dapat memenuhi tujuan implementasi diantaranya yang dapat mewakili proses data input, pemrosesan data, dan proses data output. Pembagian yang dilakukan pada tahap implementasi dari rancangan program adalah sebagai berikut:

1. Pengembangan *prototype* penggambaran peta.
2. Pengembangan *prototype entry* data kereta
3. Pengembangan *prototype* pencarian rute tercepat dengan algoritma dijkstra
4. Pengembangan *prototype entry* data jadwal perjalanan
5. Pengembangan *prototype* proses pergerakan kereta
6. Pengembangan *prototype view* data stasiun
7. Pengembangan *prototype view* data kereta
8. Pengembangan *prototype view* data jadwal.

5.2 Struktur Menu

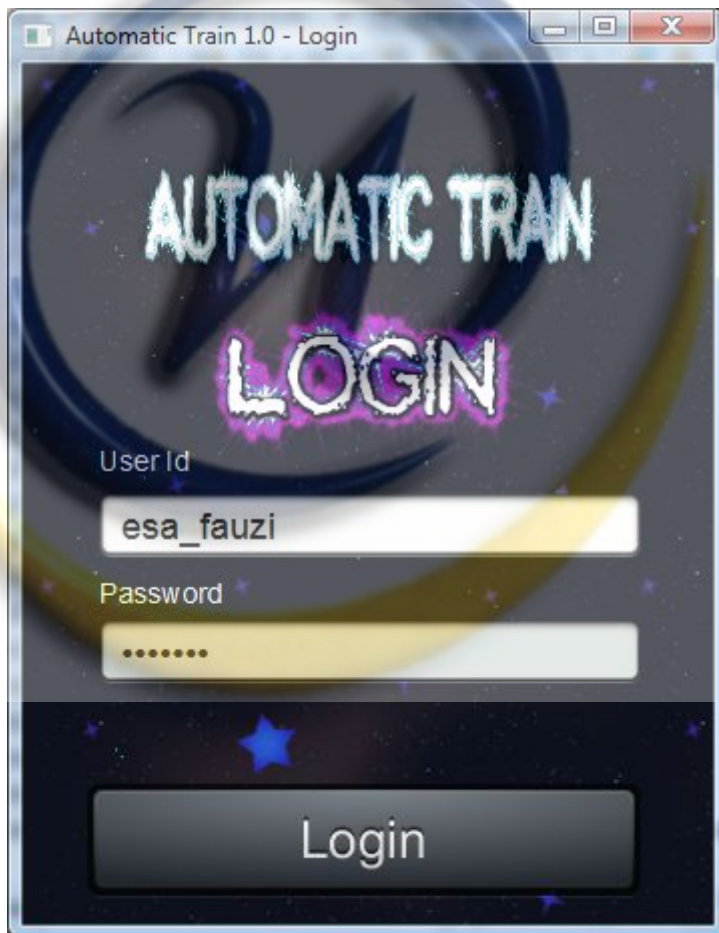
Fungsi-fungsi yang dirancang dibagi ke dalam beberapa menu yang bertujuan untuk memudahkan *user* dalam menggunakan aplikasi ini. Untuk menggambarkan menu-menu pada program, berikut adalah struktur menu dari perangkat lunak visualisasi perjalanan kereta



Gambar 5.1 Struktur Menu

Gambar 5.1 menunjukkan bahwa menu utama bisa diakses setelah melakukan login terlebih dahulu. Menu utama kemudian dibagi menjadi 5 sub menu yaitu, sub menu jadwal perjalanan, sub menu kereta, sub menu stasiun, sub menubantuan dan logout. Pada sub menu jadwal perjalanan dibagi lagi menjadi 4 menu yaitu menu *entry* jadwal, *update* jadwal, *delete* jadwal, dan *view* jadwal. Keempat menu tersebut memiliki fungsi yang sama sesuai dengan namanya, seperti *entry* jadwal yaitu untuk memasukkan jadwal baru ke database.

