



TUGAS AKHIR - KI1502

RANCANG BANGUN APLIKASI EDITOR ALBUM FOTO DIGITAL 'FOTOKITA' BERBASIS DESKTOP

**M. Arief Ridwan
NRP 5112100097**

**Dosen Pembimbing I
Rizky Januar Akbar, S.Kom., M.Eng.**

**Dosen Pembimbing II
Dr. tech. Ir. R. V. Hari Ginardi., M.Sc.**

**JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2016**

[Halaman ini sengaja dikosongkan]



UNDERGRADUATE THESES - KI1502

DESIGN AND IMPLEMENTATION OF DESKTOP BASED DIGITAL PHOTOBOOK EDITOR 'FOTOKITA'

**M. Arief Ridwan
NRP 5112100097**

**Supervisor I
Rizky Januar Akbar, S.Kom., M.Eng.**

**Supervisor II
Dr. tech. Ir. R. V. Hari Ginardi., M.Sc.**

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2016**

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

RANCANG BANGUN APLIKASI EDITOR ALBUM FOTO DIGITAL 'FOTOKITA' BERBASIS DESKTOP

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Rekayasa Perangkat Lunak
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh

M. Arief Ridwan
NRP : 5112 100 097

Disetujui oleh Dosen Pembimbing

1. Rizky Januar Akbar, S. Kom. Eng.
NIP: 19871013 201404 (Pembimbing 1)
2. Dr. tech. Ir. R. V. Hari Ginandhi, M. Sc.
NIP: 19650518 199203 1003 (Pembimbing 2)



SURABAYA
Juni, 2016

[Halaman ini sengaja dikosongkan]

RANCANG BANGUN APLIKASI EDITOR ALBUM FOTO DIGITAL ‘FOTOKITA’ BERBASIS DESKTOP

Nama Mahasiswa : M. Arief Ridwan
NRP : 5112100097
Jurusan : Teknik Informatika FTIF-ITS
Dosen Pembimbing 1 : Rizky Januar Akbar,
S.Kom.,M.Eng.
Dosen Pembimbing 2 : Dr. tech. Ir. R. V. Hari Ginardi.,
M.Sc.

Abstrak

Berfoto sudah menjadi kebiasaan yang sering dilakukan oleh masyarakat luas masa kini. Kebutuhan akan bentuk fisik dari sebuah foto kenangan dalam album merupakan salah satu kebutuhan yang saat ini juga banyak peminatnya. Salah satu solusi untuk dapat membuat album foto dengan mudah dan sesuai dengan apa yang kita inginkan adalah pada website photobookindonesia.com. Namun untuk mengaksesnya butuh koneksi internet yang cepat dan stabil mengingat kebutuhan website tersebut cukup besar.

Aplikasi Fotokita berbasis desktop menjadi salah satu solusi untuk membuat desain album foto digital secara offline. Dengan tidak dibutuhkannya koneksi internet kita dapat dengan mudah merancang album foto kenangan kita dimanapun tanpa harus terhubung dengan jaringan internet.

Untuk negara dengan kecepatan koneksi internet yang tidak terlalu cepat seperti Indonesia, aplikasi fotokita akan sangat membantu orang-orang yang ingin mengabadikan foto digital mereka dalam sebuah album tanpa harus tersambung dengan jaringan yang cepat dan stabil.

Fitur yang sudah diimplementasikan dalam aplikasi editor ini antara lain penambahan gambar, peningkatan kualitas citra (image enhancement), penambahan teks, pengubahan posisi

*gambar, perubahan ukuran gambar, dan rotasi gambar.
Masing-masing fitur berjalan dengan baik.*

Kata kunci: Album foto, Digital, Fotokita

DESIGN AND IMPLEMENTATION OF DESKTOP BASED DIGITAL PHOTOBOOK EDITOR 'FOTOKITA'

Student's Name : M. Arief Ridwan
Student's ID : 5112100097
Department : Teknik Informatika FTIF-ITS
First Advisor : Rizky Januar Akbar, S.Kom., M.Eng.
Second Advisor : Dr. tech. Ir. R. V. Hari Ginardi.,
M.Sc.

Abstract

Photograph now has become a habit that often done by the public today. Demand of a physical form of a photo album is one of the biggest demand on the public lifestyle. One of the solution is photobookindonesia.com, a web based photobook editor, but for a country with under average internet connection like Indonesia, accessing a big website like photobookindonesia.com is very hard. How we can design if we can't access easily the photobook editor, so the other solution is Fotokita.

Fotokita is a desktop based photobook editor with no need of internet access to design a photobook. With no need of connection we can easily design a photobook everywhere want including a location with no internet connection.

For a developing country like Indonesia, Fotokita will be very useful for people to keep their photo into a fancy photo album with no need of good internet connection.

Some of features that have been implemented are picture addition, image enhancement, text addition, and image transformation..

Keywords: Photobook, Digital, Photo Album

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Alhamdulillahrabbi'l'amin, segala puji bagi Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul **“RANCANG BANGUN APLIKASI EDITOR ALBUM FOTO DIGITAL ‘FOTOKITA’ BERBASIS DESKTOP”**.

Pengerjaan tugas akhir ini merupakan suatu kesempatan yang sangat baik bagi penulis. Dengan pengerjaan tugas akhir ini, penulis bisa belajar lebih banyak untuk memperdalam dan meningkatkan apa yang telah didapatkan penulis selama menempuh perkuliahan di Teknik Informatika ITS. Dengan tugas akhir ini penulis juga dapat menghasilkan suatu implementasi dari apa yang telah penulis pelajari.

Selesaiannya tugas akhir ini tidak lepas dari bantuan dan dukungan beberapa pihak. Sehingga pada kesempatan ini penulis mengucapkan syukur dan terima kasih kepada:

1. Allah SWT dan Nabi Muhammad SAW.
2. Ayah dan Ibu penulis, yang selalu mendo'akan yang terbaik bagi anak terakhirnya yaitu Saya selaku penulis agar segera menyelesaikan kuliah.
3. Bapak Rizky Januar Akbar, S.Kom., M.Eng. selaku pembimbing I, yang selalu memberikan motivasi, arahan dan membantu sekaligus membimbing penulis selama pengerjaan tugas akhir.
4. Bapak Dr. tech. Ir. Hari Ginardi., M.Sc. selaku pembimbing II yang juga telah dengan sabar membantu, membimbing dan memberikan ketenangan saat pengerjaan tugas akhir ini.
5. Bapak Dr. Darlis Herumurti S.Kom., M.Kom. selaku Kepala Jurusan Teknik Informatika ITS, Bapak Radityo Anggoro, S.Kom.,M.Sc. selaku koordinator TA, dan segenap dosen Teknik Informatika yang telah memberikan ilmunya.

6. Dewi Maya Fitriana, menemani atau ditemani, menghibur dan memberikan ketenangan disaat penulis berada pada titik terberat menjalankan tugas akhir ini.
7. Kelompok TA “bit.ly/cinTA114_” Dala, Fadri, Rina, Dwi, yang sudah susah payah menjalankan tugas akhir ini dengan curhatan.
8. “Geng DTK” Ardhana, Gugik, Raga, Eric, Novita, Ipul, Andrys, Deni, yang saling membantu satu sama lain dikala ada masalah, meskipun saya sendiri tidak mengambil RMK DTK namun kekeluargaanya akan terbawa sampai kapanpun.
9. Suhu Djuned Fernando Djusdek, yang rela susah payah membantu saya dalam mengerjakan tugas akhir dalam hal koding dan debug program.
10. Teman-teman angkatan 2012 yang telah membantu, berbagi ilmu, menjaga kebersamaan, dan memberi motivasi kepada penulis.
11. Teman-teman PSM ITS khususnya angkatan LA 2013 teman senasib dan seperjuangan tetap bertahan di PSM ITS walau sudah tahun ke 4.
12. Serta semua pihak yang telah turut membantu penulis dalam menyelesaikan tugas akhir ini.

Penulis menyadari bahwa tugas akhir ini masih memiliki banyak kekurangan. Sehingga dengan kerendahan hati, penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan ke depannya.

Surabaya, Juni 2016

DAFTAR ISI

LEMBAR PENGESAHAN	Error! Bookmark not defined.
<i>Abstrak</i>	vii
<i>Abstract</i>	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xxi
DAFTAR KODE SUMBER	xxiii
1 BAB I PENDAHULUAN	25
1.1 Latar Belakang	25
1.2 Rumusan Masalah	26
1.3 Batasan Masalah	26
1.4 Tujuan	27
1.5 Manfaat	27
1.6 Metodologi	27
1.7 Sistematika Penulisan Laporan Tugas Akhir	28
2 BAB II TINJAUAN PUSTAKA	31
2.1 Java SWT	31
2.2 Pola Perancangan Singleton	31
2.3 Pola Perancangan Observer Pattern	32
2.4 Pola Perancangan State Pattern	32
2.5 Image Processing	33
2.5.1 Contrast Enhancement	33
2.5.2 Brightness Enhancement	33
2.5.3 Sharpness Enhancement	34
2.6 OpenCV	34
2.6.1 Core	34
2.6.2 Highgui	34
2.6.3 Mat	35
2.6.4 Imgproc	35

3 BAB III ANALISIS DAN PERANCANGAN.....	37
3.1 Analisis Perangkat Lunak.....	37
3.1.1 Deskripsi Umum Perangkat Lunak.....	37
3.1.1.1 Kebutuhan Fungsional	38
3.1.1.2 Aktor.....	39
3.1.2 Spesifikasi Kebutuhan Perangkat Lunak.....	39
3.1.2.1 Kasus Penggunaan	40
3.1.2.2 Kasus Penggunaan Membuat Proyek Baru	40
3.1.2.3 Kasus Penggunaan Menambah Halaman	41
3.1.2.4 Kasus Penggunaan Menghapus Halaman	42
3.1.2.5 Kasus Penggunaan Menambah Gambar pada Halaman	44
3.1.2.6 Kasus Penggunaan Mengubah Posisi Gambar	45
3.1.2.7 Kasus Penggunaan Mengubah ukuran Gambar	46
3.1.2.8 Kasus Penggunaan Menambah Teks pada Halaman	47
3.1.2.9 Kasus Penggunaan Melakukan Peningkatan Kualitas Citra	48
3.1.2.10 Kasus Penggunaan Menambah Bingkai pada Gambar	51
3.1.2.11 Kasus Penggunaan Melakukan Rotasi pada Gambar	52
3.1.2.12 Kasus Penggunaan Menentukan Ukuran Teks.....	54
3.1.2.13 Kasus Penggunaan Menentukan Jenis Teks.....	55
3.1.2.14 Kasus Penggunaan Menentukan Warna Teks	56
3.1.2.15 Kasus Penggunaan Memberi Efek Kontras.....	58
3.1.2.16 Kasus Penggunaan Memberi Efek Pencahayaan	59
3.1.2.17 Kasus Penggunaan Memberi Efek Ketajaman	61
3.2 Perancangan.....	62
3.2.1 Perancangan Arsitektur Sistem	62
3.2.2 Perancangan Diagram Kelas	63
3.2.2.1 Diagram Kelas MainWindow	64
3.2.2.2 Diagram Kelas NewProjectWindow	70
3.2.2.3 Diagram Kelas EditImageWindow.....	71
3.2.2.4 Diagram Kelas AddTextWindow	72

3.2.2.5	Diagram Kelas Element	73
3.2.3	Perancangan Antarmuka Pengguna.....	78
3.2.3.1	Rancangan Halaman Antarmuka MainWindow... ..	78
3.2.3.2	Rancangan Halaman Antarmuka New Project Window.....	79
3.2.3.3	Rancangan Halaman Antarmuka EditImageWindow	80
3.2.3.4	Rancangan Halaman Antarmuka AddTextWindow	82
4	BAB IV IMPLEMENTASI.....	83
4.1	Lingkungan Implementasi	83
4.2	Implementasi Pola Perancangan.....	83
4.2.1	Pola Perancangan Singleton	84
4.2.2	Pola Perancangan Observer Pattern	85
4.2.3	Pola Perancangan State Pattern	86
4.3	Implementasi Editor Grafis.....	87
4.3.1	Implementasi Antarmuka	87
4.3.1.1	Halaman MainWindow	87
4.3.1.2	Halaman NewProjectWindow.....	87
4.3.1.3	Halaman EditImageWindow	88
4.3.1.4	Halaman AddTextWindow	89
4.3.2	Implementasi Fungsi Editor	89
4.3.2.1	Fungsi Menambah Gambar	90
4.3.2.2	Fungsi Mengubah Posisi Gambar	91
4.3.2.3	Fungsi Mengubah Ukuran Gambar.....	92
4.3.2.4	Fungsi Merotasi Gambar	93
4.3.2.5	Fungsi Memanipulasi Penambahan Teks	95
4.4	Implementasi Pengolahan Citra.....	95
4.4.1	Peningkatan Kontras.....	96
4.4.2	Peningkatan Pencahayaan	97
4.4.3	Peningkatan Ketajaman	97
5	BAB V PENGUJIAN DAN EVALUASI.....	99
5.1	Lingkungan Pengujian	99
5.2	Skenario Pengujian	100
5.2.1	Kasus Pengujian Membuat Proyek Baru	100

5.2.2	Kasus Pengujian Menambah Halaman	102
5.2.3	Kasus Pengujian Menghapus Halaman	103
5.2.4	Kasus Pengujian Menambah Gambar pada Halaman.....	105
5.2.5	Kasus Pengujian Mengubah Posisi Gambar.....	107
5.2.6	Kasus Pengujian Mengubah Ukuran Gambar	109
5.2.7	Kasus Pengujian Menambah Bingkai pada Gambar	111
5.2.8	Kasus Pengujian Melakukan Rotasi pada Gambar .	112
5.2.9	Kasus Pengujian Melakukan Peningkatan Kualitas Citra	113
5.2.10	Kasus Pengujian Menambah Teks pada Halaman ..	117
5.2.11	Kasus Pengujian Menentukan Ukuran Teks.....	119
5.2.12	Kasus Pengujian Menentukan Jenis Teks.....	119
5.2.13	Kasus Pengujian Menentukan Warna Teks	121
5.3	Pengujian Pembuatan Album.....	124
5.4	Evaluasi Pengujian	126
6	BAB VI KESIMPULAN DAN SARAN	129
6.1	Kesimpulan	129
6.2	Saran	130
	DAFTAR PUSTAKA	131
	BIODATA PENULIS	133

DAFTAR GAMBAR

Gambar 2.1 Tampilan antarmuka Java SWT pada Sistem Operasi Windows Vista dan Mac OS X	32
Gambar 3.1 Garis Besar Proses Bisnis Fotokita	38
Gambar 3.2 Diagram Kasus Penggunaan.....	40
Gambar 3.3 Diagram Aktivitas Membuat Proyek Baru	41
Gambar 3.4 Diagram Aktivitas Menambah Halaman	42
Gambar 3.5 Diagram Aktivitas Menghapus Halaman.....	43
Gambar 3.6 Diagram Aktivitas Menambah Gambar pada Halaman	45
Gambar 3.7 Diagram Aktivitas Mengubah Posisi Gambar	46
Gambar 3.8 Diagram Aktivitas Mengubah Ukuran Gambar..	48
Gambar 3.9 Diagram Aktivitas Memberi Teks pada Halaman	50
Gambar 3.10 Diagram Aktivitas Melakukan Peningkatan Kualitas Citra	51
Gambar 3.11 Diagram Aktivitas Menambah Bingkai pada Gambar.....	53
Gambar 3.12 Diagram Aktivitas Melakukan Rotasi pada Gambar.....	54
Gambar 3.13 Diagram Aktivitas Menentukan Ukuran Teks..	55
Gambar 3.14 Diagram Aktivitas Menentukan Jenis Teks	57
Gambar 3.15 Diagram Aktivitas Memberi Efek Kontras	58
Gambar 3.16 Diagram Aktivitas Memberi Efek Pencahayaan	60
Gambar 3.17 Diagram Aktivitas Memberi Efek Ketajaman ..	61
Gambar 3.18 Diagram Aktivitas Menentukan Warna Teks ...	63
Gambar 3.19 Diagram Kelas MainWindow.....	64
Gambar 3.20 Diagram Kelas NewProjectWindow	70
Gambar 3.21 Diagram Kelas EditImageWindow	71
Gambar 3.22 Diagram Kelas AddTextWindow.....	72
Gambar 3.23 Diagram Kelas Element.....	74
Gambar 3.24 Rancangan Antarmuka Main Window.....	79
Gambar 3.25 Rancangan Antarmuka New Project Window ..	80

Gambar 3.26 Rancangan Antarmuka Edit Image Window	81
Gambar 3.27 Rancangan Antarmuka Add Text Window	82
Gambar 4.1 Halaman Antarmuka MainWindow.....	88
Gambar 4.2 Halaman Antarmuka New Project Window	88
Gambar 4.3 Halaman Antarmuka Edit Image Window	89
Gambar 4.4 Halaman Antarmuka Add Text Window.....	90
Gambar 5.1 Tampilan Pembuatan Proyek Baru	101
Gambar 5.2 Tampilan Akhir pembuatan Proyek Baru.....	102
Gambar 5.3 Kondisi Awal Penambahan Halaman	103
Gambar 5.4 Kondisi Akhir Penambahan Halaman	104
Gambar 5.5 Kondisi Awal Penghapusan Halaman	105
Gambar 5.6 Kondisi Akhir Penghapusan Halaman.....	105
Gambar 5.7 Kondisi Awal Penambahan Gambar.....	107
Gambar 5.8 Kondisi Akhir Penambahan Gambar	107
Gambar 5.9 Kondisi Awal Pengubahan Posisi Gambar.....	108
Gambar 5.10 Kondisi Akhir Pengubahan Posisi Gambar	109
Gambar 5.11 Kondisi Awal Pengubahan Ukuran Gambar ..	110
Gambar 5.12 Kondisi Akhir Pengubahan Ukuran Gambar .	110
Gambar 5.13 Kondisi Akhir Menambah Bingkai pada Gambar	112
Gambar 5.14 Kondisi Awal Melakukan Rotasi pada Gambar	113
Gambar 5.15 Kondisi Akhir Melakukan Rotasi pada Gambar	113
Gambar 5.16 Kondisi Citra Normal.....	115
Gambar 5.17 Kondisi Citra dengan Peningkatan Kontras ...	115
Gambar 5.18 Kondisi Citra dengan Peningkatan Pencahayaan	116
Gambar 5.19 Kondisi Citra dengan Peningkatan Ketajaman	116
Gambar 5.20 Perbedaan Citra Asli dan Citra Olahan	117
Gambar 5.21 Kondisi Awal Menambah Teks pada Halaman	118
Gambar 5.22 Kondisi Akhir Menambah Teks pada Halaman	118

Gambar 5.23 Kondisi Awal Menentukan Ukuran Teks	120
Gambar 5.24 Kondisi Akhir Menentukan Ukuran Teks	120
Gambar 5.25 Kondisi Awal Menentukan Jenis Teks	121
Gambar 5.26 Kondisi Akhir Menentukan Jenis Teks	122
Gambar 5.27 Kondisi Awal Menentukan Warna Teks.....	123
Gambar 5.28 Kondisi Setelah Warna dipilih	123
Gambar 5.29 Kondisi Akhir Menentukan Warna Teks	124
Gambar 5.30 Hasil Akhir Pengujian Halaman 1.....	125
Gambar 5.31 Hasil Akhir Pengujian Halaman 2.....	125
Gambar 5.32 Hasil Akhir Pengujian Halaman 3.....	126
Gambar 5.33 Hasil Akhir Pengujian Halaman 4.....	126

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 3.1 Kebutuhan Fungsional Sistem	39
Tabel 3.2 Kasus Penggunaan Menghapus Halaman	43
Tabel 3.3 Kasus Penggunaan Menambah Gambar pada Halaman	44
Tabel 3.4 Kasus Penggunaan Mengubah Posisi Gambar	45
Tabel 3.5 Kasus Penggunaan Mengubah Ukuran Gambar	47
Tabel 3.6 Kasus Penggunaan Menambah Teks pada Halaman	49
Tabel 3.7 Kasus Penggunaan Melakukan Peningkatan Kualitas Citra	49
Tabel 3.8 Kasus Penggunaan Menambah Bingkai pada Gambar	51
Tabel 3.9 Kasus Penggunaan Melakukan rotasi pada gambar	52
Tabel 3.10 Kasus Penggunaan Menentukan Ukuran Teks	54
Tabel 3.11 Kasus Penggunaan Menentukan Jenis Teks	56
Tabel 3.12 Kasus Penggunaan Menentukan Warna Teks	57
Tabel 3.13 Kasus Penggunaan Memberi Efek Kontras	59
Tabel 3.14 Kasus Penggunaan Memberi Efek Pencahayaan	60
Tabel 3.15 Kasus Penggunaan Memberi Efek Ketajaman	62
Tabel 3.16 Fungsi Utama Kelas ATool	65
Tabel 3.17 Fungsi Utama Kelas IExplorer	66
Tabel 3.18 Fungsi Utama Kelas IListImage	67
Tabel 3.19 Fungsi Utama Kelas IImageViewer	67
Tabel 3.20 Fungsi Utama Kelas IInitTab	68
Tabel 3.21 Fungsi Utama Kelas IPage	68
Tabel 3.22 Fungsi Utama Kelas AElement	70
Tabel 3.23 Fungsi Utama Kelas AenhancementOption	71
Tabel 3.24 Fungsi Utama Kelas IimageEditor	72
Tabel 3.25 Fungsi Utama kelas ControlPoint	75
Tabel 3.26 Fungsi Utama Kelas Aeditable	75
Tabel 3.27 Fungsi Utama Kelas TwoDimensional	76
Tabel 3.28 Fungsi Utama Kelas Picture	77
Tabel 3.29 Fungsi Utama Kelas Frame	77