Pada menu sub kereta dan stasiun masing-masing juga dibagi kedalam 4 menu, yaitu menu untuk *mengentrykan*, *mengupdate*, *mendelete*, dan melihat data(*view*). Lalu pada sub menu bantuan terdapat menu yang berisikan sekilas tentang perangkat lunak visualisasi perjalanan kereta. Kemudian yang terakhir adalah sub menu *logout* untuk keluar dari aplikasi.



5.3.2 Menu Utama

Setelah login, maka akan muncul tampilan seperti pada gambar 5.3. Gambar 5.3 ini merupakan form menu utama karena dapat memanggil form-form lainnya. Pada form ini terdapat *text-text field* yang menyimpan informasi mengenai kereta api yang berjalan seperti id kereta, rute, dan sebagainya.

Menu utama di atas terdapat 5 buah menu *bar*. Masing-masing berisikan menu *item*. Pada menu *bar* jadwal terdapat menu *item entry* jadwal, *update* jadwal, *delete* jadwal dan *view* jadwal. Pada menu *bar* kereta terdapat menu *item entry* kereta, *update* kereta, *delete* kereta, dan *view* kereta. Pada menu *bar* stasiun hanya ada menu *item view* stasiun karena data stasiun tidak bisa ditambah atau dihapus (karena peta visualisasi). Pada menu *bar* bantuan terdapat menu *item* tentang aplikasi yang berisikan informasi aplikasi dan pada menu *bar* logout terdapat menu *item logout* untuk keluar aplikasi.

Pada menu utama terdapat gambar visualisasi kereta. Objek-objek stasiun digambarkan dengan rumah. Objek-objek rel digambarkan dengan *graph*. Pada *graph* yang digambar didefinisikan dua buah objek yaitu *vertex* dan *segmen*. *Vertex* sebagai titik dan *segmen* sebagai penghubung antar *vertex* yang didalamnya memuat jarak. Kemudian objek kereta akan digambarkan dengan lingkaran. Lingkaran tersebut nanti akan bergerak apabila sudah memiliki jadwal perjalanan

Automatic Train Simulation 1.0 - Primary Menu

Jadwal Perjalanan Kereta Stasiun Bantuan Sign Out

Id Kereta Jenis Kereta Nama lokomotif

Rute Posisi terakhir Warna Kelas

Stasiun Java SE ST1

Stasiun Java EE ST2

Stasiun Java ME ST3

Stasiun Java FX ST4

Stasiun Oracle ST5

Stasiun SQL ST6

Stasiun Java FX ST7

Stasiun Java FX ST8

Stasiun CSS ST9

Jumlah Stasiun : 9 User : esa_fauzi Form AT-001

Automatic Train Simulation 1.0 - Entry Jadwa...

▼ Pilih Kereta

Id Kereta

Nama Trayek

▼ Tanggal dan Waktu

Tanggal Mulai dd-mm-yyyy

Pukul hh24:mm:ss

Pengulangan Jadwal /mnt /jam /hari

Id Jadwal

▼ Pilih Stasiun

Stasiun

Rute

User: esa_fauzi FORM AT-002

Automatic Train Simulation 1.0 - View Jadwal

ID Jadwal	Id Kereta	Nama Trayek	Rute	Tanggal & Waktu	(/hari)	(/jam)	(/menit)
J1	K001	GAMBIR-BANDUNG	ST1-ST5	2013-01-15 13:38:00.0	0	0	1
J2	K002		ST5-ST1	2013-01-15 13:39:00.0	0	0	1