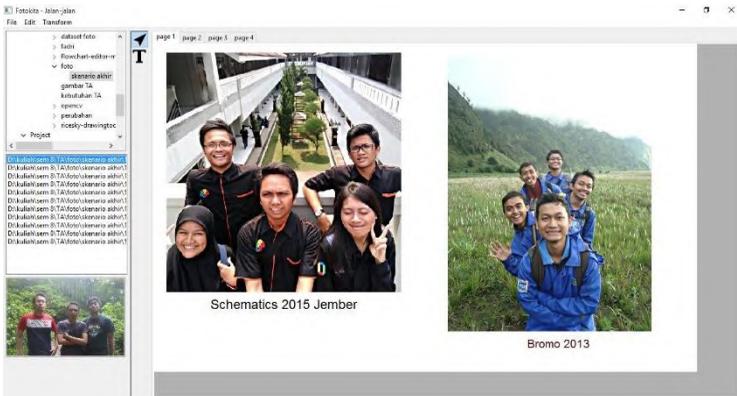
Tabel 3.30 Penjelasan Antarmuka MainWindow.....	78
Tabel 3.31 Penjelasan Antarmuka New Project Window	80
Tabel 3.32 Penjelasan Antarmuka Edit Image Window	81
Tabel 3.33 Penjelasan Antarmuka Add Text Window.....	82
Tabel 5.1 Spesifikasi Sistem Pengujian.....	99
Tabel 5.2 Daftar Testing Fitur Aplikasi.....	100
Tabel 5.3 Kasus Uji Membuat Proyek Baru.....	101
Tabel 5.4 Kasus Uji Menambah Halaman.....	102
Tabel 5.5 Kasus Uji Menghapus Halaman	104
Tabel 5.6 Kasus Uji Menambah Gambar pada Halaman	106
Tabel 5.7 Kasus Uji Mengubah Posisi Gambar.....	108
Tabel 5.8 Kasus Uji Mengubah Ukuran Gambar	109
Tabel 5.9 Kasus Uji Menambah Bingkai pada Gambar.....	111
Tabel 5.10 Kasus Uji Melakukan Rotasi pada Gambar	112
Tabel 5.11 Kasus Uji Melakukan Peningkatan Kualitas Citra	114
Tabel 5.12 Kasus Uji Menambah Teks pada Halaman	117
Tabel 5.13 Kasus Uji Menentukan Ukuran Teks	119
Tabel 5.14 Kasus Uji Menentukan Jenis Teks	120
Tabel 5.15 Kasus Uji Menentukan Warna Teks.....	122
Tabel 5.16 Biodata Pengguna	124
Tabel 5.17 Tabel Evaluasi.....	127

DAFTAR KODE SUMBER

Kode Sumber 4.1 Constructor MainWindow	84
Kode Sumber 4.2 Constructor NormalState.....	84
Kode Sumber 4.3 Constructor SizingState.....	85
Kode Sumber 4.4 Implementasi Observer Pattern Kelas Frame	86
Kode Sumber 4.5 Fungsi draw Kelas SizingState	86
Kode Sumber 4.6 Fungsi draw Kelas AElement	86
Kode Sumber 4.7 Fungsi select Kelas AElement	87
Kode Sumber 4.8 Kode Sumber Penambahan Gambar	91
Kode Sumber 4.9 Fungsi renderNormal Kelas Picture.....	91
Kode Sumber 4.10 Fungsi drag TwoDimensional.....	91
Kode Sumber 4.11 Kode Sumber Mengeksekusi Fungsi drag	92
Kode Sumber 4.12 Kode Sumber Mengubah Ukuran Gambar	93
Kode Sumber 4.13 Kode Sumber Merotasi Gambar	94
Kode Sumber 4.14 Kode Sumber Instansiasi Kelas Paragraph	95
Kode Sumber 4.15 Kode Sumber Peningkatan Kontras.....	96
Kode Sumber 4.16 Kode Sumber Peningkatan Pencahayaan	97
Kode Sumber 4.17 Kode Sumber Peningkatan Ketajaman	98

[Halaman ini sengaja dikosongkan]

Pekerjaan	Mahasiswa
Pendidikan 1. SD 2. SMP 3. SMA 4. Kuliah	1. SDN Kedungsari 8 2. SMPN 2 Jember 3. SMAN 1 Jember 4. Institut Teknologi Sepuluh Nopember (Teknik Informatika)



Gambar 5.30 Hasil Akhir Pengujian Halaman 1



Gambar 5.31 Hasil Akhir Pengujian Halaman 2



Gambar 5.32 Hasil Akhir Pengujian Halaman 3



Gambar 5.33 Hasil Akhir Pengujian Halaman 4

5.4 Evaluasi Pengujian

Pada bab ini akan dijelaskan bagaimana hasil dari pengujian aplikasi dan akan disertakan dalam bentuk tabel. Tabel evaluasi dan pengujian dapat dilihat pada Tabel 5.17

Tabel 5.17 Tabel Evaluasi

Kode	Kasus Penggunaan	Hasil Pengujian
UJ-001	Membuat proyek baru	Berhasil
UJ-002	Menambah halaman	Berhasil
UJ-003	Menghapus halaman	Berhasil
UJ-004	Menambah gambar pada halaman	Berhasil
UJ-005	Mengubah posisi gambar	Berhasil
UJ-006	Mengubah ukuran gambar	Berhasil
UJ-007	Menambah bingkai pada gambar	Berhasil
UJ-008	Melakukan rotasi pada gambar	Berhasil
UJ-009	Melakukan peningkatan kualitas citra	Berhasil
UJ-010	Menambah teks pada halaman	Berhasil
UJ-011	Menentukan ukuran teks	Berhasil
UJ-012	Menentukan jenis teks	Berhasil
UJ-013	Menentukan warna teks	Berhasil

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

1.1 Latar Belakang

Setiap orang tentu ingin mengabadikan setiap momen terindah dalam hidupnya dalam bentuk gambar. Mulai dari kalangan muda hingga dewasa, tentunya ingin membungkus momen tersebut ke dalam suatu wadah agar kelak dapat membuka kenangan tersebut. Hal ini memunculkan kebutuhan baru masyarakat mengenai sebuah wadah yang dapat menampung gambar serta mempercantik gambar tersebut. Selain dapat menyimpan foto, pengguna juga dapat mengatur banyaknya foto tersebut ke dalam sebuah buku foto, di mana seluruh foto-foto tersebut dapat dimasukkan ke dalam frame dan diatur posisinya. Kebutuhan lain dari masyarakat, terutama masyarakat Indonesia adalah bentuk fisik dari foto hasil jepretan yang telah disimpan.

Semakin maraknya aktivitas pengabdian momen, dan diiringi dengan kebutuhan masyarakat, tentunya harus ada teknologi yang dapat membantu memudahkan proses tersebut. Beberapa solusi yang pernah ada belum sesuai, dikarenakan mayoritas solusi tersebut ada dalam bentuk online, sedangkan jaringan di Indonesia belum bisa dikatakan mampu untuk melakukan semua kegiatan dalam sistem online. Salah satu contoh aplikasi yang sudah ada adalah photobookindonesia.com. Website tersebut menyediakan layanan untuk desain album foto berbasis online yang menyebabkan apabila kita tidak memiliki koneksi internet yang stabil dan baik akan menjadi semakin sulit untuk mendesain.

Oleh karena itu perlu adanya sistem yang memudahkan masyarakat untuk berkreasi menyusun hasil potret mereka secara standalone, agar pengerjaan tersebut berjalan stabil. Selain itu tentu perlu adanya tempat untuk memamerkan hasil

karya masyarakat secara luas, sehingga perlu adanya sistem yang juga mendukung pertunjukan karya tersebut secara online.

Dari diagram arsitektur sistem aplikasi Fotokita tersebut, bagian yang diimplementasi dalam tugas akhir ini adalah perancangan dan implementasi aplikasi desktop untuk membuat album foto digital.

Fitur-fitur editing yang akan diimplementasikan dalam aplikasi Fotokita adalah *rotate*, *frame*, efek (*image enhancement*), *scalling*.

Hasil yang diharapkan adalah aplikasi dapat mendesain album foto yang berisi gambar ataupun tulisan, sesuai dengan keinginan pengguna.

1.2 Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana memberi layout pada Fotokita agar layout dapat dinamis sesuai dengan keinginan pengguna?
2. Bagaimana membuat kualitas dari citra yang dipilih dapat lebih meningkat?
3. Bagaimana agar foto yang ingin kita gunakan memiliki ukuran yang sesuai dengan editor (tidak berlebihan)?

1.3 Batasan Masalah

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, yaitu sebagai berikut:

1. Hanya dapat menambah gambar (foto) dan text pada halaman.
2. Rotasi yang disediakan kelipatan 90 derajat.
3. Format input gambar yang diterima hanya .JPEG, .JPG, dan .PNG.
4. Aplikasi berbasis desktop

1.4 Tujuan

Tujuan pembuatan aplikasi ini adalah untuk memudahkan pembuatan desain album foto yang sudah cukup populer, sehingga hanya cukup satu aplikasi saja yang perlu digunakan.

1.5 Manfaat

Manfaat dari hasil tugas akhir ini adalah salah satunya efisiensi penggunaan aplikasi, kemudahan dalam perancangan album foto.

1.6 Metodologi

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut:

1. Penyusunan proposal tugas akhir.

Proposal tugas akhir ini berisi tentang deskripsi pendahuluan dari tugas akhir yang akan dibuat. Pendahuluan ini terdiri atas hal yang menjadi latar belakang diajukannya usulan tugas akhir, rumusan masalah yang diangkat, batasan masalah untuk tugas akhir, tujuan dari pembuatan tugas akhir, dan manfaat dari hasil pembuatan tugas akhir. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung pembuatan tugas akhir. Sub bab metodologi berisi penjelasan mengenai tahapan penyusunan tugas akhir mulai dari penyusunan peroposal hingga penyusunan buku tugas akhir. Terdapat pula sub bab jadwal kegiatan yang menjelaskan jadwal pengerjaan tugas akhir.

2. Studi literatur

Pada tahap ini dilakukan studi terhadap literatur yang akan banyak digunakan dalam pembuatan aplikasi.

Literatur tersebut adalah *Framework* Java SWT untuk perancangan tampilan antarmuka pengguna. Selain itu juga penggunaan library OpenCV untuk melakukan pengolahan citra.

3. Analisis dan desain perangkat lunak
Tahap ini meliputi perumusan kebutuhan fungsional, kasus penggunaan, diagram aktivitas, rancangan antarmuka pengguna dan diagram kelas.
4. Implementasi perangkat lunak
Implementasi merupakan tahap membangun rancangan program yang telah dibuat. Pada tahapan ini merealisasikan rancangan yang terdapat pada tahapan sebelumnya, sehingga menjadi sebuah program yang sesuai dengan apa yang telah direncanakan.
5. Pengujian dan evaluasi
Proses pengujian akan dilakukan dengan membuat suatu desain album foto sesuai dengan layout yang diberikan. Apabila semua fitur yang disediakan berhasil dijalankan maka pengujian dikatakan berhasil.
6. Penyusunan buku tugas akhir.
Pada tahapan ini disusun buku yang memuat dokumentasi mengenai pembuatan serta hasil dari implementasi perangkat lunak yang telah dibuat.

1.7 Sistematika Penulisan Laporan Tugas Akhir

Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku tugas akhir terdiri atas beberapa bagian seperti berikut ini.

Bab I Pendahuluan

Bab yang berisi mengenai latar belakang, tujuan, dan manfaat dari pembuatan tugas akhir. Selain itu permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penulisan juga merupakan bagian dari bab ini.

Bab II Tinjauan Pustaka

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

Bab III Perancangan Perangkat Lunak

Bab ini berisi tentang desain sistem, rancangan basis data, diagram kasus penggunaan, diagram aktivitas dan rancangan antarmuka pengguna.

Bab IV Implementasi

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa *code* yang digunakan untuk proses implementasi.

Bab V Pengujian dan Evaluasi

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

Bab VI Kesimpulan dan Saran

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

[Halaman ini sengaja dikosongkan]

BAB II

TINJAUAN PUSTAKA

Bab ini berisi penjelasan teori-teori dan pustaka yang berkaitan dengan rancang bangun aplikasi editor album foto digital yang diajukan pada pengimplementasian program. Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap aplikasi yang dibuat dan berguna sebagai penunjang dalam pengembangan perangkat lunak.

Pada masing-masing poin akan dijelaskan bagaimana teori atau pustaka digunakan pada pengembangan aplikasi. Fungsi yang digunakan juga akan dijelaskan apabila pustaka memiliki fungsi tertentu yang digunakan untuk pengembangan aplikasi.

2.1 Java SWT

Standard Widget Toolkit atau yang biasa disingkat SWT adalah pustaka perangkat lunak berbasis GUI (Graphical User Interface) untuk Java. Perbedaan dengan pustaka GUI yang lain adalah SWT berjalan lebih cepat, menggunakan memory dengan efektif, dan tampilan antarmuka yang menyesuaikan sistem operasi dimana aplikasi itu dijalankan [1].

2.2 Pola Perancangan Singleton

Singleton adalah sebuah pattern yang membatasi penggunaan class menjadi sebuah *instance* saja pada sebuah aplikasi. Biasanya ada sebuah class pada aplikasi yang pada desain hanya dibutuhkan sekali [2]. Pembatasan ini sebenarnya bisa dibuat sebagai sebuah objek statis bersifat global.



Gambar 2.1 Tampilan antarmuka Java SWT pada Sistem Operasi Windows Vista dan Mac OS X

2.3 Pola Perancangan Observer Pattern

Pola perancangan *observer pattern* adalah pola perancangan dimana sebuah objek mendaftarkan diri pada objek lain sehingga objek terdaftar dapat diketahui kondisinya. Pada pola ini ada dua buah status dari masing-masing objek yang berperan yaitu *observer* dan *observable*. *Observer* adalah objek yang mendaftarkan diri ke *observable* sehingga *observable* dapat mengetahui kondisi *observer*. Selain dapat mengetahui kondisi *observer*, *observable* juga mampu memanipulasi atribut apa saja pada *observer*. Oleh karena itu memungkinkan apabila tiap ada perubahan pada *observable*, *observer* akan menjalankan sebuah fungsi yang ditentukan. [3]

2.4 Pola Perancangan State Pattern

Pola perancangan *state pattern* berguna untuk menentukan perilaku dari objek pada kondisi tertentu. Kondisi yang diberikan pada aplikasi adalah kondisi saat objek terpilih

maka perilaku yang diberikan juga akan berbeda saat objek tidak terpilih [4].

State pattern muncul akibat adanya kondisi-kondisi yang muncul pada saat pembuatan perangkat lunak. Pattern ini mengizinkan state *transition logic* untuk disatukan dengan sebuah *state object* daripada berada dalam kondisional.

2.5 Image Processing

Pengolahan citra atau *image processing* adalah salah satu cabang ilmu informatika yang mempelajari transformasi suatu citra menjadi citra yang lain melalui teknik dan algoritma pengolahan citra tertentu [5].

Jenis *image processing* yang digunakan dalam aplikasi adalah peningkatan kualitas citra (*image enhancement*). Peningkatan kualitas citra yang disediakan ada tiga macam yaitu peningkatan kontras, peningkatan pencahayaan, dan peningkatan ketajaman.

2.5.1 Contrast Enhancement

Peningkatan kontras merupakan salah satu jenis *image enhancement* yang bertujuan untuk meningkatkan warna sebuah gambar. Warna yang kurang begitu jelas pada gambar dapat ditingkatkan dengan cara ini. Metode yang digunakan untuk meningkatkan kontras adalah *histogram equalization* [6].

2.5.2 Brightness Enhancement

Peningkatan pencahayaan merupakan salah satu jenis *image enhancement* yang bertujuan untuk meningkatkan kecerahan gambar apabila gambar dirasa kurang terang. Kondisi apabila gambar diambil dalam ruangan yang cenderung redup dapat ditingkatkan pencahayaannya dengan cara ini.

2.5.3 Sharpness Enhancement

Peningkatan ketajaman merupakan salah satu jenis *image enhancement* yang bertujuan untuk memperdetail gambar sehingga batas tiap objek pada gambar terlihat lebih jelas. Metode yang digunakan adalah *gaussian blur*.

2.6 OpenCV

OpenCV adalah salah satu pustaka untuk mempermudah pengolahan citra dengan fungsi-fungsi yang sering digunakan untuk pengolahan. OpenCV dapat berjalan dalam bahasa C, C++, Java, dan Python. OpenCV dirancang untuk efisiensi komputasional dan berfokus pada aplikasi *realtime* [6].

Versi dari OpenCV yang digunakan pada aplikasi adalah OpenCV 2.4.12 sedangkan versi yang terbaru adalah versi 2.4.13. Implementasi penggunaan OpenCV pada pemrograman Java menggunakan kelas *Core*, *Highgui*, *Mat*, dan *Imgproc*.

2.6.1 Core

Kelas *Core* merupakan kelas statis untuk menampung berbagai macam fungsi dasar pada OpenCV. Kelas *Core* ada karena untuk mengakses sebuah fungsi pada bahasa pemrograman Java harus melalui sebuah kelas penampung fungsi, beda dengan bahasa C yang dapat mengakses langsung fungsi tanpa harus ada kelas penampung. Fungsi yang digunakan pada kelas ini adalah *split*, *merge*, dan *addWeighted*,

2.6.2 Highgui

Kelas *Highgui* merupakan kelas membaca *input* file gambar yang akan diolah. Kelas ini merupakan kelas statis yang juga digunakan untuk melakukan *input* dan *output* file yang

diolah. Fungsi yang digunakan pada kelas ini adalah *imread* dan *imwrite*.

2.6.3 Mat

Kelas Mat merupakan kelas untuk mengolah gambar yang dibaca oleh kelas Highgui. Semua gambar yang dibaca akan diolah sebagai kelas Mat sehingga pengolahan citra yang terjadi adalah pengolahan kelas ini yang menerima *input* file gambar dari kelas Highgui.

2.6.4 Imgproc

Kelas Imgproc merupakan kelas statis yang digunakan khusus untuk pengolahan kontras gambar. Kelas ini memiliki fungsi *equalizeHist* yang bertujuan untuk mengolah satu *channel* warna dari gambar yang dibaca sehingga gambar menjadi lebih kontras.

[Halaman ini sengaja dikosongkan]

BAB III

ANALISIS DAN PERANCANGAN

Pada bab ini akan membahas analisis perancangan aplikasi desktop untuk mempermudah tahap implementasi aplikasi. Perancangan sistem ini akan direpresentasikan dengan menggunakan diagram *Unified Modelling Language* (selanjutnya disebut UML). Dalam UML nantinya akan terlihat dengan jelas bagaimana struktur rancangan aplikasi dalam diagram-diagram yang dijelaskan.

3.1 Analisis Perangkat Lunak

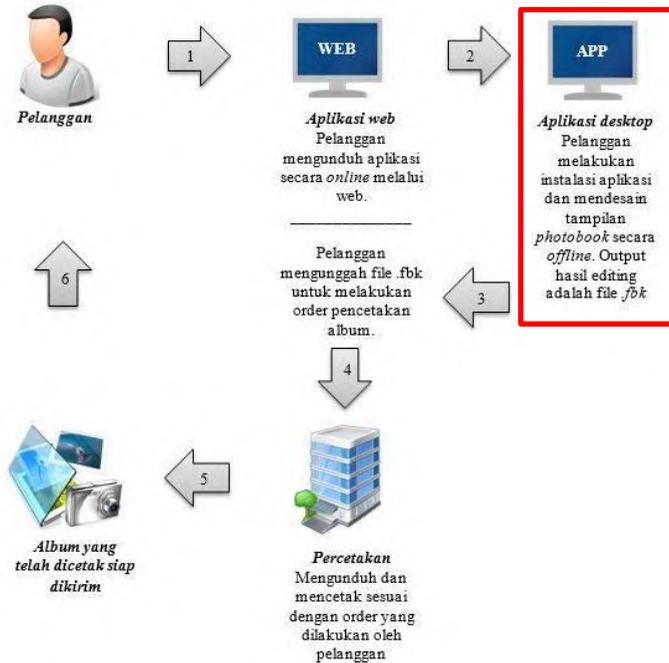
Pada subbab ini akan dibahas mengenai analisis perancangan dan arsitektur perangkat lunak aplikasi editor album foto digital ‘Fotokita’ berbasis desktop.

3.1.1 Deskripsi Umum Perangkat Lunak

Pada tugas akhir ini akan dilakukan pengembangan aplikasi editor album foto digital berbasis desktop untuk memudahkan perancangan album foto sebelum nantinya akan dicetak. Foto yang akan digunakan juga dapat diproses melalui peningkatan kualitas citra agar gambar yang diinginkan terlihat lebih baik.

Fitur yang tersedia pada aplikasi diantaranya menambahkan gambar, menambahkan teks, menambahkan bingkai pada gambar, memberi efek pada gambar, merotasi gambar, mengubah letak gambar, dan mengubah ukuran gambar.

Bagian yang juga dibahas pada subbab ini adalah kebutuhan fungsional aplikasi dan aktor yang terlibat dalam penggunaan perangkat lunak. Secara keseluruhan garis besar rancangan proses bisnis dari aplikasi Fotokita dapat dilihat pada



Gambar 3.1 Garis Besar Proses Bisnis Fotokita

Pada tugas akhir ini yang dibahas adalah proses implementasi pembuatan aplikasi desktop, sedangkan bagian lain dibahas pada tugas akhir lain.

3.1.1.1 Kebutuhan Fungsional

Kebutuhan fungsional dari aplikasi akan dijelaskan pada subbab ini. Penyajian kebutuhan fungsional aplikasi akan disajikan dalam bentuk tabel yang dapat dilihat pada Tabel 3.1.

Tabel 3.1 Kebutuhan Fungsional Sistem

No.	Kebutuhan Fungsional	Deskripsi
1	Membuat proyek baru	Pengguna dapat membuat rancangan album foto dari awal
2	Menambahkan halaman	Pengguna dapat menambahkan halaman sesuai dengan yang diinginkan
3	Mengatur Layout	Pengguna dapat meletakkan gambar yang diinginkan pada posisi yang disediakan
4	Menambah gambar pada masing-masing halaman	Pengguna dapat menambahkan gambar pada halaman yang diinginkan
5	Mengatur bingkai pada gambar	Pengguna dapat memberikan efek bingkai pada gambar yang dipilih
7	Menambahkan teks pada halaman	Pengguna dapat menambahkan teks pada halaman
8	Menambahkan efek	Pengguna dapat meningkatkan kualitas citra dari citra yang dipilih
9	Merotasi gambar	Pengguna dapat merotasi gambar yang dipilih pada halaman

3.1.1.2 Aktor

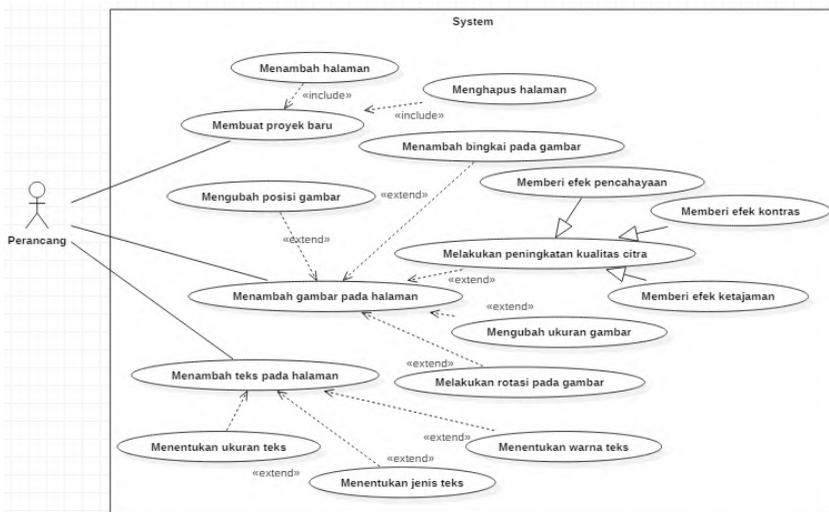
Pengertian aktor adalah pihak-pihak, baik manusia maupun aktor/perangkat lunak lain yang terlibat dan berinteraksi langsung dengan aktor. Dalam tugas akhir ini hanya memiliki 1 aktor yaitu pengguna.

3.1.2 Spesifikasi Kebutuhan Perangkat Lunak

Sesuai dengan uraian mengenai cakupan perangkat lunak yang dibangun, dibutuhkan adanya spesifikasi perangkat lunak agar dapat memberikan solusi dari permasalahan yang diberikan dan dapat bekerja dengan baik dalam mengakomodasi kebutuhan. Diharapkan dengan adanya spesifikasi ini dapat menyesuaikan kebutuhan-kebutuhan pengguna.

3.1.2.1 Kasus Penggunaan

Bagian ini menjelaskan secara rinci kasus penggunaan yang terdapat pada perangkat lunak. Selain itu, terdapat juga spesifikasi kasus penggunaan, diagram aktivitas dan diagram urutan untuk tiap-tiap kasus penggunaan. Sesuai dengan penjelasan kebutuhan fungsional, maka perangkat lunak memiliki 13 kasus penggunaan yang dapat dilihat pada Gambar 3.2.



Gambar 3.2 Diagram Kasus Penggunaan

3.1.2.2 Kasus Penggunaan Membuat Proyek Baru

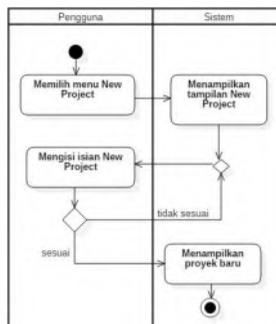
Pada kasus penggunaan ini, pengguna dapat membuat proyek baru mulai dari awal. Proyek baru tersebut berisi nama penulis, jumlah halaman, ukuran kertas, dan judul. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. dan diagram aktivitas pada Gambar 3.3.

Tabel 3.2 Kasus Penggunaan Membuat Proyek Baru

Komponen	Deskripsi
Nama	Membuat Proyek Baru
Nomor	UC-001
Deskripsi	Kasus penggunaan ini digunakan untuk membuat proyek album foto mulai dari awal.
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Proyek belum dibuat
Kondisi Akhir	Proyek sudah dibuat
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih menu <i>New Project</i>. 2. Sistem menampilkan tampilan untuk membuat proyek baru. 3. Pengguna memasukkan data nama penulis, halaman, judul, dan ukuran kertas. 4. Sistem membuat proyek baru pada aplikasi.

3.1.2.3 Kasus Penggunaan Menambah Halaman

Pada kasus penggunaan ini, apabila jumlah halaman pada proyek apabila dirasa kurang. Halaman yang ditambahkan adalah halaman paling belakang. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.3 dan diagram aktivitas terdapat pada Gambar 3.4.

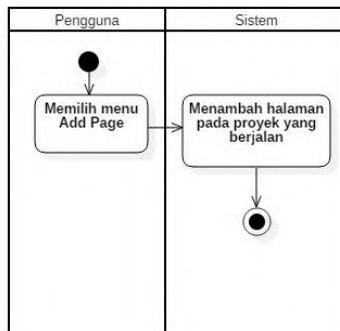
**Gambar 3.3 Diagram Aktivitas Membuat Proyek Baru**

Tabel 3.3 Kasus Penggunaan Menambah Halaman

Komponen	Deskripsi
Nama	Menambah Halaman
Nomor	UC-002
Deskripsi	Pengguna dapat menambahkan halaman pada proyek yang sedang terbuka
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Halaman belum ditambahkan
Kondisi Akhir	Halaman sudah ditambahkan
Prekondisi	Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih menu Add Page. 2. Sistem menambahkan halaman pada proyek yang sedang berjalan.

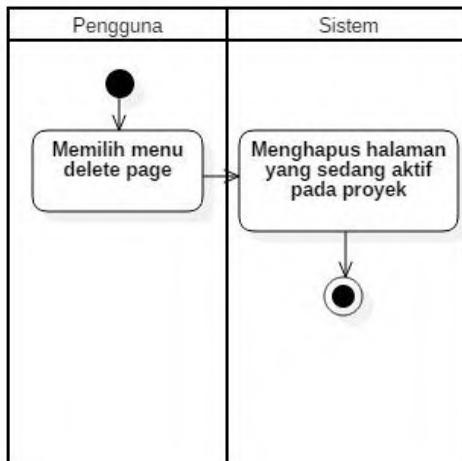
3.1.2.4 Kasus Penggunaan Menghapus Halaman

Pada kasus penggunaan ini, pengguna dapat menghapus halaman yang dipilih jika dirasa perlu untuk dihapus. Jika halaman dihapus maka semua elemen yang ada pada halamana juga dihapus. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.2 dan diagram aktivitas terdapat pada Gambar 3.5.

**Gambar 3.4 Diagram Aktivitas Menambah Halaman**

Tabel 3.2 Kasus Penggunaan Menghapus Halaman

Komponen	Deskripsi
Nama	Menambah Halaman
Nomor	UC-003
Deskripsi	Pengguna dapat menghapus halaman pada proyek yang sedang terbuka
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Halaman belum dihapus
Kondisi Akhir	Halaman sudah dihapus
Prekondisi	Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> Pengguna memilih menu Delete Page. Sistem menghapus halaman pada proyek yang sedang berjalan.

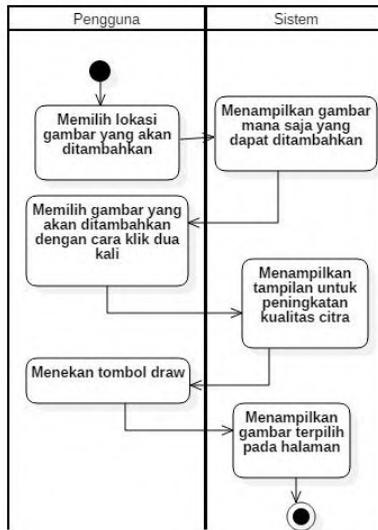
**Gambar 3.5 Diagram Aktivitas Menghapus Halaman**

3.1.2.5 Kasus Penggunaan Menambah Gambar pada Halaman

Pada kasus penggunaan ini, pengguna dapat menambahkan gambar yang diinginkan pada halaman terpilih. Gambar yang bisa dipilih hanya berformat JPG, JPEG, dan PNG. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.3 dan diagram aktivitas terdapat pada Gambar 3.6

Tabel 3.3 Kasus Penggunaan Menambah Gambar pada Halaman

Komponen	Deskripsi
Nama	Menambah gambar pada halaman
Nomor	UC-004
Deskripsi	Pengguna dapat menambah gambar pada halaman
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Gambar belum ditambahkan
Kondisi Akhir	Gambar sudah ditambahkan
Prekondisi	Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih lokasi dari gambar pada kiri antarmuka pengguna. 2. Sistem menampilkan gambar mana saja yang terdapat pada lokasi terpilih 3. Pengguna melakukan <i>double click</i> pada gambar terpilih 4. Sistem menampilkan tampilan <i>editWindow</i> 5. Pengguna menekan tombol <i>draw</i> 6. Sistem menambahkan gambar pada halaman terpilih



Gambar 3.6 Diagram Aktivitas Menambah Gambar pada Halaman

3.1.2.6 Kasus Penggunaan Mengubah Posisi Gambar

Pada kasus penggunaan ini, pengguna dapat mengubah posisi gambar yang sudah ditambahkan ke halaman yang sedang aktif. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.4 dan Gambar 3.7.

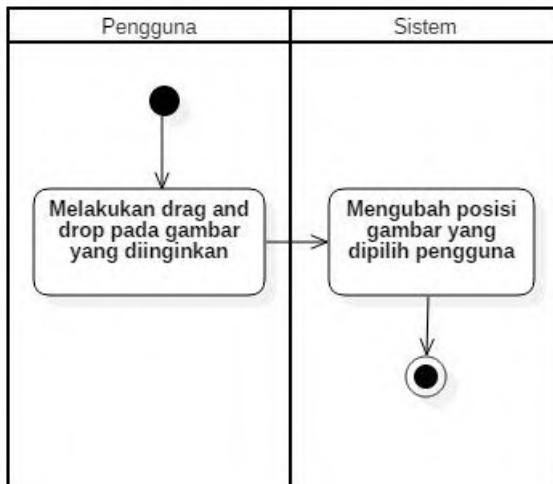
Tabel 3.4 Kasus Penggunaan Mengubah Posisi Gambar

Komponen	Deskripsi
Nama	Mengubah posisi gambar
Nomor	UC-005
Deskripsi	Pengguna dapat mengubah posisi gambar pada halaman

Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	-
Kondisi Akhir	Gambar berubah posisi
Prekondisi	- Harus ada proyek yang sedang berjalan - Harus ada gambar yang dimuat pada halaman
Alur Normal	1. Pengguna melakukan <i>drag and drop</i> pada gambar yang dipilih pada halaman 2. Gambar berubah posisi

3.1.2.7 Kasus Penggunaan Mengubah ukuran Gambar

Pada kasus penggunaan ini, pengguna dapat mengubah ukuran gambar yang sudah ditambahkan ke halaman yang sedang aktif. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.5 dan diagram aktivitas pada Gambar 3.8.



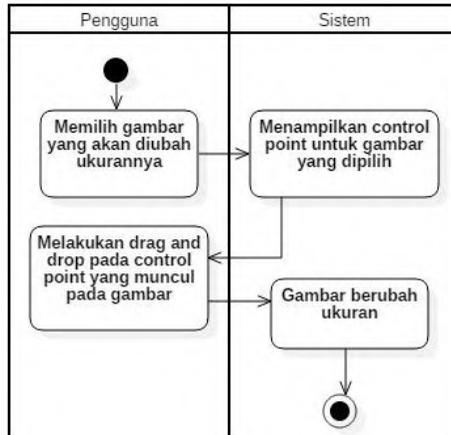
Gambar 3.7 Diagram Aktivitas Mengubah Posisi Gambar

Tabel 3.5 Kasus Penggunaan Mengubah Ukuran Gambar

Komponen	Deskripsi
Nama	Mengubah ukuran gambar
Nomor	UC-006
Deskripsi	Pengguna dapat mengubah ukuran gambar
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Ukuran gambar masih dalam ukuran awal
Kondisi Akhir	Ukuran gambar berubah
Prekondisi	- Harus ada proyek yang sedang berjalan - Harus ada gambar yang dimuat pada halaman
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih gambar yang akan diubah ukurannya 2. Sistem menampilkan <i>control point</i> untuk gambar yang dipilih 3. Pengguna melakukan <i>drag and drop</i> pada <i>control point</i> yang muncul pada gambar 4. Gambar berubah ukuran

3.1.2.8 Kasus Penggunaan Menambah Teks pada Halaman

Pada kasus penggunaan ini, pengguna dapat menambahkan teks pada halaman yang sedang aktif apabila diinginkan. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.6 dan diagram aktivitas terdapat pada Gambar 3.9.



Gambar 3.8 Diagram Aktivitas Mengubah Ukuran Gambar

3.1.2.9 Kasus Penggunaan Melakukan Peningkatan Kualitas Citra

Pada kasus penggunaan ini, pengguna dapat melakukan peningkatan kualitas citra untuk citra yang ingin ditambahkan ke dalam halaman yang sedang aktif. Peningkatan kualitas citra ini tidak harus dilakukan apabila dirasa tidak dibutuhkan oleh pengguna. Peningkatan kualitas citra yang disediakan adalah peningkatan kontras, peningkatan pencahayaan, serta peningkatan ketajaman citra. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.7 dan diagram aktivitas terdapat pada Gambar 3.10.

Tabel 3.6 Kasus Penggunaan Menambah Teks pada Halaman

Komponen	Deskripsi
Nama	Menambah teks pada halaman
Nomor	UC-007
Deskripsi	Pengguna dapat menambah teks pada halaman
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Teks belum ditambahkan ke halaman
Kondisi Akhir	Ukuran gambar berubah
Prekondisi	- Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih <i>text tool</i> pada <i>toolbox</i> 2. Pengguna memilih tempat akan diletakkannya teks. 3. Sistem menampilkan tampilan untuk memberi teks pada halaman 4. Pengguna mengetikkan teks yang ingin dimasukkan ke halaman 5. Teks ditambahkan ke halaman

Tabel 3.7 Kasus Penggunaan Melakukan Peningkatan Kualitas Citra

Komponen	Deskripsi
Nama	Melakukan peningkatan kualitas citra
Nomor	UC-008
Deskripsi	Pengguna dapat meningkatkan kualitas dari citra yang dipilih bila perlu
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Citra awal belum ditingkatkan kualitasnya
Kondisi Akhir	Citra sudah ditingkatkan kualitasnya
Prekondisi	- Harus ada proyek yang sedang berjalan

Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih lokasi citra yang ingin ditingkatkan kualitasnya 2. Sistem menampilkan citra apa saja yang dapat dipilih 3. Pengguna melakukan <i>double click</i> pada citra yang dipilih 4. Sistem menampilkan tampilan <i>edit window</i> 5. Pengguna memilih tipe peningkatan kualitas citra yang disediakan 6. Sistem melakukan peningkatan kualitas citra sesuai dengan yang dipilih pengguna 7. Pengguna menekan tombol <i>draw</i> 8. Citra hasil olahan berhasil ditambahkan ke halaman
--------------------	---



Gambar 3.9 Diagram Aktivitas Memberi Teks pada Halaman

3.1.2.10 Kasus Penggunaan Menambah Bingkai pada Gambar

Pada kasus penggunaan ini pengguna dapat memberi bingkai pada gambar yang dipilih sebelum gambar ditambahkan pada halaman yang sedang aktif. Rincian kasus penggunaan tersebut dapat dilihat pada Gambar 3.11 dan Tabel 3.8.



Gambar 3.10 Diagram Aktivitas Melakukan Peningkatan Kualitas Citra

Tabel 3.8 Kasus Penggunaan Menambah Bingkai pada Gambar

Komponen	Deskripsi
Nama	Menambahkan bingkai pada gambar
Nomor	UC-009

Deskripsi	Pengguna dapat menambahkan bingkai pada gambar yang dipilih
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	-
Kondisi Akhir	Gambar dimuat dalam halaman dengan bingkai yang dipilih
Prekondisi	- Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih lokasi gambar yang ingin ditambahkan 2. Sistem menampilkan gambar apa saja yang dapat dipilih 3. Pengguna melakukan <i>double click</i> pada gambar yang dipilih 4. Sistem menampilkan tampilan <i>edit window</i> 5. Pengguna memilih bingkai yang akan ditambahkan 6. Pengguna menekan tombol <i>draw</i> 7. Gambar berhasil ditambahkan bersama bingkai yang dipilih

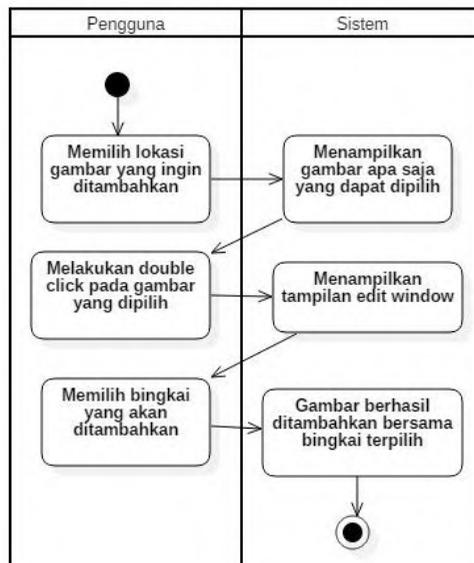
3.1.2.11 Kasus Penggunaan Melakukan Rotasi pada Gambar

Pada kasus penggunaan ini pengguna dapat melakukan rotasi kelipatan 90° pada gambar yang dipilih. Arah rotasi yang disediakan adalah searah jarum jam dan berlawanan jarum jam. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.9 dan Gambar 3.12.

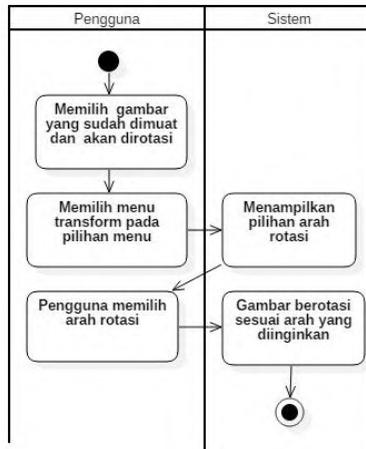
Tabel 3.9 Kasus Penggunaan Melakukan rotasi pada gambar

Komponen	Deskripsi
Nama	Melakukan rotasi pada gambar
Nomor	UC-010

Deskripsi	Pengguna dapat menambahkan bingkai pada gambar yang dipilih
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Gambar belum dirotasi
Kondisi Akhir	Gambar sudah dirotasi
Prekondisi	- Harus ada proyek yang sedang berjalan - Harus ada gambar yang sudah dimuat
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih gambar yang sudah dimuat dan akan dirotasi 2. Pengguna memilih menu <i>transform</i> pada pilihan menu 3. Sistem menampilkan pilihan arah rotasi 4. Pengguna memilih arah rotasi 5. Gambar berotasi sesuai arah yang diinginkan



Gambar 3.11 Diagram Aktivitas Menambah Bingkai pada Gambar



Gambar 3.12 Diagram Aktivitas Melakukan Rotasi pada Gambar

3.1.2.12 Kasus Penggunaan Menentukan Ukuran Teks

Pada kasus penggunaan ini pengguna dapat menentukan ukuran teks yang akan ditambahkan ke dalam halaman yang sedang aktif. Rincian kasus penggunaan tersebut dapat dilihat pada Gambar 3.13 dan Tabel 3.10.

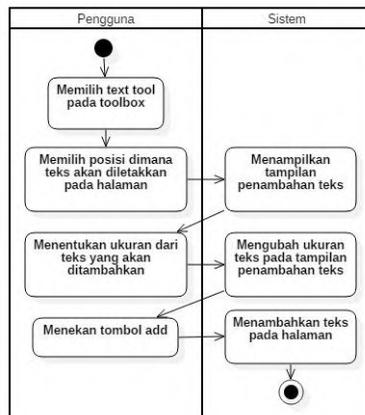
Tabel 3.10 Kasus Penggunaan Menentukan Ukuran Teks

Komponen	Deskripsi
Nama	Menentukan ukuran teks
Nomor	UC-011
Deskripsi	Pengguna dapat menentukan ukuran teks yang akan ditambahkan
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	-

Kondisi Akhir	Ukuran teks sudah ditentukan
Prekondisi	- Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih <i>text tool</i> pada <i>tool box</i> 2. Pengguna memilih posisi dimana teks akan diletakkan pada halaman. 3. Sistem menampilkan tampilan penambahan teks. 4. Pengguna menentukan ukuran dari teks yang akan ditambahkan 5. Sistem mengubah ukuran teks pada tampilan penambahan teks 6. Pengguna menekan tombol <i>add</i> 7. Sistem menambahkan teks ke halaman

3.1.2.13 Kasus Penggunaan Menentukan Jenis Teks

Pada kasus penggunaan ini pengguna dapat menentukan jenis teks yang akan ditambahkan ke dalam halaman yang sedang aktif. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.11 dan Gambar 3.14.



Gambar 3.13 Diagram Aktivitas Menentukan Ukuran Teks

Tabel 3.11 Kasus Penggunaan Menentukan Jenis Teks

Komponen	Deskripsi
Nama	Menentukan jenis teks
Nomor	UC-012
Deskripsi	Pengguna dapat menentukan jenis teks yang akan ditambahkan
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	-
Kondisi Akhir	Jenis teks sudah ditentukan
Prekondisi	- Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih <i>text tool</i> pada <i>tool box</i> 2. Pengguna memilih posisi dimana teks akan diletakkan pada halaman. 3. Sistem menampilkan tampilan penambahan teks. 4. Pengguna menentukan ukuran dari teks yang akan ditambahkan 5. Sistem mengubah jenis teks pada tampilan penambahan teks 6. Pengguna menekan tombol <i>add</i> 7. Sistem menambahkan teks ke halaman

3.1.2.14 Kasus Penggunaan Menentukan Warna Teks

Pada kasus penggunaan ini pengguna dapat menentukan warna dari teks yang akan ditambahkan ke dalam halaman yang sedang aktif. Rincian kasus penggunaan tersebut dapat dilihat pada Gambar 3.18 dan Tabel 3.12.



Gambar 3.14 Diagram Aktivitas Menentukan Jenis Teks

Tabel 3.12 Kasus Penggunaan Menentukan Warna Teks

Komponen	Deskripsi
Nama	Menentukan warna teks
Nomor	UC-013
Deskripsi	Pengguna dapat menentukan jenis teks yang akan ditambahkan
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	-
Kondisi Akhir	Warna teks sudah ditentukan
Prekondisi	- Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih <i>text tool</i> pada <i>tool box</i> 2. Pengguna memilih posisi dimana teks akan diletakkan pada halaman.

	<ol style="list-style-type: none"> 3. Sistem menampilkan tampilan penambahan teks. 4. Pengguna menekan tombol <i>color</i> pada tampilan penambahan teks 5. Sistem menampilkan tampilan pemilihan warna 6. Pengguna memilih warna yang tersedia 7. Pengguna menekan tombol OK 8. Pengguna menekan tombol <i>add</i> 9. Sistem menambahkan teks ke halaman
--	--

3.1.2.15 Kasus Penggunaan Memberi Efek Kontras

Pada kasus penggunaan ini pengguna dapat memberikan efek kontras pada gambar yang dipilih untuk ditambahkan pada halaman. Rincian kasus penggunaan tersebut dapat dilihat pada.