Refresh Edit Hapus

User : esa_fauzi FORM AT-004

Automatic Train Simulation 1.0 - Entry Kereta

▼ Jenis Kereta
Jenis: Kereta Penumpang Baru Hapus
Simpan Batal

▼ Kelas Kereta
Kelas: Bisnis Baru Hapus
Simpan Batal

▼ Lokomotif
Lokomotif: C301 Baru Hapus
Simpan Batal

▼ Nama Kereta
Nama: Argo Parahyangan Baru Hapus
Simpan Batal

▼ Identitas Kereta
Id Kereta: K001
Lokasi: V114
Warna: BLUE
Simpan Batal

User: esa_fauzi FORM AT-003

Automatic Train Simulation 1.0 - View Kereta

Jenis Kereta	Kelas	Lokomotif	Nama/Deskripsi Kereta	Id Kereta	Lokasi	Warna
Kereta P...	Argo	C300	Argo Parahyangan	K001	V128	AQUA
Kereta P...	Argo	C300	Argo Parahyangan	K002	V54	BLUE

Refresh Edit Delete

User : esa_fauzi FORM AT-006



```
SQL*Plus: Release 11.2.0.2.0 Production on Kam Jan 10 23:11:50 2013
Copyright (c) 1982, 2010, Oracle. All rights reserved.

SQL> conn
Enter user-name: esa_fauzi
Enter password:
Connected.
SQL> select count(*) from hasil_dijkstra;

  COUNT(*)
-----
         0

SQL>
```

```
Run SQL Command Line
SQL*Plus: Release 11.2.0.2.0 Production on Kam Jan 10 23:11:50 2013
Copyright (c) 1982, 2010, Oracle. All rights reserved.
SQL> conn
Enter user-name: esa_fauzi
Enter password:
Connected.
SQL> select count(*) from hasil_dijikstra;

COUNT(*)
-----
0

SQL> select count(*) from hasil_dijikstra;

COUNT(*)
-----
21462

SQL>
```

Oracle SQL Developer : Table ESA_FAUZI.HASIL_DIJIKSTRA@ESA_FAUZI

File Edit View Navigate Run Versioning Tools Help

Connections x ... x

Connections

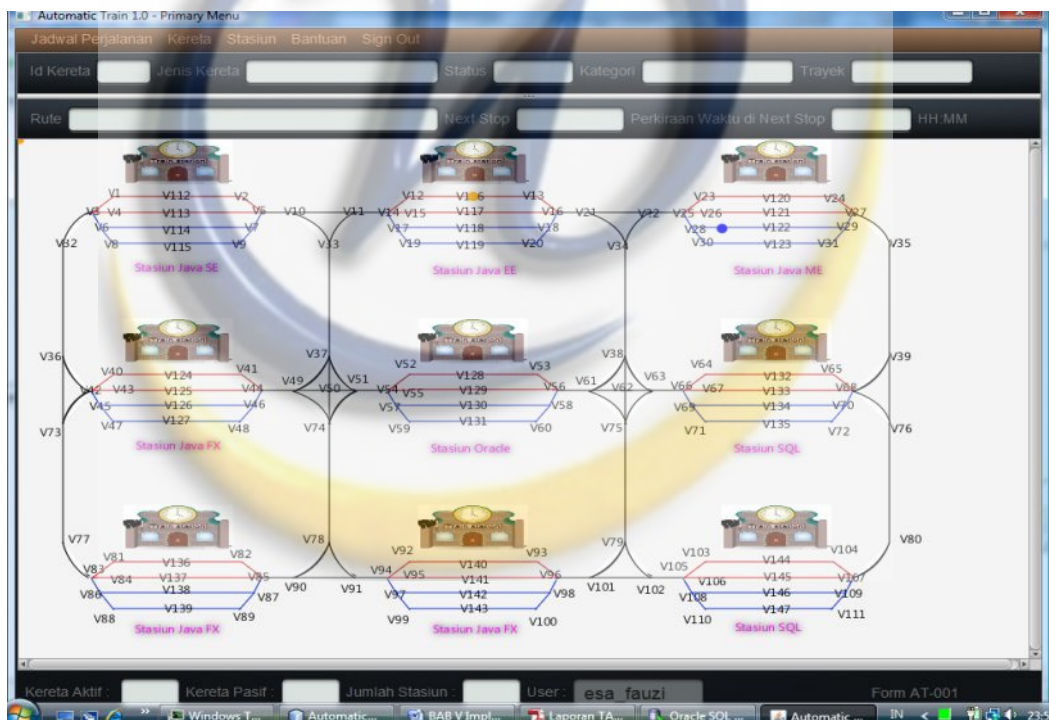
- Auto-Generated Local Cor
- / as sysdba
- ANONYMOUS
- ESA_FAUZI
 - Tables (Filtered)
 - FORM_MENU
 - HASIL_DIJIKS
 - START
 - FINISH
 - VIA
 - JARAK
 - JADWAL
 - JENIS KERET
 - KELAS KERET