Gambar 3.15 Diagram Aktivitas Memberi Efek Kontras

Tabel 3.13 Kasus Penggunaan Memberi Efek Kontras

Komponen	Deskripsi
Nama	Memberi efek kontras
Nomor	UC-014
Deskripsi	Memberi efek kontras pada gambar yang dipilih
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Citra belum diberi efek kontras
Kondisi Akhir	Citra sudah diberi efek kontras
Prekondisi	- Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih lokasi citra yang ingin ditingkatkan kualitasnya 2. Sistem menampilkan citra apa saja yang dapat dipilih. 3. Pengguna melakukan <i>double click</i> pada citra yang dipilih. 4. Sistem menampilkan tampilan <i>edit window</i> 5. Pengguna memilih tipe peningkatan kontras 6. Sistem melakukan peningkatan kontras 7. Pengguna menekan tombol <i>draw</i> 8. Citra hasil olahan berhasil ditambahkan ke halaman

3.1.2.16 Kasus Penggunaan Memberi Efek Pencahayaan

Pada kasus penggunaan ini pengguna dapat memberikan efek pencahayaan pada gambar yang dipilih untuk ditambahkan pada halaman. Rincian kasus penggunaan tersebut dapat dilihat pada



Gambar 3.16 Diagram Aktivitas Memberi Efek Pencahayaan

Tabel 3.14 Kasus Penggunaan Memberi Efek Pencahayaan

Komponen	Deskripsi
Nama	Memberi efek pencahayaan
Nomor	UC-015
Deskripsi	Memberi efek pencahayaan pada gambar yang dipilih
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Citra belum diberi efek pencahayaan
Kondisi Akhir	Citra sudah diberi efek pencahayaan
Prekondisi	- Harus ada proyek yang sedang berjalan
Alur Normal	1. Pengguna memilih lokasi citra yang ingin ditingkatkan kualitasnya

	<ol style="list-style-type: none"> 2. Sistem menampilkan citra apa saja yang dapat dipilih. 3. Pengguna melakukan <i>double click</i> pada citra yang dipilih. 4. Sistem menampilkan tampilan <i>edit window</i> 5. Pengguna memilih tipe peningkatan pencahayaan 6. Sistem melakukan peningkatan pencahayaan 7. Pengguna menekan tombol <i>draw</i> 8. Citra hasil olahan berhasil ditambahkan ke halaman
--	---

3.1.2.17 Kasus Penggunaan Memberi Efek Ketajaman

Pada kasus penggunaan ini pengguna dapat memberikan efek ketajaman pada gambar yang dipilih untuk ditambahkan pada halaman. Rincian kasus penggunaan tersebut dapat dilihat pada



Gambar 3.17 Diagram Aktivitas Memberi Efek Ketajaman

Tabel 3.15 Kasus Penggunaan Memberi Efek Ketajaman

Komponen	Deskripsi
Nama	Memberi efek ketajaman
Nomor	UC-016
Deskripsi	Memberi efek ketajaman pada gambar yang dipilih
Tipe	Fungsional
Aktor	Pengguna
Kondisi Awal	Citra belum diberi efek ketajaman
Kondisi Akhir	Citra sudah diberi efek ketajaman
Prekondisi	- Harus ada proyek yang sedang berjalan
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih lokasi citra yang ingin ditingkatkan kualitasnya 2. Sistem menampilkan citra apa saja yang dapat dipilih. 3. Pengguna melakukan <i>double click</i> pada citra yang dipilih. 4. Sistem menampilkan tampilan <i>edit window</i> 5. Pengguna memilih peningkatan ketajaman 6. Sistem melakukan peningkatan ketajaman 7. Pengguna menekan tombol <i>draw</i> 8. Citra hasil olahan berhasil ditambahkan ke halaman

3.2 Perancangan

Pada subbab perancangan akan dijelaskan mengenai arsitektur sistem yang digunakan dan perancangan antarmuka pengguna.

3.2.1 Perancangan Arsitektur Sistem

Arsitektur sistem yang digunakan pada tugas akhir ini menggunakan arsitektur *standalone* karena aplikasi berdiri sendiri pada sebuah perangkat.

3.2.2 Perancangan Diagram Kelas

Perancangan diagram kelas aplikasi ini akan dijelaskan dalam beberapa diagram kelas, yaitu diagram kelas antarmuka dan diagram kelas elemen. Kelas antarmuka adalah kelas yang menampilkan antarmuka pengguna pada komputer sedangkan kelas elemen adalah kelas yang digunakan untuk menampung apapun yang dapat kita tambahkan ke dalam halaman *editor*.

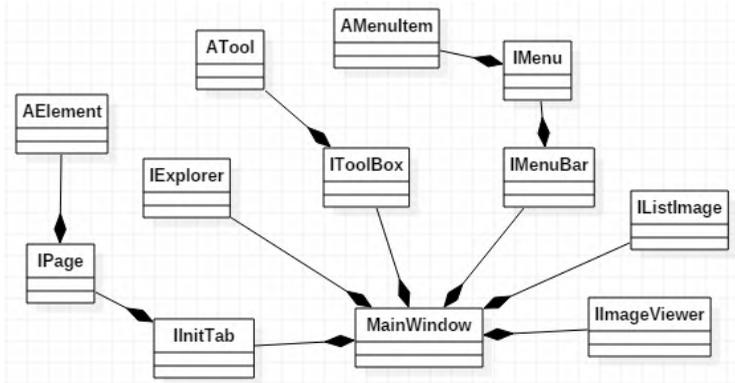


Gambar 3.18 Diagram Aktivitas Menentukan Warna Teks

3.2.2.1 Diagram Kelas MainWindow

Diagram kelas MainWindow merupakan rancangan tampilan utama pada aplikasi. Proses perancangan album foto juga dilakukan pada antarmuka ini. Kelas ini merupakan kelas Singleton seperti yang dijelaskan pada bab tinjauan pustaka.

Mengapa kelas MainWindow dibuat sebagai Singleton karena tiap-tiap elemen dari kelas ini akan sering diakses dari kelas lain sehingga lebih mudah jika dijadikan Singleton. Karena kelas MainWindow juga hanya dipanggil sekali maka sangat cocok dijadikan Singleton agar dapat diakses dari kelas lain tanpa harus mendefinisikan pada kelas lain.



Gambar 3.19 Diagram Kelas MainWindow

3.2.2.1.1 Kelas IMenuBar

Kelas IMenuBar merupakan *interface* yang diimplementasi oleh kelas MenuBar, berguna sebagai kelas yang menampung semua menu yang ada pada MainWindow. Kelas ini mempunyai *List<IMenu>* untuk menampung menu yang ditampilkan.

3.2.2.1.2 Kelas IMenu

Kelas Menu merupakan kelas yang berguna untuk menampung MenuItem, yaitu pilihan yang muncul apabila Menu dipilih. Kelas ini digunakan untuk mengelompokkan jenis-jenis MenuItem yang saling berhubungan.

3.2.2.1.3 Kelas AmenuItem

Kelas AmenuItem merupakan kelas abstrak dari MenuItem yang masing-masing akan diturunkan pada kelas kongkrit tertentu. Kelas ini berguna sebagai abstraksi utama MenuItem sehingga fungsi-fungsi dasar pada MenuItem tidak perlu ditulis ulang pada kelas turunannya.

3.2.2.1.4 Kelas IToolBox

Kelas IToolBox merupakan kelas yang digunakan untuk menampung ATool. Kelas ini merupakan kelas yang berisi *List<ATool>* yang berguna untuk memanipulasi halaman.

3.2.2.1.5 Kelas ATool

Kelas ATool merupakan kelas abstrak yang diturunkan pada masing-masing ToolItem. Fungsi-fungsi pada kelas ini merupakan fungsi dasar yang dimiliki tiap ToolItem. Penjelasan masing-masing fungsi terdapat pada Tabel 3.16.

Tabel 3.16 Fungsi Utama Kelas ATool

Fungsi	Kegunaan
<code>void generateIcon()</code>	Menampilkan icon untuk ToolItem

<code>void dragDetected(DragDetectEvent e)</code>	Mengubah kondisi <i>drag</i> pada <i>ATool</i> menjadi <i>true</i>
<code>IPage getActivePage()</code>	Mendapatkan halaman yang aktif
<code>void select()</code>	Memilih <i>ToolItem</i> yang aktif
<code>void deselect()</code>	Menonaktifkan <i>ToolItem</i>

3.2.2.1.6 Kelas IExplorer

Kelas *IExplorer* merupakan kelas *Tree* yang diturunkan dengan penambahan beberapa fungsi. Kelas ini berfungsi untuk menampilkan *directory* file komputer sehingga kita dapat memilih lokasi file tempat menyimpan gambar. Penjelasan tentang fungsi utama kelas dapat dilihat pada Tabel 3.17.

Tabel 3.17 Fungsi Utama Kelas IExplorer

Fungsi	Kegunaan
<code>void handleEvent(Event e)</code>	Menampilkan anak <i>directory</i> dari <i>directory</i> terpilih
<code>void widgetSelected(Selection Event e)</code>	Menampilkan gambar yang berada pada <i>directory</i> , gambar-gambar yang dapat ditambahkan akan dimuat pada kelas <i>IListImage</i>

3.2.2.1.7 Kelas IListImage

Kelas *IListImage* berfungsi untuk menampilkan gambar apa saja yang dapat ditambahkan pada halaman. Kelas ini berhubungan dengan kelas *IExplorer*. Penjelasan mengenai fungsi-fungsi utama pada kelas ini dapat dilihat pada Tabel 3.18. Penjelasan mengenai fungsi-fungsi utama pada kelas ini dapat dilihat pada Tabel 3.18.

Tabel 3.18 Fungsi Utama Kelas IListItem

Fungsi	Kegunaan
<code>void clearList()</code>	Menghapus semua isi dari IListItem
<code>void addItem(String item)</code>	Menambahkan data gambar pada IListItem
<code>void handleEvent(Event e)</code>	Menampilkan tampilan EditImageWindow yang digunakan untuk menambahkan gambar ke dalam halaman
<code>void widgetSelected(Event e)</code>	Melihat <i>preview</i> dari gambar terpilih pada ImageViewer

3.2.2.1.8 Kelas ImageViewer

Kelas ImageViewer berguna untuk menampilkan gambar yang terpilih pada kelas IListItem sehingga pengguna dapat melihat *preview* gambar sebelum akhirnya dipilih. Penjelasan fungsi utama kelas ini dapat dilihat pada Tabel 3.19.

Tabel 3.19 Fungsi Utama Kelas ImageViewer

Fungsi	Kegunaan
<code>void setImage(String imagePath)</code>	Menampilkan <i>preview</i> gambar yang dipilih dari kelas IListItem

3.2.2.1.9 Kelas IInitTab

Kelas IInitTab berfungsi untuk menampung halaman dari aplikasi. Kelas ini memiliki *List<IPage>* untuk menampung IPage. Kelas ini akan dipanggil saat aplikasi berjalan dengan kondisi jumlah IPage masih *null*. Apabila proyek baru dibuat maka jumlah IPage akan bertambah sesuai

dengan jumlah yang ditentukan. Penjelasan fungsi utama kelas ini dapat dilihat pada Tabel 3.20.

Tabel 3.20 Fungsi Utama Kelas IInitTab

Fungsi	Kegunaan
<code>void clearList()</code>	Menghapus semua isi dari <code>IListImage</code>
<code>void addItem(String item)</code>	Menambahkan data gambar pada <code>IListImage</code>
<code>void handleEvent(Event e)</code>	Menampilkan tampilan <code>EditImageWindow</code> yang digunakan untuk menambahkan gambar ke dalam halaman
<code>void widgetSelected(Event e)</code>	Melihat <i>preview</i> dari gambar terpilih pada <code>ImageViewer</code>

3.2.2.1.10 Kelas IPage

Kelas `Page` berfungsi untuk menampung *Canvas* tempat pengguna menggambar elemen yang diinginkan. Kelas `IPage` juga memiliki kelas *Composite* agar ukuran *Canvas* tidak dinamis mengikuti ukuran *Window*. Penjelasan fungsi utama kelas `IPage` dapat dilihat pada Tabel 3.21.

Tabel 3.21 Fungsi Utama Kelas IPage

Fungsi	Kegunaan
<code>void close()</code>	Menutup halaman yang sedang aktif
<code>void clearCanvas()</code>	Membersihkan <i>Canvas</i> sehingga menjadi putih
<code>void addElement()</code>	Menambahkan elemen pada <code>List<IElement></code> yang dimiliki kelas <code>IPage</code>
<code>void removeElement()</code>	Menghapus satu elemen pada <code>List<IElement></code>

<code>void draw()</code>	Menggambar seluruh elemen yang tersimpan dalam <i>List<IElement></i>
<code>void mouseDoubleClick(MouseEvent e)</code>	Menjalankan perintah <i>double click</i> apabila dieksekusi. Perintah ini berbeda-beda, tergantung <i>Tool</i> apa yang sedang aktif
<code>void mouseDown(MouseEvent e)</code>	Menjalankan perintah saat <i>mouse</i> diklik ke <i>Canvas</i> . Perintah ini berbeda-beda tergantung <i>Tool</i> apa yang sedang aktif
<code>void mouseUp(MouseEvent e)</code>	Menjalankan perintah saat setelah <i>mouse</i> diklik ke <i>Canvas</i> . Perintah ini berbeda-beda tergantung <i>Tool</i> apa yang sedang aktif.
<code>Void mouseMove(MouseEvent e)</code>	Menjalankan perintah saat <i>mouse</i> digerakkan. Perintah ini berbeda-beda tergantung <i>Tool</i> apa yang sedang aktif.
<code>Void dragDetected(MouseEvent e)</code>	Menjalankan perintah saat <i>mouse</i> didrag. Perintah ini berbeda-beda tergantung <i>Tool</i> apa yang sedang aktif
<code>void IElement getElement(int x, int y)</code>	Mendapatkan elemen apa yang dipilih di <i>Canvas</i> saat koordinat <i>x, y</i> diklik.
<code>Void TwoDimensional getTwoDimensional(int x, int y)</code>	Mendapatkan <i>TwoDimensional</i> apa yang dipilih di <i>Canvas</i> saat koordinat <i>x, y</i> diklik.
<code>Void deselectAll()</code>	Membatalkan pilihan elemen setelah elemen dipilih
<code>void List<IElement> getSelectedElements()</code>	Mendapatkan semua elemen yang dipilih

3.2.2.1.11 Kelas AElement

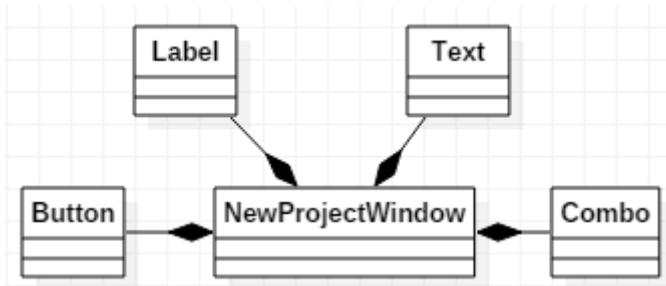
Kelas *AElement* merupakan kelas yang digunakan sebagai abstraksi dari kelas kongkrit yang ada. Kelas kongkrit

yang dimaksud adalah kelas ControlPoint, Picture, Paragraph, dan Frame. Kelas ini berisi fungsi-fungsi dasar yang ada pada tiap kelas kongkrit. Penjelasan fungsi utama kelas AElement dapat dilihat pada Tabel 3.22.

Tabel 3.22 Fungsi Utama Kelas AElement

Fungsi	Kegunaan
<code>void select()</code>	Membuat elemen yang terpilih masuk <i>SizingState</i>
<code>void deselect()</code>	Membatalkan pilihan elemen
<code>void isActive()</code>	Bernilai benar jika elemen terpilih
<code>void draw()</code>	Menggambar elemen tergantung dari <i>state</i> elemen saat ini
<code>void drag(int x, int y, int x2, int y2)</code>	Memindahkan posisi elemen yang dipilih

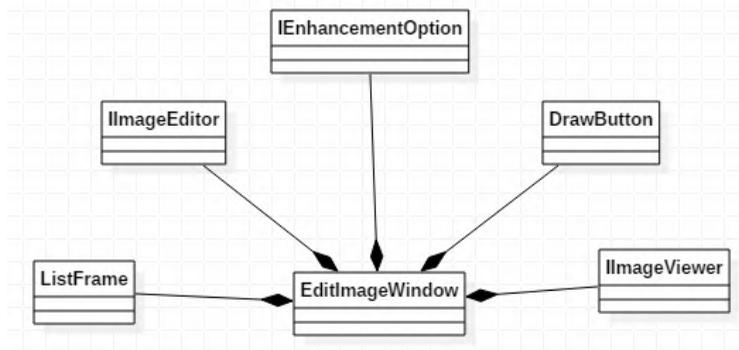
3.2.2.2 Diagram Kelas NewProjectWindow



Gambar 3.20 Diagram Kelas NewProjectWindow

Diagram kelas `NewProjectWindow` merupakan rancangan tampilan untuk membuat proyek baru pada aplikasi. Kelas ini terdiri dari beberapa `Label`, `Text`, sebuah `Button` dan sebuah `Combo`.

3.2.2.3 Diagram Kelas EditImageWindow



Gambar 3.21 Diagram Kelas EditImageWindow

Diagram kelas *EditImageWindow* merupakan rancangan tampilan untuk melakukan *image processing* dan memberikan bingkai pada gambar terpilih.

3.2.2.3.1 Kelas AenhancementOption

Kelas ini merupakan kelas yang berfungsi untuk melakukan pengolahan citra yang dipilih. Kelas ini merupakan kelas abstract yang fungsi inisialisasinya dibedakan untuk tiap kelas kongkritnya. Kelas ini diturunkan menjadi empat kelas kongkrit yaitu kelas *BrightnessEnhancement*, *ContrastEnhancement*, *NormalEnhancement*, dan *SharpEnhancement*. Penjelasan dari fungsi utama kelas ini dapat dilihat pada

Tabel 3.23 Fungsi Utama Kelas AenhancementOption

Fungsi	Kegunaan
<code>void initialize()</code>	Fungsi yang nantinya akan diturunkan dan berbeda-beda untuk tiap kelas turunannya

<code>void setEnhancementName (String name)</code>	Memberi nama <i>enhancement</i> yang muncul pada <i>window</i> .
--	--

3.2.2.3.2 Kelas ImageEditor

Kelas ini berguna untuk menampilkan gambar hasil olahan aplikasi. Saat pengguna memilih *enhancement* yang disediakan maka kelas ini akan menampilkan *preview* hasil olahan gambar sehingga pengguna dapat menentukan *enhancement* mana yang paling cocok untuk gambar. Penjelasan fungsi utama kelas ini dapat dilihat pada Tabel 3.24.

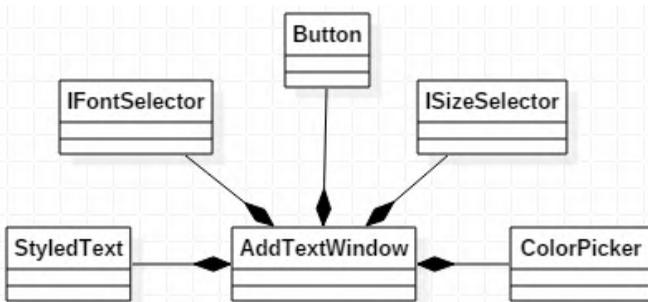
Tabel 3.24 Fungsi Utama Kelas ImageEditor

Fungsi	Kegunaan
<code>void reCallPicture(ImageData imgdata)</code>	Mengganti <i>preview</i> gambar tiap ada perubahan <i>enhancement</i>

3.2.2.3.3 Kelas ListFrame

Kelas ListFrame berguna untuk menampilkan Frame apa saja yang dapat dipilih pengguna untuk ditambahkan ke dalam halaman beserta dengan gambar. Fungsi utama kelas ini hampir sama seperti kelas ListImage.

3.2.2.4 Diagram Kelas AddTextWindow



Gambar 3.22 Diagram Kelas AddTextWindow

Diagram kelas `AddTextWindow` merupakan rancangan tampilan untuk menambahkan teks pada halaman. Dalam kelas ini kita dapat mengubah jenis teks, ukuran teks, dan warna teks sesuai dengan yang kita inginkan.

3.2.2.4.1 Kelas `StyledText`

Kelas `StyledText` merupakan kelas bawaan Java SWT. Kelas ini berguna untuk menuliskan teks yang nantinya akan ditambahkan ke halaman.

3.2.2.4.2 Kelas `IFontSelector`

Kelas `IFontSelector` berguna untuk memilih jenis teks yang akan ditambahkan ke dalam halaman. Kelas ini merupakan turunan dari kelas *Combo* milik Java SWT.

3.2.2.4.3 Kelas `ISizeSelector`

Kelas `ISizeSelector` berguna untuk memilih ukuran teks yang akan ditambahkan ke dalam halaman. Kelas ini merupakan kelas turunan dari kelas *Combo* milik Java SWT.

3.2.2.4.4 Kelas `IColorPicker`

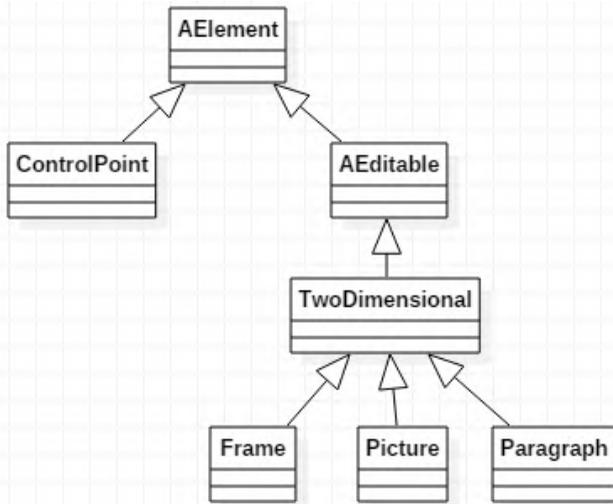
Kelas `IColorPicker` adalah kelas yang berguna untuk memilih warna dari teks yang nantinya akan ditambahkan ke dalam halaman. Kelas ini memiliki fungsi `getColor()` yang berguna untuk mendapatkan warna yang nantinya dipasang pada *Font* sehingga teks menjadi berwarna.

3.2.2.5 Diagram Kelas `Element`

Diagram kelas *Element* merupakan struktur dari obyek apapun yang dapat dimuat pada halaman. Pada dasarnya obyek yang dapat dimuat pada halaman adalah gambar dan teks saja

namun pada diagram kelas *Element* akan terdapat beberapa kelas yang merupakan abstraksi dari kelas dibawahnya. Sebagai contoh kelas *TwoDimensional* adalah kelas abstrak dari kelas dibawahnya yaitu kelas *Paragraph* dan kelas *Picture*.

Total pada diagram kelas *Element* ini terdapat 7 kelas yang nantinya berfungsi saling berhubungan untuk membentuk kelas kongkrit yang dapat diinstansiasi sehingga membentuk obyek.



Gambar 3.23 Diagram Kelas Element

3.2.2.5.1 Kelas ControlPoint

Kelas *ControlPoint* merupakan kelas yang dibuat apabila pengguna memilih elemen sehingga elemen yang dipilih masuk pada *SizingState* dan dapat dimanipulasi. Penjelasan fungsi utama kelas *ControlPoint* dapat dilihat pada Tabel 3.25.

Tabel 3.25 Fungsi Utama kelas ControlPoint

Fungsi	Kegunaan
<code>void select()</code>	Mengubah kondisi elemen menjadi <i>SizingState</i>
<code>void renderNormal()</code>	Memanggil fungsi <code>select</code> dan <code>renderSize</code>
<code>void renderSize()</code>	Membentuk <code>ControlPoint</code>
<code>void checkBoundary(int x, int y, int x2, int y2)</code>	Mengecek apakah ada <code>ControlPoint</code> yang dipilih
<code>void drag(int x, int y, int x2, int y2)</code>	Memindahkan <code>ControlPoint</code>

3.2.2.5.2 Kelas Aeditable

Kelas ini merupakan kelas abstrak untuk semua elemen yang dapat berubah posisi dan ukuran. Turunan dari kelas ini adalah kelas abstrak lain yaitu kelas `TwoDimensional`. Kelas ini merepresentasikan perilaku dari objek yang dapat diubah. Penjelasan fungsi utama dari kelas ini dapat dilihat pada Tabel 3.26.

Tabel 3.26 Fungsi Utama Kelas Aeditable

Fungsi	Kegunaan
<code>void select()</code>	Mengubah kondisi elemen menjadi <i>SizingState</i>
<code>void deselect()</code>	Mengubah kondisi elemen menjadi normal
<code>void renderSize()</code>	Membentuk <code>ControlPoint</code>
<code>void checkBoundary(int x, int y, int x2, int y2)</code>	Mengecek apakah ada elemen <code>Aeditable</code> yang dipilih
<code>void drag(int x, int y, int x2, int y2)</code>	Memindahkan <code>ControlPoint</code>
<code>void addControlPoint (ControlPoint controlPoint)</code>	Menambahkan sebuah <code>ControlPoint</code> pada <code>List<ControlPoint></code> kelas ini.
<code>Void createControlPoint (ArrayList<Point> points)</code>	Bersama dengan <code>addControlPoint</code> fungsi ini

	berguna untuk membentuk seluruh ControlPoint
--	--

3.2.2.5.3 Kelas TwoDimensional

Kelas TwoDimensional merupakan kelas abstrak yang nantinya akan diturunkan menjadi kelas kongkrit Paragraph, Picture, dan Frame. Kelas ini merepresentasikan perilaku dari kelas yang dapat ditambahkan kedalam halaman. Penjelasan fungsi-fungsi utama kelas ini dapat dilihat pada

Tabel 3.27 Fungsi Utama Kelas TwoDimensional

Fungsi	Kegunaan
<code>void setLocation(int x, int y)</code>	Memposisikan koordinat x dan y pada lokasi yang ditentukan
<code>void createControlPoints()</code>	Membentuk dan menampilkan ControlPoint pada elemen terpilih
<code>void setBoundary(int x, int y, int x2, int y2)</code>	Menentukan pembatas dari kelas TwoDimensional
<code>void checkBoundary(int x, int y, int x2, int y2)</code>	Memeriksa apakah ada elemen TwoDimensional yang dipilih
<code>void drag(int x, int y, int x2, int y2, IElement e)</code>	Menggeser elemen jika e adalah TwoDimensional, dan mengubah pembatas bila e adalah ControlPoint

3.2.2.5.4 Kelas Picture

Kelas Picture merupakan representasi dari tiap gambar yang dimuat dalam halaman. Kelas ini merupakan kelas turunan dan kelas kongkrit dari kelas TwoDimensional. Setiap pengguna menambahkan gambar maka kelas Picture akan diinstansiasi menjadi sebuah objek. Penjelasan fungsi utama dari kelas ini dapat dilihat pada Tabel 3.28.

Tabel 3.28 Fungsi Utama Kelas Picture

Fungsi	Kegunaan
<code>ImageData scaleImage (ImageData imgdata)</code>	Mengubah ukuran asli gambar sehingga dapat disesuaikan dengan ukuran halaman
<code>void renderNormal()</code>	Menggambar pada halaman
<code>void notify(IElement observable)</code>	Menyesuaikan posisi dan ukuran dengan Frame, apabila gambar memiliki bingkai

3.2.2.5.5 Kelas Frame

Kelas Frame merupakan representasi dari bingkai yang nantinya dapat ditambahkan ke dalam halaman beserta dengan gambar yang terpilih. Fungsi-fungsi utama kelas ini sebenarnya hampir sama dengan kelas Picture, akan tetapi pada kelas ini memiliki *observer* yang digunakan untuk implementasi *observer pattern* yang dijelaskan pada Bab 2. Penjelasan fungsi utama kelas Frame dapat dilihat pada Tabel 3.29.

Tabel 3.29 Fungsi Utama Kelas Frame

Fungsi	Kegunaan
<code>ImageData scaleImage (ImageData imgdata)</code>	Mengubah ukuran asli gambar sehingga dapat disesuaikan dengan ukuran halaman
<code>void renderNormal()</code>	Menggambar pada halaman
<code>void notifyAllObservers()</code>	Menjalankan fungsi <i>notify</i> pada <i>observer</i> kelas Frame

3.2.2.5.6 Kelas Paragraph

Kelas Paragraph merupakan kelas yang merepresentasi teks yang akan ditambahkan ke dalam halaman. Kelas ini menampung segala informasi tentang ukuran teks, warna teks, dan jenis teks. Fungsi-fungsi pada kelas ini hanyalah fungsi *setter* dan *getter* tentang informasi *font*.

3.2.3 Perancangan Antarmuka Pengguna

Perancangan antarmuka pengguna merupakan hal yang penting dalam melakukan perancangan aplikasi. Antarmuka pengguna yang berhubungan langsung dengan faktor harus memiliki kemudahan-kemudahan bagi penggunanya. Sistem memiliki 4 antarmuka pengguna, yaitu halaman *MainWindow*, *EditImageWindow*, *NewProjectWindow*, *AddTextWindow*.

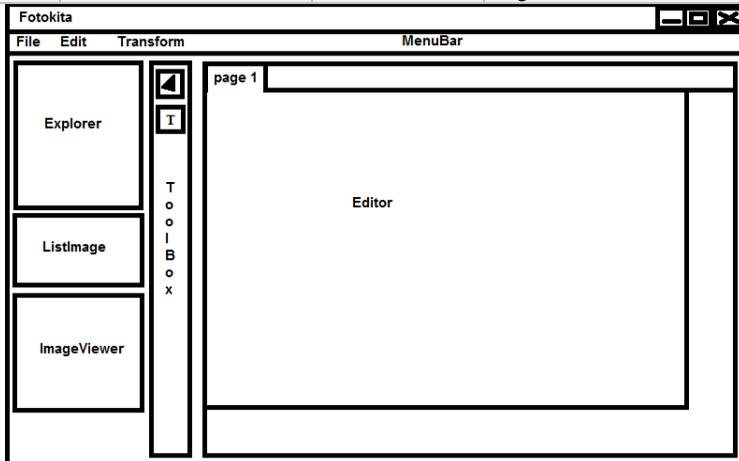
3.2.3.1 Rancangan Halaman Antarmuka MainWindow

Halaman ini merupakan halaman utama dari aplikasi. Pada halaman ini terdiri dari dua bagian, yaitu bagian editor dan file explorer. Bagian file explorer adalah sekumpulan *widgets* untuk memilih gambar yang akan ditambahkan ke halaman album foto. Bagian file explorer terdiri dari *explorer*, *list image*, dan *image viewer*. *Explorer* berfungsi sebagai alat bantu untuk memilih lokasi dari gambar yang akan ditambahkan. *List image* berfungsi untuk menampung gambar mana saja pada lokasi yang dipilih untuk ditambahkan. *Image viewer* berfungsi sebagai penampil gambar secara sementara agar pengguna dapat mengetahui bagaimana gambar yang dimaksud. Rancangan halaman antarmuka dapat dilihat pada Gambar 3.32 dan Tabel 3.30.

Tabel 3.30 Penjelasan Antarmuka MainWindow

No.	Nama Antarmuka	Atribut	Jenis Atribut	Kegunaan
1	<i>Explorer</i>		<i>Tree</i>	Menampilkan <i>directory</i> komputer
2	<i>ListImage</i>		<i>List</i>	Menampilkan gambar mana saja yang dapat dipilih.

3	<i>ImageViewer</i>	<i>Canvas</i>	Menampilkan <i>preview</i> gambar
4	<i>Editor</i>	<i>TabFolder</i>	Sebagai halaman album, tempat merancang album foto
5	<i>ToolBox</i>	<i>ToolBar</i>	Menampilkan tool apa saja yang dapat digunakan



Gambar 3.24 Rancangan Antarmuka Main Window

3.2.3.2 Rancangan Halaman Antarmuka New Project Window

Antarmuka ini digunakan untuk membuat proyek baru pada aplikasi. Pada antarmuka ini terdapat isian nama penulis, jumlah halaman, judul album, dan ukuran album. Rancangan antarmuka ini dapat dilihat pada Gambar 3.25 dan Tabel 3.31.

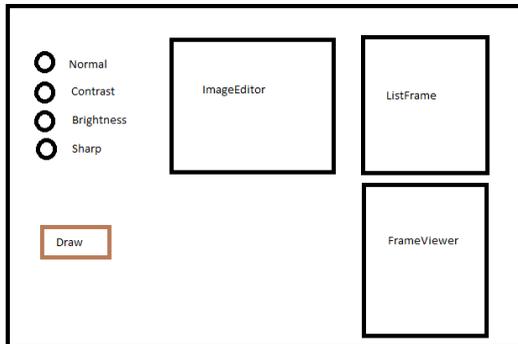
Gambar 3.25 Rancangan Antarmuka New Project Window

Tabel 3.31 Penjelasan Antarmuka New Project Window

No.	Nama Antarmuka	Atribut	Jenis Atribut	Kegunaan
1	<i>Author</i>		<i>Text</i>	Menyimpan nama penulis
2	<i>Pages</i>		<i>Text</i>	Menyimpan jumlah halaman
3	<i>Title</i>		<i>Text</i>	Memberi judul album
4	<i>Size</i>		<i>Text</i>	Menentukan ukuran kertas yang dipakai

3.2.3.3 Rancangan Halaman Antarmuka EditImageWindow

Antarmuka ini digunakan untuk memberikan efek pada gambar yang akan ditambahkan ke dalam halaman. Efek yang disediakan adalah peningkatan kontras, peningkatan pencahayaan, serta peningkatan ketajaman gambar. Gambar yang ingin ditambahkan tidak harus diberi efek. Halaman antarmuka dapat dilihat pada Gambar 3.26 dan Tabel 3.32.



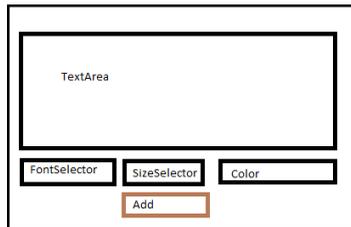
Gambar 3.26 Rancangan Antarmuka Edit Image Window

Tabel 3.32 Penjelasan Antarmuka Edit Image Window

No.	Nama Antarmuka	Atribut	Jenis Atribut	Kegunaan
1	<i>Normal</i>		<i>Radio</i>	Tidak memberi efek apapun pada gambar, menormalkan gambar
2	<i>Contrast</i>		<i>Radio</i>	Meningkatkan kontras gambar
3	<i>Brightness</i>		<i>Radio</i>	Meningkatkan pencahayaan gambar
4	<i>Sharp</i>		<i>Radio</i>	Meningkatkan ketajaman gambar
5	<i>Draw</i>		<i>Button</i>	Menambahkan gambar ke halaman aktif
6	<i>ListFrame</i>		<i>List</i>	Menampilkan pilihan bingkai
7	<i>FrameViewer</i>		<i>Canvas</i>	Menampilkan contoh bingkai
8	<i>ImageEditor</i>		<i>Canvas</i>	Menampilkan contoh gambar yang akan dimuat

3.2.3.4 Rancangan Halaman Antarmuka AddTextWindow

Antarmuka ini digunakan untuk menambahkan teks pada halaman yang sedang aktif. Kita dapat memilih jenis teks yang kita inginkan dan ukuran teks yang kita inginkan. Halaman antarmuka dapat dilihat pada Gambar 3.27 dan Tabel 3.33



Gambar 3.27 Rancangan Antarmuka Add Text Window

Tabel 3.33 Penjelasan Antarmuka Add Text Window

No.	Nama Antarmuka	Atribut	Jenis Atribut	Kegunaan
1	<i>TextArea</i>		<i>StyledText</i>	Tempat pengguna mengetikkan teks yang ingin ditambahkan
2	<i>FontSelector</i>		<i>Combo</i>	Berguna untuk menentukan jenis teks
3	<i>SizeSelector</i>		<i>Combo</i>	Berguna untuk menentukan ukuran teks
4	<i>ColorButton</i>		<i>Button</i>	Menampilkan tampilan pilihan warna
5	<i>AddButton</i>		<i>Button</i>	Menambahkan teks ke halaman aktif

BAB IV

IMPLEMENTASI

Pada bab ini akan dibahas mengenai implementasi 83system sesuai dengan analisis dan perancangan sistem secara umum pada aplikasi editor album foto digital yang dijabarkan pada bab sebelumnya.

Implementasi yang dijelaskan meliputi lingkungan pembangunan perangkat lunak, kode sumber utama dan implementasi antarmuka perangkat lunak. Bahasa pemrograman yang digunakan adalah Java dengan *framework* untuk tampilan Java SWT.

4.1 Lingkungan Implementasi

Lingkungan implementasi yang akan digunakan untuk melakukan implementasi pada tugas akhir ini dilakukan pada lingkungan dengan kakas sebagai berikut.

1. Sistem operasi Windows 10 Enterprise 64 bit.
2. Eclipse Mars digunakan untuk IDE.
3. StarUML 2.7 digunakan untuk membuat diagram kasus penggunaan dan diagram aktivitas.

4.2 Implementasi Pola Perancangan

Pada subbab ini penulis akan menjelaskan tentang pola perancangan yang digunakan pada aplikasi. Pola perancangan yang digunakan sesuai pada Bab 2. Pada masing-masing penjelasan akan disertakan kode sumber sebagai implementasi dalam aplikasi.

4.2.1 Pola Perancangan Singleton

Dalam aplikasi yang dibangun, pola perancangan *singleton* digunakan pada kelas `MainWindow`, `NormalState`, dan `SizingState`. Ciri-ciri dari kelas *singleton* dapat dilihat dari fungsi *constructor* yang dimiliki. Masing-masing *constructor* dari tiap kelas dapat dilihat pada Kode Sumber 4.1, Kode Sumber 4.2, dan Kode Sumber 4.3.

```

Public class MainWindow extends Awindow {
    private ImenuBar menuBar;
    private static MainWindow instance;
    ...
    private MainWindow() {
    }

    public static MainWindow getInstance() {
        if (instance == null) {
            instance = new MainWindow();
        }
        return instance;
    }
}

```

Kode Sumber 4.1 Constructor MainWindow

```

public class NormalState implements IDrawingState{
    private static IDrawingState instance;

    private NormalState() {
    }

    public static IDrawingState getInstance() {
        if (instance == null) {
            instance = new NormalState();
        }
        return instance;
    }
}

```

Kode Sumber 4.2 Constructor NormalState

```

public class SizingState implements IDrawingState{
    private static IDrawingState instance;

    private SizingState() {
    }

    public static IDrawingState getInstance() {
        if (instance == null) {
            instance = new SizingState();
        }
        return instance;
    }
}

```

Kode Sumber 4.3 Constructor SizingState

4.2.2 Pola Perancangan Observer Pattern

Pola perancangan *observer pattern* dalam aplikasi diimplementasi pada kelas `Frame` yang memiliki `ArrayList<IElement>` sebagai *observer*. Kelas `Frame` yang menjadi *observable* akan menjalankan fungsi `notifyAllObserver` saat objek dari kelas `Frame` menjalankan fungsi `drag`. Implementasi *observer pattern* dapat dilihat pada Kode Sumber 4.4.

```

Public class Frame extends TwoDimensional {
    private ArrayList<IElement> observers =
new ArrayList<>();
    ...
    public void notifyAllObserver(){
        for(IElement element: observers){
            element.notify(this);
        }
    }
    @Override
    public void drag(int x1, int y1, int x2, int
y2, IElement e) {
        super.drag(x1, y1, x2, y2, e);
        notifyAllObserver();
    }
    @Override

```

```

        public void drag(int x1, int y1, int x2, int
y2) {
            super.drag(x1, y1, x2, y2);
            notifyAllObserver();
        }

```

Kode Sumber 4.4 Implementasi Observer Pattern Kelas Frame

4.2.3 Pola Perancangan State Pattern

Pola perancangan *state pattern* diimplementasi pada kelas *NormalState* dan *SizingState*. Penggunaan kelas-kelas tersebut digunakan pada tiap *state*. Tiap kelas *state* memiliki fungsi *draw* yang berbeda. Kelas tersebut mengimplementasi fungsi *draw* dari *interface* *IDrawingState*. Sebagai contoh akan dijelaskan kode sumber kelas *SizingState* yang dapat dilihat pada Kode Sumber 4.5.

```

Public class SizingState implements IDrawingState{
    ...
    @Override
    public void draw(IElement element) {
        element.renderSize();
    }
}

```

Kode Sumber 4.5 Fungsi draw Kelas SizingState

Fungsi *draw* pada masing-masing *state* akan dipanggil saat fungsi *draw* pada *AElement* dipanggil. Perubahan *state* akan diimplementasi pada fungsi *select* kelas *AElement*. Masing-masing fungsi dapat dilihat pada Kode Sumber 4.6 dan Kode Sumber 4.7.

```

Public void draw(){
    state.draw(this);
};

```

Kode Sumber 4.6 Fungsi draw Kelas AElement

```
public void select() {  
    if (state == NormalState.getInstance()) {  
        state = SizingState.getInstance();  
    }else {  
        state = NormalState.getInstance();  
    }  
}
```

Kode Sumber 4.7 Fungsi select Kelas AElement

4.3 Implementasi Editor Grafis

Implementasi editor grafis aplikasi menggunakan Java SWT. Pada subbab ini menjelaskan dan menampilkan tampilan halaman sesuai dengan rancangan antarmuka yang terdapat pada Bab 3.

4.3.1 Implementasi Antarmuka

Pada bagian ini penulis akan menjelaskan tentang hasil dari implementasi dari rancang bangun antarmuka yang sesuai dengan apa yang telah dijelaskan pada Bab 3.

4.3.1.1 Halaman MainWindow

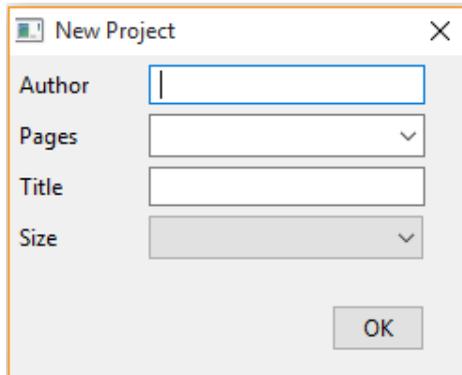
Implementasi kelas antarmuka MainWindow merupakan tampilan utama dari aplikasi. Tampilan antarmuka MainWindow dapat dilihat pada Gambar 4.1.

4.3.1.2 Halaman NewProjectWindow

Implementasi NewProjectWindow merupakan halaman yang digunakan untuk membuat proyek baru pada aplikasi. Tampilan untuk membuat proyek baru dapat dilihat pada Gambar 4.2.



Gambar 4.1 Halaman Antarmuka MainWindow

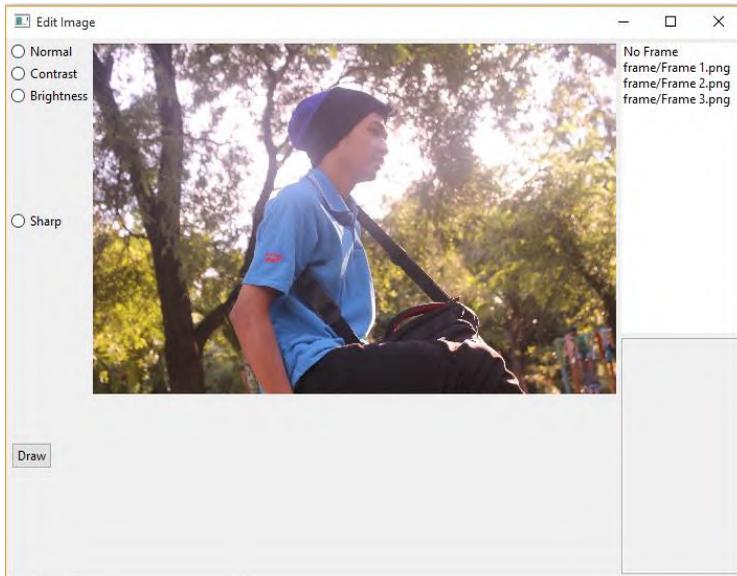


Gambar 4.2 Halaman Antarmuka New Project Window

4.3.1.3 Halaman EditImageWindow

Implementasi EditImageWindow merupakan implementasi dari halaman yang digunakan untuk menambahkan gambar pada halaman. Pada tampilan ini juga pengguna dapat mengatur efek yang akan digunakan pada gambar terpilih. Pada tampilan ini juga pengguna dapat

menambahkan bingkai. Tampilan EditImageWindow dapat dilihat pada Gambar 4.3.