Columns Data Constraints Grants Statistics Triggers Flashback Dependencies De

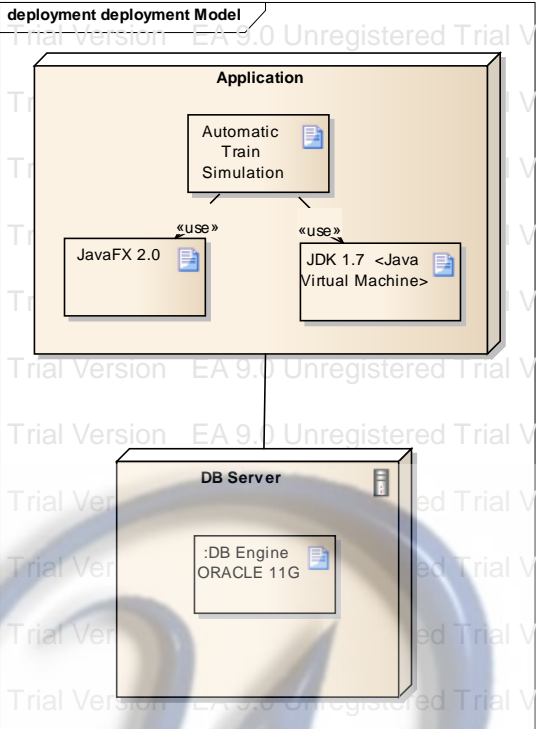
Sort.. Filter: Actions...

	START	FINISH	VIA
1	V19	V71	V19-V17-V14-V15-V117-V16-V21-V34-V38-V6:
2	V41	V80	V41-V44-V49-V74-V78-V91-V94-V95-V141-V94
3	V41	V81	V41-V44-V46-V126-V45-V42-V73-V77-V83-V8:
4	V41	V82	V41-V44-V46-V126-V45-V42-V73-V77-V83-V8:
5	V41	V83	V41-V44-V46-V126-V45-V42-V73-V77-V83
6	V41	V84	V41-V44-V46-V126-V45-V42-V73-V77-V83-V8:
7	V41	V85	V41-V44-V49-V74-V78-V90-V85
8	V41	V86	V41-V44-V49-V74-V78-V90-V85-V87-V138-V84
9	V41	V87	V41-V44-V49-V74-V78-V90-V85-V87
10	V41	V88	V41-V44-V49-V74-V78-V90-V85-V87-V89-V139
11	V41	V89	V41-V44-V49-V74-V78-V90-V85-V87-V89

Editing







5.6. Hasil Pengujian

Hasil pengujian dibagi menjadi beberapa bagian dan diurutkan berdasarkan bagian paling penting pada aplikasi ini.