Gambar 4.3 Halaman Antarmuka Edit Image Window

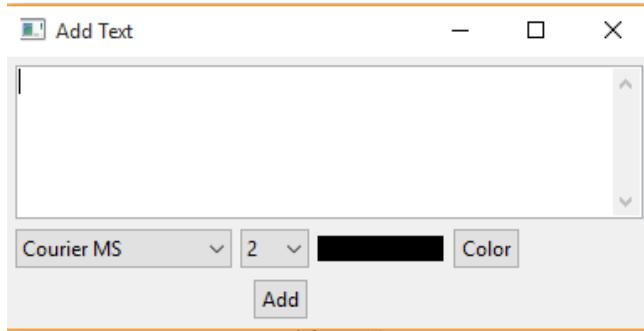
4.3.1.4 Halaman AddTextWindow

Implementasi AddTextWindow merupakan implementasi dari tampilan untuk menambahkan teks pada halaman. Dalam tampilan ini pengguna mengisi bagian TextArea untuk *preview* teks. Pengguna dapat mengatur jenis teks, ukuran teks, dan warna dari teks yang akan ditambahkan. Tampilan AddTextWindow dapat dilihat pada Gambar 4.4.

4.3.2 Implementasi Fungsi Editor

Dalam subbab ini akan dijelaskan fungsi-fungsi dasar editor grafis pada aplikasi. Fungsi yang dijelaskan antara lain fungsi untuk menambah gambar, mengubah posisi gambar,

mengubah ukuran gambar, merotasi gambar, dan memanipulasi penambahan teks pada halaman. Pada masing-masing bagian akan disertakan pula kode sumber sebagai pendukung bila diperlukan.



Gambar 4.4 Halaman Antarmuka Add Text Window

4.3.2.1 Fungsi Menambah Gambar

Fungsi menambah gambar merupakan fungsi menginstansiasi kelas `Picture` dan menambahkannya ke dalam `List` yang digunakan untuk menampung tiap elemen terdaftar. Setelah elemen ditambahkan ke dalam `List`, tiap elemen akan digambar pada halaman. Kode sumber mengenai mekanisme penambahan gambar dapat dilihat pada Kode Sumber 4.8. Untuk proses penggambaran terdapat pada fungsi `renderNormal` pada kelas `Picture`, dapat dilihat pada Kode Sumber 4.9.

```
int height = editWindow.getImageData().height;
int width = editWindow.getImageData().width;
ImageData imgeData = editWindow.getImageData();
String path = editWindow.getImagePath();
String effect = editWindow.getEffect();
```

```
Picture pic = new Picture(0, 0, height, width,
    imageData, path, effect);

MainWindow.getInstance().getInitTab().getActivePage()
    .addElement(pic);
```

Kode Sumber 4.8 Kode Sumber Penambahan Gambar

```
GC gc = new
GC(MainWindow.getInstance().getInitTab().getActivePage()
    .getCanvas());
gc.drawImage(getImage(), 0, 0, getImgData().width,
    getImgData().height, getX(), getY(), getWidth(),
    getHeight());
gc.dispose();
```

Kode Sumber 4.9 Fungsi renderNormal Kelas Picture

4.3.2.2 Fungsi Mengubah Posisi Gambar

Fungsi mengubah posisi gambar merupakan fungsi untuk mengubah nilai x dan y pada gambar. Fungsi ini terdapat pada kelas abstrak *TwoDimensional* yaitu fungsi *drag*. Kode sumber dapat dilihat pada Kode Sumber 4.10.

```
Public void drag(int x1, int y1, int x2, int y2) {
    int x = getX() + x2 - x1;
    int y = getY() + y2 - y1;
    setLocation(x, y);
    super.drag(x1, y1, x2, y2);
}
```

Kode Sumber 4.10 Fungsi drag TwoDimensional

Fungsi tersebut akan dipanggil saat cursor memilih *PointerTool* dan melakukan *drag and drop* pada halaman. Implementasi eksekusi fungsi ini dapat dilihat pada kelas *PointerTool* fungsi *mouseUp* seperti yang diberikan pada Kode Sumber 4.11.

```

...
case DRAG_MULTIPLE:
    ArrayList<IElement> temp = new ArrayList<>();
    for (IElement element : getActivePage().getSelectedElements()) {
        element.drag(downTemp.x, downTemp.y, e.x, e.y);
        temp.add(element);
    }
    getActivePage().deselectAll();
    for(IElement element : temp){
        element.select();
    }
    break;
...

```

Kode Sumber 4.11 Kode Sumber Mengeksekusi Fungsi drag

4.3.2.3 Fungsi Mengubah Ukuran Gambar

Fungsi mengubah ukuran gambar terdapat pada kelas `TwoDimensional` pada fungsi `drag`. Fungsi tersebut membedakan saat pengguna memilih `ControlPoint` atau tidak. Saat pengguna memilih `ControlPoint` maka perlakuan akan berbeda saat pengguna melakukan *drag and drop*. Kode sumber fungsi dapat dilihat pada Kode Sumber 4.12.

```

Public void drag(int x1, int y1, int x2, int y2,
IElement e) {
    if (e instanceof ControlPoint) {
        switch (((ControlPoint) e).getCode()) {
            case ControlPoint.TOP_LEFT:
                setBoundary(x2, y2, getX() + getWidth(),
                getY() + getHeight());
                break;
            case ControlPoint.TOP_MIDDLE:
                setBoundary(getX(), y2, getX() + getWidth(),
                getY() + getHeight());
                break;
            case ControlPoint.TOP_RIGHT:
                setBoundary(x2, y2, getX(), getY() +
                getHeight());
                break;
            case ControlPoint.MIDDLE_RIGHT:

```

```

        setBoundary(x2, getY() + getHeight(),
        getX(), getY());
        break;
        case ControlPoint.BOTTOM_RIGHT:
            setBoundary(x2, y2, getX(), getY());
            break;
        case ControlPoint.BOTTOM_MIDDLE:
            setBoundary(getX() + getWidth(), y2, getX(),
            getY());
            break;
        case ControlPoint.BOTTOM_LEFT:
            setBoundary(x2, y2, getX() + getWidth(),
            getY());
            break;
        case ControlPoint.MIDDLE_LEFT:
            setBoundary(x2, getY(), getX() + getWidth(),
            getY() + getHeight());
            break;
        }
    }
    super.drag(x1, y1, x2, y2, e);
}

```

Kode Sumber 4.12 Kode Sumber Mengubah Ukuran Gambar

4.3.2.4 Fungsi Merotasi Gambar

Fungsi merotasi gambar berada pada kelas statis Rotator pada fungsi *rotate*. Fungsi ini merotasi *ImageData* yang dimiliki kelas *Picture*. Kode sumber fungsi *rotate* dapat dilihat pada Kode Sumber 4.13.

```

Public static ImageData rotate(ImageData srcData, int
direction) {
int bytesPerPixel = srcData.bytesPerLine / srcData.width;
int destBytesPerLine = (direction == SWT.DOWN) ?
srcData.width * bytesPerPixel : srcData.height
* bytesPerPixel;
byte[] newData = new byte[(direction == SWT.DOWN) ?
srcData.height * destBytesPerLine
: srcData.width * destBytesPerLine];

```

```

int width = 0, height = 0;
for (int srcY = 0; srcY < srcData.height; srcY++) {
    for (int srcX = 0; srcX < srcData.width; srcX++){
        int destX = 0, destY = 0, destIndex = 0,
        srcIndex = 0;
        switch (direction) {
            case SWT.LEFT:
                destX = srcY;
                destY = srcData.width - srcX - 1;
                width = srcData.height;
                height = srcData.width;
                break;
            case SWT.RIGHT:
                destX = srcData.height - srcY - 1;
                destY = srcX;
                width = srcData.height;
                height = srcData.width;
                break;
            case SWT.DOWN:
                destX = srcData.width - srcX - 1;
                destY = srcData.height - srcY - 1;
                width = srcData.width;
                height = srcData.height;
                break;
        }
        destIndex = (destY * destBytesPerLine) +
        (destX * bytesPerPixel);
        srcIndex = (srcY * srcData.bytesPerLine) +
        (srcX * bytesPerPixel);
        System.arraycopy(srcData.data, srcIndex,
        newData, destIndex, bytesPerPixel);
    }
}
return new ImageData(width, height,
srcData.depth, srcData.palette, destBytesPerLine,
newData);
}

```

Kode Sumber 4.13 Kode Sumber Merotasi Gambar

Setelah gambar asli dirotasi maka kelas Picture yang merupakan representasi dari gambar yang dimuat akan

mengubah atribut `ImageData` yang dimilikinya dan menggambar ulang pada halaman aktif.

4.3.2.5 Fungsi Memanipulasi Penambahan Teks

Mekasnisme penambahan teks merupakan mekanisme instansiasi dari kelas `Paragraph`. Saat pengguna menambahkan teks pada halaman pengguna mengatur semua atribut yang dibutuhkan untuk menginstansiasi kelas `Paragraph`. Fungsi untuk menginstansiasi kelas `Paragraph` dapat dilihat pada Kode Sumber 4.14.

```
public Paragraph(int x, int y, int width, int height,
String text, Font font, Color color, String
fontStyle, int fontSize) {
    super(x, y, width, height);
    setText(text);
    setFont(font);
    setColor(color);
    setFontStyle(fontStyle);
    setFontSize(fontSize);
}
```

**Kode Sumber 4.14 Kode Sumber Instansiasi Kelas
Paragraph**

4.4 Implementasi Pengolahan Citra

Pengolahan citra yang diimplementasi pada aplikasi menggunakan pustaka `OpenCV` versi 2.4.12. Seluruh kebutuhan untuk pengolahan citra telah disediakan dalam `OpenCV`. Kelas dan fungsi yang digunakan seperti yang dijelaskan pada Bab 2 subbab `OpenCV`. Masing-masing jenis pengolahan citra memiliki algoritma yang berbeda-beda. Algoritma pengolahan akan dijelaskan pada masing-masing subbab.

4.4.1 Peningkatan Kontras

Proses peningkatan kontras pada OpenCV menggunakan fungsi *equalizeHist* yang terdapat pada kelas *Imgproc*. Potongan kode sumber untuk meningkatkan kontras citra dapat dilihat pada Kode Sumber 4.15.

```
System.loadLibrary(Core.NATIVE_LIBRARY_NAME);
Mat source = Highgui.imread(editWindow.getImagePath(),
Highgui.CV_LOAD_IMAGE_COLOR);

List<Mat> bgr = new ArrayList<>();
List<Mat> dest = new ArrayList<>();

Mat destination = new Mat(source.rows(), source.cols(),
source.type());
Core.split(source, bgr);
Core.split(source, dest);
Imgproc.equalizeHist(bgr.get(1), dest.get(1));
Core.merge(dest, destination);
Highgui.imwrite("temp/contrast.jpg", destination);
ImageData imgData = new ImageData("temp/contrast.jpg");

editWindow.setImageData(imgData);
editWindow.getImageEditor().reCallPicture(imgData);
editWindow.setEffect("contrast");
```

Kode Sumber 4.15 Kode Sumber Peningkatan Kontras

Kelas *Highgui* digunakan untuk membaca file yang terpilih dan disimpan ke variabel bertipe *Mat*. *Mat* asal nantinya akan dipecah menjadi 3 lapis warna pembentuk gambar yaitu *red*, *green*, dan *blue*. Indeks 1 merupakan kode untuk layer warna hijau yang akan dijalankan fungsi *equalizeHist* untuk meningkatkan kontras hijau dan selanjutnya digabungkan dengan layer lain sehingga membentuk suatu citra olahan baru yang kontrasnya meningkat.

4.4.2 Peningkatan Pencahayaan

Proses peningkatan pencahayaan pada OpenCV menggunakan fungsi dari kelas `Mat` yaitu `convertTo`. Fungsi tersebut digunakan dengan jenis kedalaman yang sama seperti `Mat input`, faktor skala 1, dan delta bernilai 31. Potongan kode sumber untuk meningkatkan pencahayaan citra dapat dilihat pada Kode Sumber 4.16.

```
System.LoadLibrary( Core.NATIVE_LIBRARY_NAME );
Mat source =
Highgui.imread(editWindow.getImagePath(),Highgui.CV_LOAD
_IMAGE_COLOR);

Mat destination = new Mat(source.rows(),source.cols(),
source.type());

source.convertTo(destination, -1, alpha, beta);
Highgui.imwrite("temp/brightness.jpg", destination);

ImageData imgData = new
ImageData("temp/brightness.jpg");
editWindow.setImageData(imgData);

editWindow.getImageEditor().reCallPicture(imgData);
editWindow.setEffect("brightness");
```

**Kode Sumber 4.16 Kode Sumber Peningkatan
Pencahayaan**

4.4.3 Peningkatan Ketajaman

Proses peningkatan ketajaman citra pada OpenCV menggunakan fungsi `addWeighted` pada kelas `Core` milik OpenCV. Selain itu juga menggunakan fungsi `gaussianBlur` milik kelas `Imgproc`. Potongan kode sumber untuk meningkatkan ketajaman citra dapat dilihat pada Kode Sumber 4.17.

```
System.LoadLibrary( Core.NATIVE_LIBRARY_NAME );
Mat source = Highgui.imread(editWindow.getImagePath(),
Highgui.CV_LOAD_IMAGE_COLOR);

Mat destination = new
Mat(source.rows(),source.cols(),source.type());
Imgproc.GaussianBlur(source, destination, new Size(0,0), 10);
Core.addWeighted(source, 1.5, destination, -0.5, 0, destination);
Highgui.imwrite("temp/sharp.jpg", destination);

ImageData imgData = new ImageData("temp/sharp.jpg");
editWindow.setImageData(imgData);
editWindow.getImageEditor().reCallPicture(imgData);
editWindow.setEffect("sharp");
```

Kode Sumber 4.17 Kode Sumber Peningkatan Ketajaman

BAB V

PENGUJIAN DAN EVALUASI

Pada bab ini akan dijelaskan proses pengujian aplikasi untuk tiap fitur yang disediakan. Masing-masing fitur akan dijelaskan bagaimana pengujiannya dan hasil akhir pengujian fitur. Tujuan akhir pengujian ini adalah untuk menilai apakah aplikasi sudah dapat merancang album foto sesuai dengan keinginan pengguna. Oleh karena itu, pada akhir bab ini akan dilakukan simulasi pembuatan rancangan album foto untuk pengujian akhir.

5.1 Lingkungan Pengujian

Lingkungan pengujian adalah lingkungan, baik perangkat keras maupun perangkat lunak tempat pengujian skenario dilakukan. Pengujian fitur-fitur dilakukan pada skenario dengan spesifikasi pada Tabel 5.1.

Tabel 5.1 Spesifikasi Sistem Pengujian

Spesifikasi	Deskripsi
CPU	Intel ® Core ™ i5-5200 CPU @ 2.20 GHz
RAM	4.00 GB
Sistem Operasi	Windows 10 Enterprise 64 bit

Pengujian dilakukan pada tanggal 6 Juni 2015 pukul 15.29-selesai bertempat di Laboratorium Dasar dan Terapan Komputer dengan pengujian fitur-fitur yang tersedia pada Tabel 5.2.

Tabel 5.2 Daftar Testing Fitur Aplikasi

No.	Fitur
1	Membuat proyek baru
2	Menambah halaman
3	Menghapus halaman
4	Menambah gambar
5	Melakukan peningkatan kualitas citra
6	Menambahkan bingkai
7	Mengubah posisi gambar
8	Mengubah ukuran gambar
9	Melakukan rotasi 90° pada gambar
10	Menambahkan teks
11	Menentukan ukuran teks
12	Menentukan jenis teks
13	Menentukan warna teks

5.2 Skenario Pengujian

Pada subbab ini dijelaskan mengenai pengujian yang digunakan. Seperti yang sudah dijelaskan pada pengantar di bab 5, bahwa pengujian merupakan pengujian fungsionalitas yang menggunakan metode *blackbox*. Metode *blackbox* merupakan metode dimana pengujian ditekankan pada pola *input* dan *output* yang sesuai dengan. Pengujian dilakukan dengan mengacu kasus penggunaan yang dijelaskan pada bab 3, analisis dan perancangan.

5.2.1 Kasus Pengujian Membuat Proyek Baru

Pada kasus uji ini pengguna akan mencoba membuat proyek baru dari awal aplikasi berjalan. Detail pengujian berupa skenario, kondisi awal dan kondisi akhir yang terdapat pada Tabel 5.3.

Tabel 5.3 Kasus Uji Membuat Proyek Baru

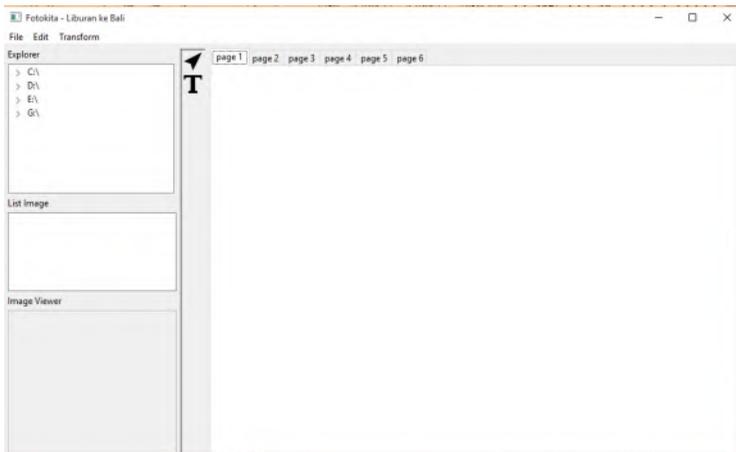
ID	UJ-001
Kasus Penggunaan	Membuat proyek baru.
Nama	Pengujian pembuatan proyek baru.
Tujuan Pengujian	Menguji inisiasi proyek dari awal.
Kondisi Awal	Proyek baru belum diinisiasi.
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih menu <i>File</i>. 2. Pengguna memilih <i>New Project Window</i> 3. Pengguna mengisi data yang dibutuhkan untuk membuat proyek baru.
Hasil yang Diharapkan	Proyek baru berhasil dibuat berdasar data yang diberikan pengguna.
Hasil yang Didapat	Proyek baru berhasil dibuat berdasar data yang diberikan pengguna.
Kondisi Akhir	Proyek baru berhasil dibuat.

Bentuk pengisian data dilakukan dengan masukan pada skenario berupa teks dan *dropdown*. Hasil dari skenario adalah proyek baru telah dibuat dan memiliki halaman yang dapat diisi oleh gambar dan teks dari pengguna. Tampilan pembuatan proyek baru dan kondisi akhir dari skenario terdapat pada Gambar 5.1 dan Gambar 5.2.

Gambar 5.1 Tampilan Pembuatan Proyek Baru

5.2.2 Kasus Pengujian Menambah Halaman

Pada kasus uji ini, pengguna dapat menambah jumlah halaman pada proyek apabila dirasa kurang. Halaman yang ditambahkan adalah halaman paling belakang. Detail pengujian berupa skenario, kondisi awal, dan kondisi akhir terdapat pada Tabel 5.4.



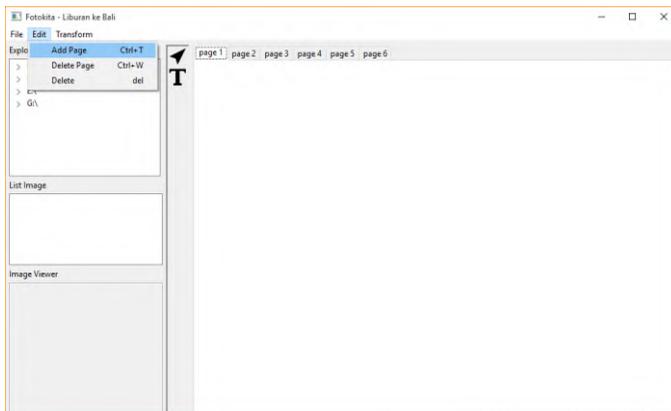
Gambar 5.2 Tampilan Akhir pembuatan Proyek Baru

Tabel 5.4 Kasus Uji Menambah Halaman

ID	UJ-002
Kasus Penggunaan	Menambah halaman
Nama	Pengujian penambahan halaman.
Tujuan Pengujian	Menguji fitur penambahan halaman pada proyek yang sedang aktif.
Kondisi Awal	Halaman proyek masih sesuai inisiasi awal.
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih menu <i>Edit</i>. 2. Pengguna memilih <i>Add Page</i>
Hasil yang Diharapkan	Halaman dari proyek bertambah satu buah halaman dibagian akhir.

Hasil yang Didapat	Halaman proyek bertambah satu halaman di bagan akhir.
Kondisi Akhir	Halaman berhasil ditambahkan.

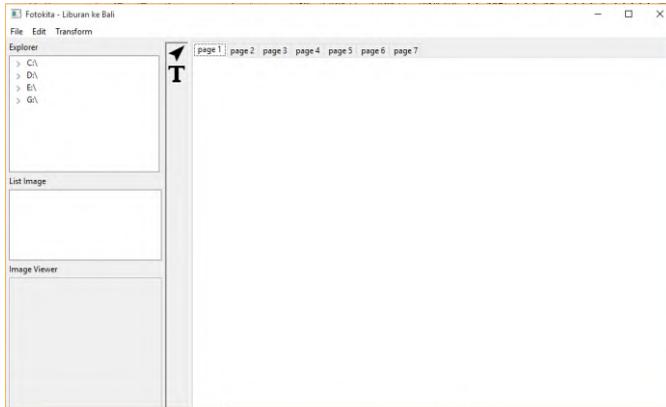
Alur skenario pengujian berupa kondisi awal dan kondisi akhir terdapat pada Gambar 5.3 dan Gambar 5.4.



Gambar 5.3 Kondisi Awal Penambahan Halaman

5.2.3 Kasus Pengujian Menghapus Halaman

Pada subbab ini akan menjelaskan kasus uji untuk penghapusan halaman yang sedang aktif pada proyek. Kondisi awal dari kasus uji ini seperti kondisi akhir pada kasus uji UJ-002. Detail pengujian berupa kondisi awal dan kondisi akhir yang terdapat pada Tabel 5.5.