Tabel 5.1 Pengujian Proses Perjalanan Kereta

Prosedur Pengujian	Data Masukan	Data Keluaran	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
ambilRuteTercepat()	Vertex Stasiun : List	Rute Tercepat : List	Rute tercepat dari setiap lokasi vertex stasiun, setiap stasiun memiliki 4 vertex jadi terdapat 4 rute tercepat	Mendapatkan rute tercepat dari 4 vertex lokasi stasiun dari posisi awal	sesuai
sortAlternatifJalur()	Rute Tercepat : List	-	Rute terpendek dari vertex-vertex stasiun di sorting dimulai dari yang terkecil	Rute dengan jarak yang terpendek berada di posisi pertama	sesuai
pesanJalan()	Rute	Status pesanJalan	Melakukan pemesanan terhadap vertex-vertex yang akan dilewati, namun apabila gagal vertex yang dipesan dilepas kembali	Vertex-vertex berhasil dipesan apabila vertex tersebut belum dipesa oleh kereta lain. Vertex yang dipesan juga akan dilepas kembali jika pemesanan gagal	Sesuai
pesanVertex()	Vertex	Status pesan vertex	Mengganti status sebuah vertex dari belum dipesan menjadi sudah dipesan	Status berhasil dirubah jika vertex belum dipesan	Sesuai
lepasVertex()	Vertex	status lepas vertex	Mengganti status sebuah vertex dari sudah dipesan menjadi belum dipesan	Status berhasil dirubah	Sesuai

Tabel 5.2 Pengujian Proses Perjalanan Kereta (lanjut)

Prosedur Pengujian	Data Masukan	Data Keluaran	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
Jalan()	Rute	-	Kereta berjalan berdasarkan rute yang telah dipesan	Kereta berjalan berdasarkan rute yang telah dipesan	Sesuai
ExecuteJadwal	Jadwal	-	Kereta melakukan proses perjalanansesuai jadwal yang telah disimpan	Kereta melakukan proses perjalanansesuai jadwal yang telah disimpan dan jadwal dapat diulang sesuia dengan pengulangan	Sesuai
Menjalankan 1 Kereta dengan 1 tujuan stasiun	Stasiun tujuan	-	Kereta berhasil menuju stasiun yang dituju	Kereta berhasil menuju stasiun yang dituju	Sesuai
Menjalankan 2 Kereta dengan rute yang tidak beririsan	Stasiun tujuan	-	Kedua kereta berjalan dan tidak mengalami masalah	Kedua kereta berjalan dan tidak mengalami masalah	Sesuai
Menjalankan 2 Kereta dengan rute yang beririsan	Stasiun tujuan	-	Semua kereta berjalan dan tidak mengalami masalah	Semua kereta berjalan dan tidak mengalami masalah	Sesuai
Menjalankan 3 Kereta dengan rute yang beririsan	Stasiun tujuan	-	Semua kereta berjalan dan tidak mengalami masalah	Semua kereta berjalan dan tidak mengalami masalah	Sesuai
Menjalankan 4 Kereta dengan rute yang beririsan	Stasiun tujuan	-	Semua kereta berjalan dan tidak mengalami masalah	Semua kereta berjalan dan tidak mengalami masalah	Sesuai
Menjalankan 6 Kereta dengan rute yang beririsan	Stasiun tujuan	-	Semua kereta berjalan dan tidak mengalami masalah	Semua kereta berjalan dan tidak mengalami masalah	Sesuai

Tabel 5.3 Pengujian *Entry* dan *View* Kereta

Prosedur Pengujian	Data Masukan	Data Keluaran	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
Menampilkan data jenis kereta	Jenis kereta	-	Menampilkan data jenis kereta pada combo box	Berhasil Menampilkan data jenis kereta pada combo box	Sesuai
Menampilkan data kelas kereta	Kelas kereta	-	Menampilkan data kelas kereta pada combo box	Berhasil Menampilkan data kelas kereta pada combo box	Sesuai
Menampilkan data lokomotif kereta	Lokomotif kereta	-	Menampilkan data lokomotif kereta pada combo box	Berhasil Menampilkan data lokomotif kereta pada combo box	sesuai
Menampilkan data nama kereta	Nama kereta	-	Menampilkan data nama kereta pada combo box	Berhasil Menampilkan data nama kereta pada combo box	Sesuai
Insert delete data jenis kereta	Jenis kereta	-	Data dapat disimpan dan dihapus	Data dapat disimpan dan dihapus	Sesuai
Insert delete data kelas kereta	Kelas kereta	-	Data dapat disimpan dan dihapus	Data dapat disimpan dan dihapus	Sesuai
Insert delete data lokomotif kereta	Lokomotif kereta	-	Data dapat disimpan dan dihapus	Data dapat disimpan dan dihapus	Sesuai
Insert delete data nama kereta	Nama kereta	-	Data dapat disimpan dan dihapus	Data dapat disimpan dan dihapus	Sesuai
Insert data kereta	kereta	-	Data kereta berhasil disimpan	Data kereta berhasil disimpan	Sesuai
Validasi input terhadap data yang sudah dimasukkan (input data yang sama)	Jenis/kelas/lokomotif>Nama/kereta yang bernilai sama	Pesan error	Data tidak bisa disimpan karena harus unik	Data tidak berhasil disimpan, ada pesan gagal	Sesuai
View data kereta	-	list kereta	Data kereta yang disimpan dapat ditampilkan	Data kereta yang disimpan dapat ditampilkan	Sesuai
Edit data kereta	kereta	-	Data dapat dirubah	Data dapat dirubah	Sesuai
Hapus data kereta	kereta	-	Data dapat dihapus	Data dapat dihapus	Sesuai
Hapus data kereta yang sudah memiliki jadwal	kereta	-	Data tidak bisa dihapus, harus menghapus data jadwal dulu	Data tidak bisa dihapus	Sesuai

Tabel 5.4 Pengujian *Entry Jadwal, View Jadwal, dan View stasiun*

Prosedur Pengujian	Data Masukan	Data Keluaran	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
Menampilkan data id kereta	kereta	-	Menampilkan data jenis kereta pada combo box	Berhasil Menampilkan data id kereta pada combo box	Sesuai
Menampilkan data stasiun kereta	stasiun	-	Menampilkan data stasiun pada combo box	Berhasil Menampilkan data stasiun pada combo box	Sesuai
Memasukkan dan menghapus stasiun tujuan	stasiun	-	Memasukkan dan menghapus stasiun tujuan	Berhasil Memasukkan dan menghapus stasiun tujuan box	sesuai
Menyimpan data jadwal	Jadwal	-	Menyimpan data jadwal	Berhasil Menyimpan data jadwal	Sesuai
Validasi terhadap insert data yang sama	jadwal	-	Gagal <i>insert</i> , karena data sudah dimasukkan	Gagal <i>insert</i> , karena data sudah dimasukkan	Sesuai
Validasi terhadap tipe data masukan	jadwal	-	Gagal insert, karena tipe data yang dimasukkan tidak sesuai	Gagal insert, karena tipe data yang dimasukkan tidak sesuai	Sesuai
View data jadwal	-	list jadwal	Data jadwal yang disimpan dapat ditampilkan	Data jadwal yang disimpan dapat ditampilkan	Sesuai
Edit data jadwal	jadwal	-	Data dapat dirubah	Data dapat dirubah	Sesuai
Hapus data jadwal	jadwal	-	Data dapat dihapus	Data dapat dihapus	Sesuai

Tabel 5.5 Pengujian Pembentukan Rute Tercepat

Prosedur Pengujian	Data Masukan	Data Keluaran	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
makeMatrixAdjacency	List Vertex, List Segmen	Matrix[][]	Membuat matrix ketetanggan dari vertex dan segmen	Berhasil membuat matrix ketetanggan dari vertex dan segmen	Sesuai
cariPath	Listt vertex, list segmen	Path	Mencari path dari vertex satu ker vertex lain	Berhasil mencari path dari vertex satu ker vertex lain	Sesuai
calculateDistance	Rute: List	-	Menhitung jarak total	Berhasil mengitung Jarak total	Sesuai
getShortedPath()	Vertex List, List Segmen, jarak	Path terpendek	Mendapatkan rute terpendek	Berhasil mendapatkan rute terpendek	Sesuai
insertDijkstratoDb	Path terpendek	-	Menyimpan data rute terpendek ke dalam database	Berhasil menyimpan data rute terpendek ke dalam database	Sesuai