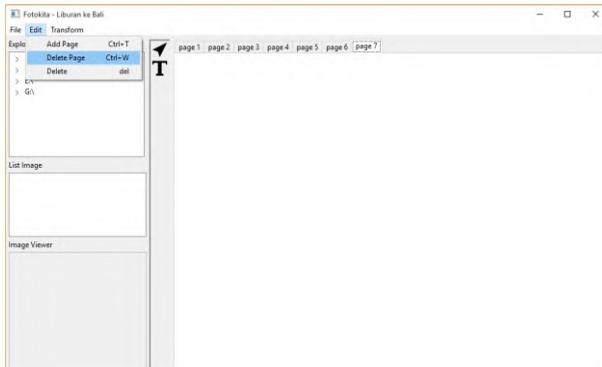


Gambar 5.4 Kondisi Akhir Penambahan Halaman

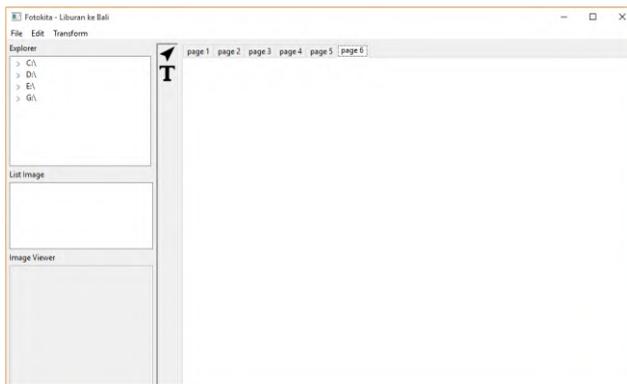
Tabel 5.5 Kasus Uji Menghapus Halaman

ID	UJ-003
Kasus Penggunaan	Menghapus halaman
Nama	Pengujian penghapusan halaman.
Tujuan Pengujian	Menguji fitur menghapus halaman pada proyek yang sedang aktif.
Kondisi Awal	Halaman proyek masih sesuai inisiasi awal.
Langkah Pengujian	1. Pengguna memilih menu <i>Delete Page</i> .
Hasil yang Diharapkan	Halaman dari proyek berkurang satu buah halaman pada halaman aktif.
Hasil yang Didapat	Halaman proyek berkurang satu halaman di halaman yang sedang aktif.
Kondisi Akhir	Halaman berhasil dihapus.

Halaman yang akan dihapus adalah halaman yang sedang aktif pada saat pengguna menekan *Delete Page*. Tampilan kondisi awal dan kondisi akhir dari skenario terdapat pada Gambar 5.5 dan Gambar 5.6.



Gambar 5.5 Kondisi Awal Penghapusan Halaman



Gambar 5.6 Kondisi Akhir Penghapusan Halaman

5.2.4 Kasus Pengujian Menambah Gambar pada Halaman

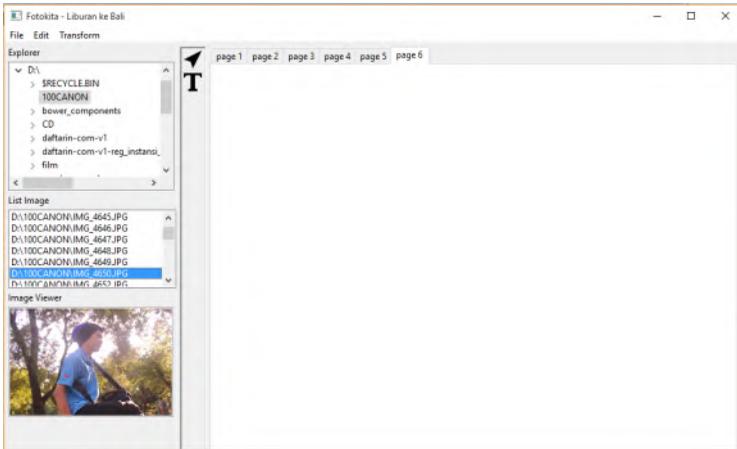
Pada subbab ini akan menjelaskan kasus uji untuk penambahan gambar pada halaman. Halaman akan secara otomatis ditambahkan pada halaman aktif proyek yang sedang

berjalan. Detail pengujian berupa skenario, kondisi awal, dan kondisi akhir terdapat pada Tabel 5.6

Tabel 5.6 Kasus Uji Menambah Gambar pada Halaman

ID	UJ-004
Kasus Penggunaan	Menambah gambar pada halaman
Nama	Pengujian penambahan gambar halaman.
Tujuan Pengujian	Menguji fitur penambahan gambar pada halaman yang sedang aktif.
Kondisi Awal	Gambar belum dimuat pada halaman
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih <i>directory</i> dari gambar yang ingin ditambahkan. 2. Pada <i>directory</i> yang dipilih pengguna memilih gambar yang tersedia pada <i>directory</i> tersebut dengan cara <i>double click</i>. 3. Setelah muncul tampilan penambahan gambar, pengguna menekan tombol <i>draw</i>.
Hasil yang Diharapkan	Gambar yang dipilih dapat dimuat dalam halaman yang sedang aktif.
Hasil yang Didapat	Gambar yang dipilih berhasil dimuat dalam halaman yang sedang aktif.
Kondisi Akhir	Gambar berhasil ditambahkan ke halaman.

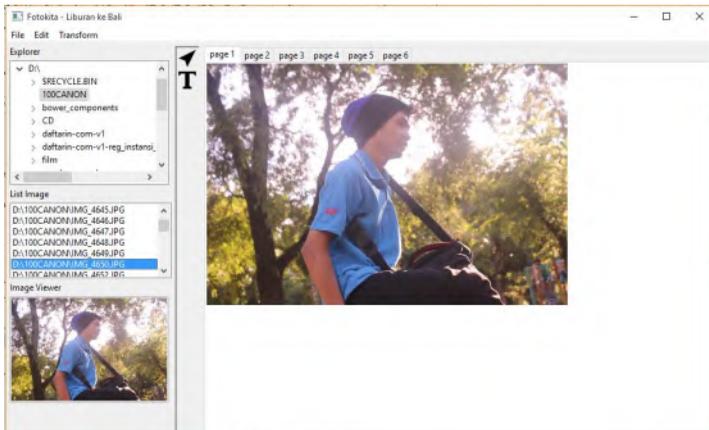
Tampilan kondisi awal dan akhir pada kasus uji ini dapat dilihat pada Gambar 5.7 dan Gambar 5.8.



Gambar 5.7 Kondisi Awal Penambahan Gambar

5.2.5 Kasus Pengujian Mengubah Posisi Gambar

Pada subbab ini akan menjelaskan kasus uji untuk mengubah posisi gambar yang sudah ditambahkan ke halaman. Detail pengujian berupa skenario, kondisi awal, dan kondisi akhir yang terdapat pada Tabel 5.7.

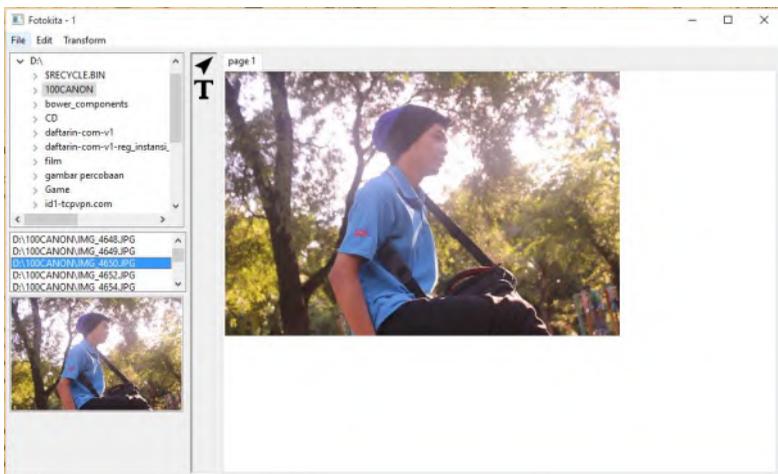


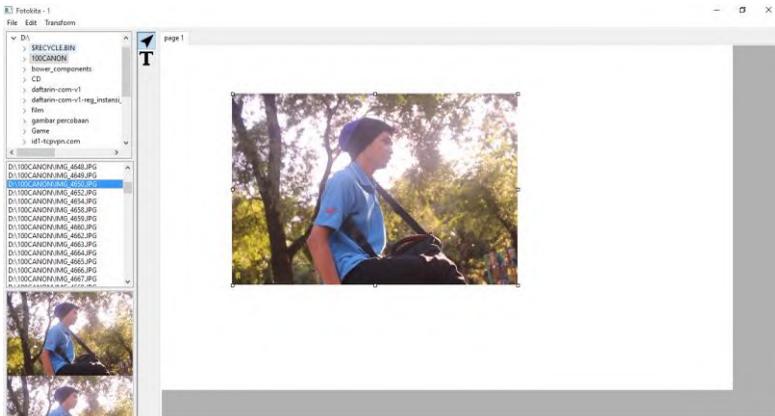
Gambar 5.8 Kondisi Akhir Penambahan Gambar

Tabel 5.7 Kasus Uji Mengubah Posisi Gambar

ID	UJ-005
Kasus Penggunaan	Mengubah posisi gambar
Nama	Pengujian perubahan posisi gambar pada halaman.
Tujuan Pengujian	Menguji fitur perubahan posisi gambar pada halaman yang sedang aktif.
Kondisi Awal	Posisi gambar masih terletak di pojok kiri atas halaman
Langkah Pengujian	1. Pengguna melakukan <i>drag and drop</i> pada gambar yang dipilih.
Hasil yang Diharapkan	Gambar yang dipilih dapat berubah posisi
Hasil yang Didapat	Gambar yang dipilih berubah posisi sesuai tujuan pengguna
Kondisi Akhir	Gambar berhasil diubah posisi

Tampilan kondisi awal dan akhir pada kasus uji ini dapat dilihat pada Gambar 5.9 dan Gambar 5.10.

**Gambar 5.9 Kondisi Awal Pengubahan Posisi Gambar**



Gambar 5.10 Kondisi Akhir Pengubahan Posisi Gambar

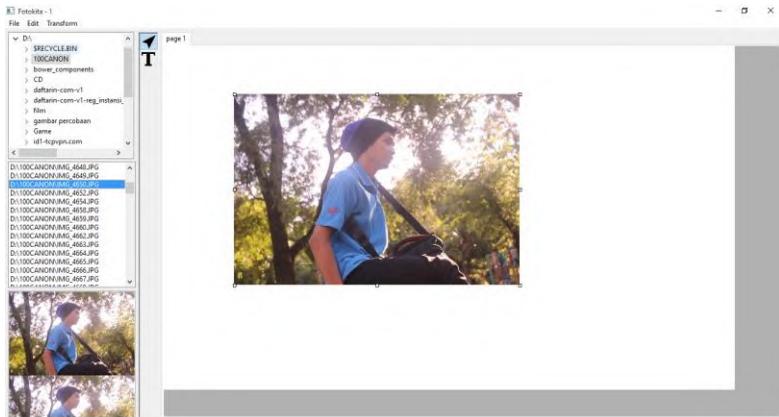
5.2.6 Kasus Pengujian Mengubah Ukuran Gambar

Pada subbab ini akan menjelaskan proses pengubahan ukuran gambar yang sudah dimuat dalam halaman. Penjelasan detail dari kasus uji ini dapat dilihat pada Tabel 5.8.

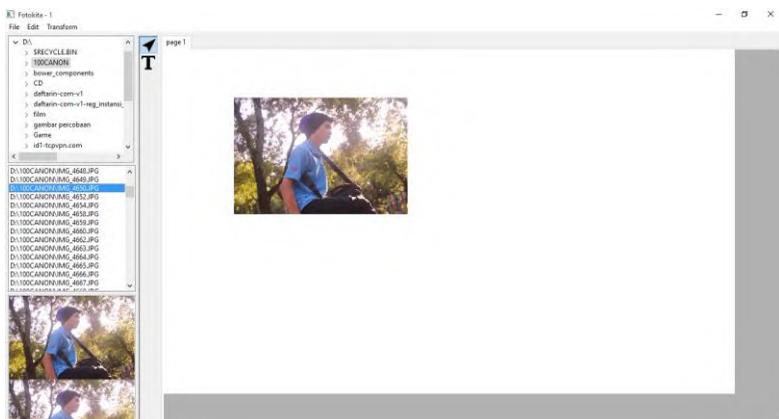
Tabel 5.8 Kasus Uji Mengubah Ukuran Gambar

ID	UJ-006
Kasus Penggunaan	Mengubah ukuran gambar
Nama	Pengujian perubahan ukuran gambar.
Tujuan Pengujian	Menguji fitur pengubahan ukuran gambar pada halaman yang sedang aktif.
Kondisi Awal	Ukuran gambar masih dalam ukuran normal
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih gambar yang akan diubah ukurannya 2. Pengguna melakukan <i>drag and drop</i> pada <i>control point</i> yang muncul pada gambar terpilih
Hasil yang Diharapkan	Gambar yang dipilih dapat diubah ukurannya
Hasil yang Didapat	Gambar yang dipilih berubah ukurannya
Kondisi Akhir	Gambar berhasil berubah ukurannya.

Kondisi awal dan akhir dari kasus uji mengubah ukuran gambar dapat dilihat pada Gambar 5.11 dan Gambar 5.12.



Gambar 5.11 Kondisi Awal Pengubahan Ukuran Gambar



Gambar 5.12 Kondisi Akhir Pengubahan Ukuran Gambar

5.2.7 Kasus Pengujian Menambah Bingkai pada Gambar

Pada subbab ini akan menjelaskan proses penambahan bingkai pada gambar yang akan ditambahkan ke dalam halaman. Penjelasan detail dari kasus uji ini dapat dilihat pada Tabel 5.9.

Tabel 5.9 Kasus Uji Menambah Bingkai pada Gambar

ID	UJ-007
Kasus Penggunaan	Menambah bingkai pada gambar
Nama	Pengujian penambahan bingkai pada gambar.
Tujuan Pengujian	Menguji fitur penambahan bingkai pada gambar yang dipilih.
Kondisi Awal	-
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih gambar yang akan ditambahkan ke halaman dan ditambahkan bingkai 2. Pengguna memilih jenis bingkai yang akan ditambahkan pada gambar 3. Pengguna menekan tombol <i>draw</i>
Hasil yang Diharapkan	Gambar yang dipilih ditambahkan pada halaman bersama bingkai yang dipilih
Hasil yang Didapat	Gambar yang dipilih ditambahkan bersama bingkai
Kondisi Akhir	Gambar berhasil berhasil ditambahkan bersama bingkai

Kondisi akhir dari kasus uji menambah bingkai pada gambar dapat dilihat pada Gambar 5.13.



Gambar 5.13 Kondisi Akhir Menambah Bingkai pada Gambar

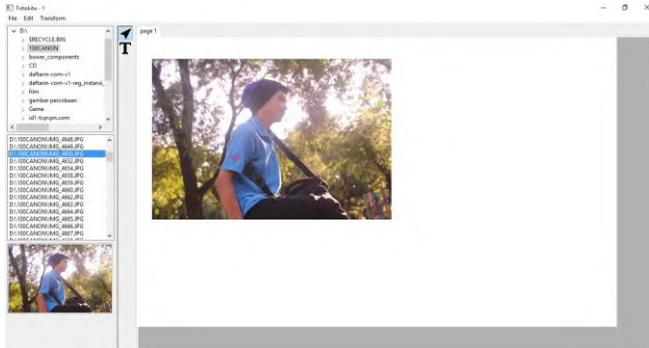
5.2.8 Kasus Pengujian Melakukan Rotasi pada Gambar

Pada subbab ini akan menjelaskan proses rotasi pada gambar yang sudah dimuat dalam halaman. Penjelasan detail dari kasus uji ini dapat dilihat pada Tabel 5.10.

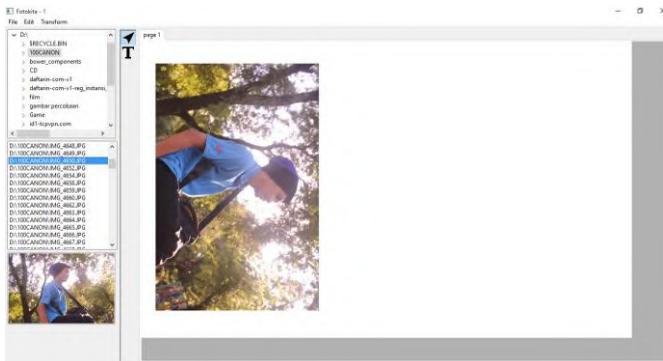
Tabel 5.10 Kasus Uji Melakukan Rotasi pada Gambar

ID	UJ-008
Kasus Penggunaan	Melakukan rotasi pada gambar
Nama	Pengujian rotasi pada gambar
Tujuan Pengujian	Menguji fitur rotasi pada gambar.
Kondisi Awal	Posisi gambar masih normal
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih gambar yang akan dirotasi pada halaman aktif. 2. Pengguna memilih menu <i>Transform</i> pada pilihan menu. 3. Pengguna memilih pilihan rotasi yang tersedia.
Hasil yang Diharapkan	Gambar yang dipilih akan berotasi sesuai dengan pilihan pengguna
Hasil yang Didapat	Gambar yang dipilih berotasi sesuai pilihan pengguna
Kondisi Akhir	Gambar berhasil dirotasi sebanyak 90° sesuai arah pilihan pengguna

Kondisi awal dan akhir dari kasus uji mengubah ukuran gambar dapat dilihat pada Gambar 5.14 dan Gambar 5.15.



Gambar 5.14 Kondisi Awal Melakukan Rotasi pada Gambar



Gambar 5.15 Kondisi Akhir Melakukan Rotasi pada Gambar

5.2.9 Kasus Pengujian Melakukan Peningkatan Kualitas Citra

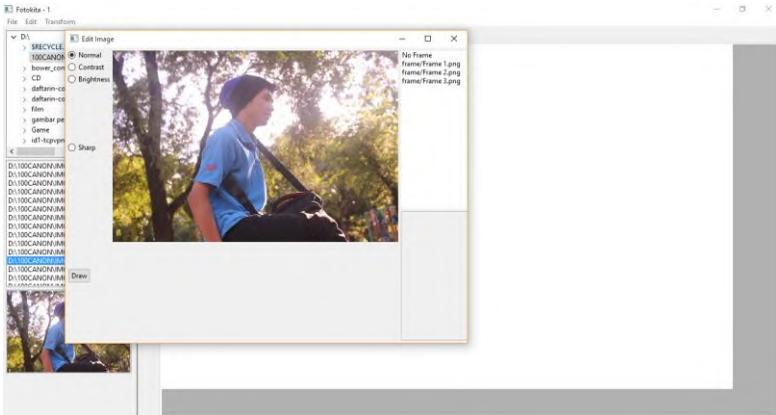
Pada subbab ini akan menjelaskan proses peningkatan kualitas citra pada citra yang dipilih. Peningkatan kualitas yang disediakan adalah peningkatan kontras, peningkatan kecerahan,

dan peningkatan ketajaman. Penjelasan dari kasus uji ini dapat dilihat pada Tabel 5.11Tabel 5.11.

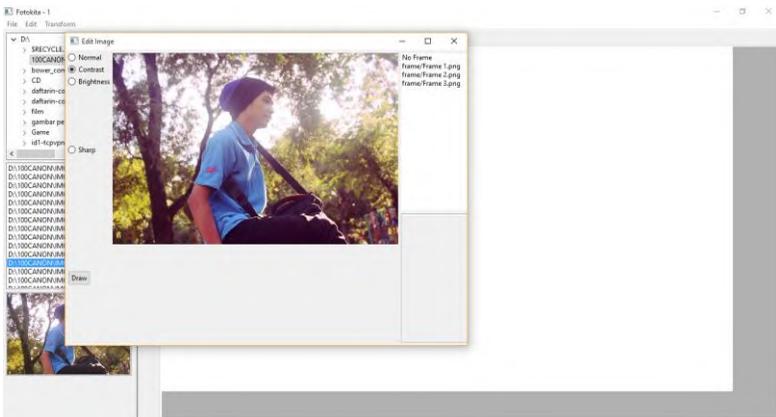
Tabel 5.11 Kasus Uji Melakukan Peningkatan Kualitas Citra

ID	UJ-009
Kasus Penggunaan	Melakukan peningkatan kualitas citra
Nama	Pengujian rotasi pada gambar
Tujuan Pengujian	Menguji fitur peningkatan kualitas citra pada citra yang akan ditambahkan.
Kondisi Awal	-
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih citra yang akan ditambahkan ke halaman 2. Pengguna memilih pilihan peningkatan kualitas citra yang disediakan 3. Pengguna menekan tombol <i>draw</i>
Hasil yang Diharapkan	Gambar yang dipilih akan berubah sesuai dengan peningkatan kualitas citra yang dipilih
Hasil yang Didapat	Gambar yang dipilih berubah sesuai dengan peningkatan kualitas citra yang dipilih
Kondisi Akhir	Gambar berhasil diubah kualitasnya

Skenario awal sampai dengan gambar dimuat dari kasus uji ini akan dijelaskan pada Gambar 5.16 dan Gambar 5.17. Pada Gambar 5.20 akan terlihat perbedaan dari citra yang sudah diolah dan citra yang belum diolah. Citra sebelah kiri atas merupakan citra asli sedangkan sebelah kanan atas merupakan citra hasil olahan dengan peningkatan kontras, citra sebelah kiri bawah merupakan hasil olahan dengan peningkatan pencahayaan dan kanan bawah merupakan citra hasil olahan peningkatan ketajaman.



Gambar 5.16 Kondisi Citra Normal



Gambar 5.17 Kondisi Citra dengan Peningkatan Kontras



Gambar 5.20 Perbedaan Citra Asli dan Citra Olahan

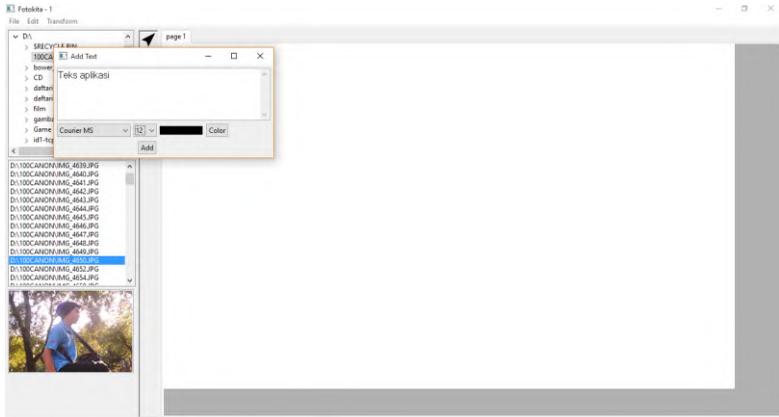
5.2.10 Kasus Pengujian Menambah Teks pada Halaman

Pada subbab ini akan menjelaskan tentang proses penambahan teks pada halaman yang sedang aktif. Penjelasan dari kasus uji ini dapat dilihat pada Tabel 5.12.

Tabel 5.12 Kasus Uji Menambah Teks pada Halaman

ID	UJ-010
Kasus Penggunaan	Menambah teks pada halaman
Nama	Pengujian penambahan teks pada halaman
Tujuan Pengujian	Menguji fitur penambahan teks pada halaman
Kondisi Awal	Teks belum ditambahkan ke halaman
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih <i>text tool</i> pada <i>toolbox</i> 2. Pengguna memilih posisi pada halaman tempat ditambakkannya teks 3. Pengguna memasukkan teks yang ingin ditambahkan pada tampilan yang muncul 4. Pengguna menekan tombol <i>add</i>
Hasil yang Diharapkan	Teks yang ditambahkan berhasil dimuat pada halaman
Hasil yang Didapat	Teks berhasil dimuat pada halaman
Kondisi Akhir	Tek berhasil dimuat pada halaman

Kondisi awal dan akhir dari kasus uji menambah teks pada gambar dapat dilihat pada Gambar 5.21 dan Gambar 5.22.



Gambar 5.21 Kondisi Awal Menambah Teks pada Halaman



Gambar 5.22 Kondisi Akhir Menambah Teks pada Halaman

5.2.11 Kasus Pengujian Menentukan Ukuran Teks

Pada subbab ini akan menjelaskan tentang mengubah ukuran teks yang akan dimuat pada halaman aktif. Penjelasan dari kasus uji ini dapat dilihat pada Tabel 5.13.

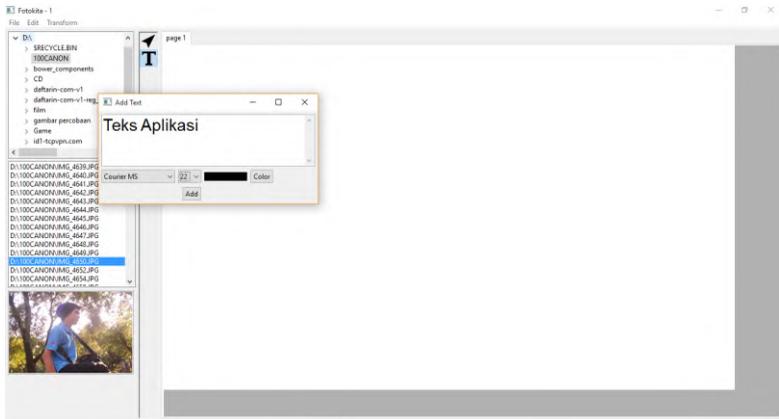
Tabel 5.13 Kasus Uji Menentukan Ukuran Teks

ID	UJ-011
Kasus Penggunaan	Menentukan ukuran teks
Nama	Pengujian mengubah ukuran teks
Tujuan Pengujian	Menguji fitur perubahan ukuran teks
Kondisi Awal	Teks belum ditambahkan ke halaman
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih <i>text tool</i> pada <i>toolbox</i> 2. Pengguna memilih posisi pada halaman tempat ditambakkannya teks 3. Pengguna memasukkan teks yang ingin ditambahkan pada tampilan yang muncul 4. Pengguna memilih ukuran teks pada tampilan yang muncul 5. Pengguna menekan tombol <i>add</i>
Hasil yang Diharapkan	Teks yang ditambahkan dimuat pada halaman sesuai dengan ukuran teks yang ditentukan
Hasil yang Didapat	Teks yang ditambahkan berhasil dimuat pada halaman sesuai dengan ukuran teks yang ditentukan
Kondisi Akhir	Tek berhasil dimuat pada halaman dengan ukuran yang ditentukan

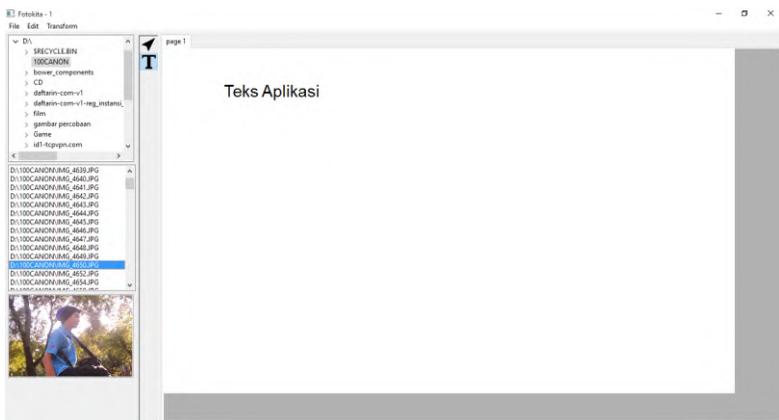
Kondisi awal dan akhir dari kasus uji menambah teks pada gambar dapat dilihat pada Gambar 5.23 dan Gambar 5.24 Gambar 5.24.

5.2.12 Kasus Pengujian Menentukan Jenis Teks

Pada subbab ini akan menjelaskan tentang mengubah jenis teks yang akan dimuat pada halaman aktif. Penjelasan dari kasus uji ini dapat dilihat pada Tabel 5.14.



Gambar 5.23 Kondisi Awal Menentukan Ukuran Teks



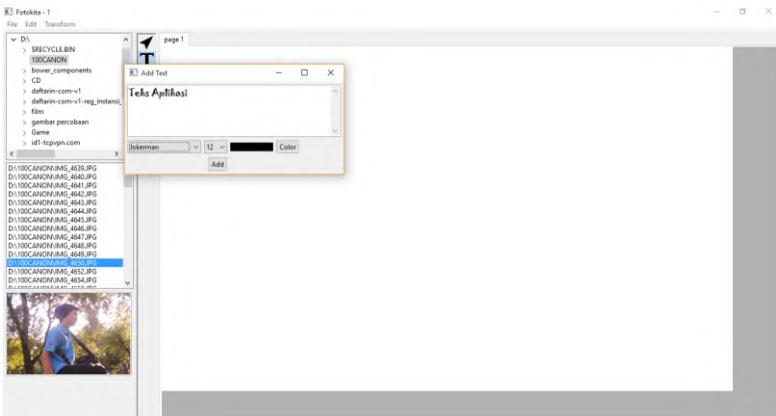
Gambar 5.24 Kondisi Akhir Menentukan Ukuran Teks

Tabel 5.14 Kasus Uji Menentukan Jenis Teks

ID	UJ-012
Kasus Penggunaan	Menentukan jenis teks
Nama	Pengujian mengubah jenis teks
Tujuan Pengujian	Menguji fitur perubahan jenis teks
Kondisi Awal	Teks belum ditambahkan ke halaman
Langkah Pengujian	1. Pengguna memilih <i>text tool</i> pada <i>toolbox</i>

	<ol style="list-style-type: none"> 2. Pengguna memilih posisi pada halaman tempat ditambahkan teks 3. Pengguna memasukkan teks yang ingin ditambahkan pada tampilan yang muncul 4. Pengguna memilih jenis teks pada tampilan yang muncul 5. Pengguna menekan tombol <i>add</i>
Hasil yang Diharapkan	Teks yang ditambahkan dimuat pada halaman sesuai dengan jenis teks yang ditentukan
Hasil yang Didapat	Teks yang ditambahkan berhasil dimuat pada halaman sesuai dengan jenis teks yang ditentukan
Kondisi Akhir	Tek berhasil dimuat pada halaman dengan jenis yang ditentukan

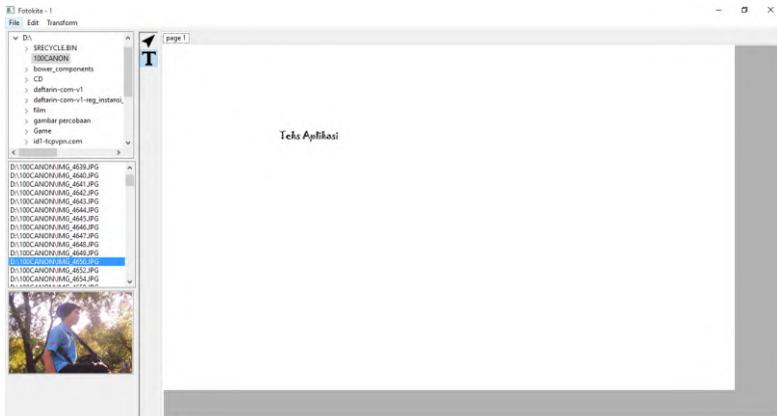
Kondisi awal dan akhir dari kasus uji menambah teks pada gambar dapat dilihat pada Gambar 5.25 dan Gambar 5.26.



Gambar 5.25 Kondisi Awal Menentukan Jenis Teks

5.2.13 Kasus Pengujian Menentukan Warna Teks

Pada subbab ini akan menjelaskan tentang mengubah warna teks yang akan dimuat pada halaman aktif. Penjelasan dari kasus uji ini dapat dilihat pada Tabel 5.15.

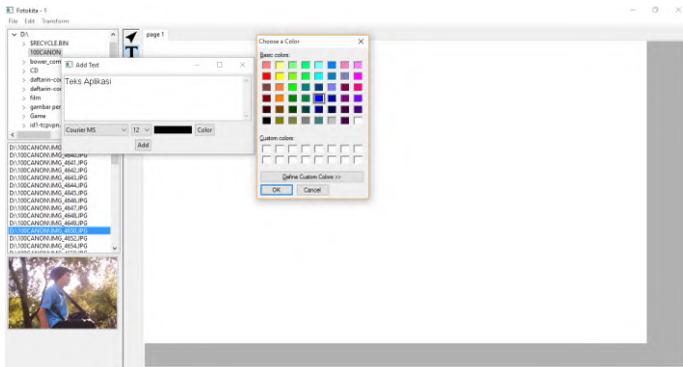


Gambar 5.26 Kondisi Akhir Menentukan Jenis Teks

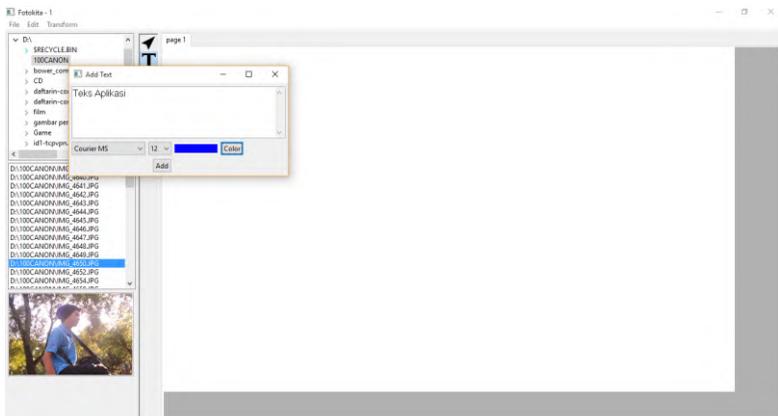
Tabel 5.15 Kasus Uji Menentukan Warna Teks

ID	UJ-013
Kasus Penggunaan	Menentukan warna teks
Nama	Pengujian mengubah warna teks
Tujuan Pengujian	Menguji fitur perubahan warna teks
Kondisi Awal	Teks belum ditambahkan ke halaman
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih <i>text tool</i> pada <i>toolbox</i> 2. Pengguna memilih posisi pada halaman tempat ditambakkannya teks 3. Pengguna memasukkan teks yang ingin ditambahkan pada tampilan yang muncul 4. Pengguna menekan tombol <i>color</i> 5. Pengguna memilih warna dari pilihan yang ada 6. Pengguna menekan tombol <i>OK</i> 7. Pengguna menekan tombol <i>add</i>
Hasil yang Diharapkan	Teks yang ditambahkan dimuat pada halaman sesuai dengan warna teks yang ditentukan
Hasil yang Didapat	Teks yang ditambahkan berhasil dimuat pada halaman sesuai dengan warna teks yang ditentukan
Kondisi Akhir	Tek berhasil dimuat pada halaman dengan warna yang ditentukan

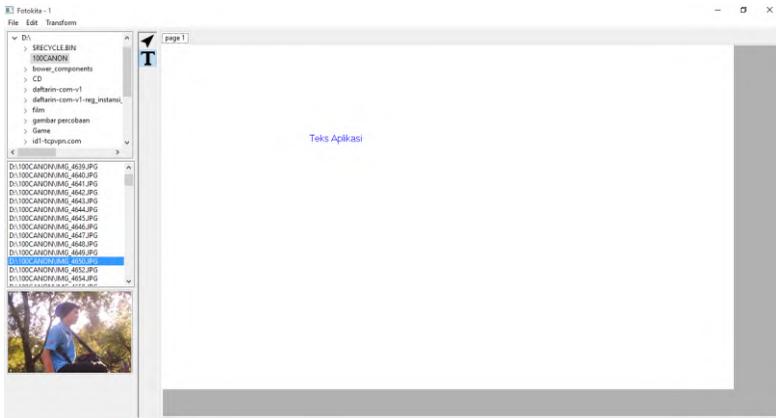
Kondisi awal sampai akhir dari kasus uji menambah teks pada gambar dapat dilihat pada Gambar 5.27, Gambar 5.28, dan Gambar 5.29.



Gambar 5.27 Kondisi Awal Menentukan Warna Teks



Gambar 5.28 Kondisi Setelah Warna dipilih



Gambar 5.29 Kondisi Akhir Menentukan Warna Teks

5.3 Pengujian Pembuatan Album

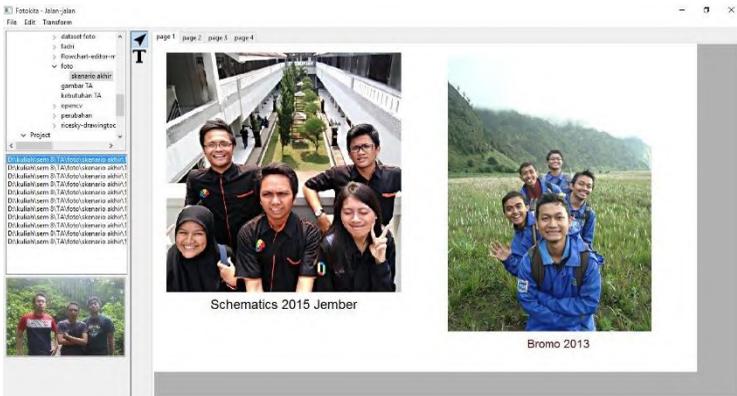
Pada bab ini akan diujicobakan skenario pembuatan album foto berdasarkan keinginan pengguna. Data pengguna dapat dilihat pada Tabel 5.16. Dalam skenario ini pengguna juga diminta untuk menggunakan semua fitur yang disediakan aplikasi guna memastikan fitur berjalan dengan baik.

Dalam bab ini juga akan ditampilkan hasil akhir dari rancangan album foto yang dibuat oleh pengguna. Hasil akhir tersebut nantinya akan menjadi tolok ukur keberhasilan aplikasi apakah sudah dapat membuat rancangan album foto sesuai dengan keinginan pengguna. Hasil akhir dari rancangan album foto pengguna dapat dilihat pada Gambar 5.30, Gambar 5.31, Gambar 5.32, dan Gambar 5.33.

Tabel 5.16 Biodata Pengguna

Nama	Fadrian Merdianto
Tempat lahir	Jember
Tanggal lahir	13 Desember 1993
Usia	22
Jenis kelamin	Laki-laki

Pekerjaan	Mahasiswa
Pendidikan 1. SD 2. SMP 3. SMA 4. Kuliah	1. SDN Kedungsari 8 2. SMPN 2 Jember 3. SMAN 1 Jember 4. Institut Teknologi Sepuluh Nopember (Teknik Informatika)



Gambar 5.30 Hasil Akhir Pengujian Halaman 1



Gambar 5.31 Hasil Akhir Pengujian Halaman 2



Gambar 5.32 Hasil Akhir Pengujian Halaman 3



Gambar 5.33 Hasil Akhir Pengujian Halaman 4

5.4 Evaluasi Pengujian

Pada bab ini akan dijelaskan bagaimana hasil dari pengujian aplikasi dan akan disertakan dalam bentuk tabel. Tabel evaluasi dan pengujian dapat dilihat pada Tabel 5.17

Tabel 5.17 Tabel Evaluasi

Kode	Kasus Penggunaan	Hasil Pengujian
UJ-001	Membuat proyek baru	Berhasil
UJ-002	Menambah halaman	Berhasil
UJ-003	Menghapus halaman	Berhasil
UJ-004	Menambah gambar pada halaman	Berhasil
UJ-005	Mengubah posisi gambar	Berhasil
UJ-006	Mengubah ukuran gambar	Berhasil
UJ-007	Menambah bingkai pada gambar	Berhasil
UJ-008	Melakukan rotasi pada gambar	Berhasil
UJ-009	Melakukan peningkatan kualitas citra	Berhasil
UJ-010	Menambah teks pada halaman	Berhasil
UJ-011	Menentukan ukuran teks	Berhasil
UJ-012	Menentukan jenis teks	Berhasil
UJ-013	Menentukan warna teks	Berhasil

[Halaman ini sengaja dikosongkan]

BAB VI

KESIMPULAN DAN SARAN

Bab ini membahas mengenai kesimpulan yang dapat diambil dari hasil uji coba yang telah dilakukan sebagai jawaban dari rumusan masalah yang dikemukakan. Selain kesimpulan, juga terdapat saran yang ditujukan untuk pengembangan penelitian lebih lanjut.

Melalui pengujian dan evaluasi aplikasi dapat diambil beberapa kesimpulan dan saran yang nantinya bisa dijadikan acuan pengembangan aplikasi untuk kedepannya. Beberapa permasalahan yang diangkat dalam Bab 1 juga dapat diselesaikan melalui proses yang sudah dijelaskan pada Bab 4 dan diujicobakan pada Bab 5.

6.1 Kesimpulan

Dari hasil uji coba yang telah dilakukan terhadap pembuatan rancangan album foto, dapat diambil kesimpulan sebagai berikut:

1. Aplikasi dapat membantu pengguna merancang album foto berdasarkan fitur yang tersedia.
2. Untuk pemberian *layout* pada gambar agar dapat dipindahkan sesuai dengan keinginan merupakan implementasi dari kelas abstrak *TwoDimensional* yang memiliki fungsi-fungsi pergeseran objek.
3. Proses penyesuaian ukuran gambar agar tidak berlebihan terhadap *editor* dapat dilakukan dengan mengubah ukuran dengan fungsi *scaledTo* milik kelas *ImageData* pada Java SWT.
4. Gambar yang dipilih dapat ditingkatkan kualitasnya melalui peningkatan kualitas citra menggunakan fungsi bawaan pada pustaka *OpenCV* bergantung dengan jenis peningkatan kualitas yang diinginkan.

5. Peningkatan kontras pada gambar merupakan implementasi fungsi *equalizeHist* pada OpenCV untuk *channel* warna hijau pada citra yang dipilih kemudian menyatukan *channel* yang sudah diolah dengan *channel* merah dan biru yang asli.
6. Peningkatan pencahayaan pada gambar merupakan implementasi fungsi *convertTo* milik kelas Mat untuk memanipulasi tingkat kecerahan citra.
7. Peningkatan ketajaman pada gambar merupakan implementasi fungsi *GaussianBlur* milik kelas statis *Imgproc* sehingga gambar menjadi lebih tajam.
8. Aplikasi sudah memenuhi seluruh fitur yang tersedia dengan melakukan pengujian fungsionalitas pada tanggal 6 Juni 2016 di Laboratorium Dasar dan Terapan Komputasi.

6.2 Saran

Saran yang diberikan untuk pengembangan aplikasi ini adalah:

1. Fitur dapat diperbanyak lagi untuk pengolahan citra.
2. Latar belakang dari halaman juga dapat ditambahkan untuk memperindah album.
3. Tampilan antarmuka pengguna bisa lebih diperbaiki karena terlalu sederhana.
4. Rotasi berdasarkan gerakan pointer dapat ditambahkan.

DAFTAR PUSTAKA

- [1] M. Scarpiano, S. Holder, S. Ng and L. Mihalkovic, SWT/JFace in Action, Greenwich: Manning Publication, 2005.
- [2] "<https://maleskoding.wordpress.com/2009/04/29/singleton/>," [Online]. Available: <https://maleskoding.wordpress.com/2009/04/29/singleton/>. [Accessed 12 06 2016].
- [3] "sourcemaking.com," [Online]. Available: https://sourcemaking.com/design_patterns/observer. [Accessed 29 06 2016].
- [4] "sourcemaking.com," [Online]. Available: https://sourcemaking.com/design_patterns/state. [Accessed 29 06 2016].
- [5] "engineersgarage.com," [Online]. Available: <http://www.engineersgarage.com/articles/image-processing-tutorial-applications>. [Accessed 29 06 2016].
- [6] "tutorialspoint.com," [Online]. Available: http://www.tutorialspoint.com/dip/histogram_equalization.htm. [Accessed 29 06 2016].
- [7] OpenCV, "OpenCV.org," [Online]. Available: <http://opencv.org/>. [Accessed 29 06 2016].

[Halaman ini sengaja dikosongkan]

BIODATA PENULIS



Mochammad Arief Ridwan lahir di kota Surabaya tanggal 31 Oktober 1994. Penulis merupakan anak ketiga dari tiga bersaudara.

Riwayat pendidikan yang telah dialami yaitu SDN Wedoro I (2000–2006), SMP Negeri 22 Surabaya (2006-2009), dan SMA Negeri 15 Surabaya (2009-2012). Saat ini beliau sedang menempuh pendidikan di Institut Teknologi Sepuluh Nopember di jurusan Teknik Informatika angkatan 2012.

Bidang minat yang digeluti penulis adalah rekayasa perangkat lunak. Penulis juga memiliki ketertarikan pada bidang basis data, desain web, dan pemrograman. Penulis juga pernah menjadi asisten dosen mengajar mata kuliah Teori Graf dan Otomata, Pemrograman Berorientasi Objek, dan Pemrograman Web.

Saat menjalani masa kuliah, penulis juga aktif dalam organisasi mahasiswa baik lingkup jurusan maupun lingkup institut. Penulis tergabung dalam Himpunan Mahasiswa Teknik Computer-Informatika (HMTC) serta unit kegiatan Paduan Suara Mahasiswa ITS (PSM ITS). Penulis dapat dihubungi melalui alamat surel aridwan05@gmail.com