



TUGAS AKHIR - K1141502

RANCANG BANGUN APLIKASI ABSENSI
PERKULIAHAN MAHASISWA DENGAN
PENGENALAN WAJAH

Muhammad Yusuf
NRP 5111100 703

Dosen Pembimbing
Dr.tech. Ir. R. V. Hari Ginardi, M. Sc.
Adhatas Solichah A., S. Kom., M. Sc.

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



FINAL PROJECT - KI141502

DESIGN AND IMPLEMENTATION OF CLASS ATTENDANCE USING FACE RECOGNITION

Muhammad Yusuf
NRP 5111100 703

Advisor
Dr.tech. Ir. R. V. Hari Ginardi, M. Sc.
Adhatus Solichah A., S. Kom., M. Sc.

DEPARTMENT OF INFORMATICS
Faculty of Information Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2016

LEMBAR PENGESAHAN

RANCANG BANGUN APLIKASI ABSENSI PERKULIAHAN MAHASISWA DENGAN PENGENALAN WAJAH

TUGAS AKHIR

Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Rumpun Mata Kuliah Manajemen Informasi
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh :
MUHAMMAD YUSUF
NRP : 5111 100 703

Disetujui oleh Dosen Pembimbing Tugas Akhir:

Dr.tech. Ir. R. V. Hari Ginardi, M.Sc.
NIP: 196505181992031005 (Pembimbing 1)

Adhatus Solichah A., S.Kom., M.Sc.
NIP: 198508262015042002 (Pembimbing 2)

SURABAYA
JULI, 2016

RANCANG BANGUN APLIKASI ABSENSI PERKULIAHAN MAHASISWA DENGAN PENGENALAN WAJAH

Nama Mahasiswa : Muhammad Yusuf
NRP : 51111000703
Jurusan : Teknik Informatika FTIf-ITS
Dosen Pembimbing 1 : Dr.tech. Ir. R. V. Hari Ginardi, M. Sc.
Dosen Pembimbing 2 : Adhatu Solichah A., S. Kom., M. Sc.

ABSTRAK

Proses absensi yang dilakukan secara manual dinilai kurang efektif karena terbukanya kesempatan melakukan kecurangan. Selain itu, proses rekapitulasi manual membutuhkan waktu yang lama. Sistem absensi dengan teknologi dapat diterapkan untuk membantu proses absensi dan rekapitulasi yang efektif. Pada tugas akhir ini, teknologi yang digunakan adalah sistem pengenalan wajah.

Pembuatan aplikasi absensi dengan pengenalan wajah ini menggunakan metode Eigenface untuk melakukan proses pengenalan wajah. Sedangkan data-data yang dibutuhkan sistem adalah data mata kuliah, dosen, jadwal, kelas, mahasiswa, dan dataset foto wajah yang disimpan dalam sistem manajemen relasional basis data.

Hasil dari aplikasi yang dibangun yaitu dapat mengelola data-data pada sistem, serta melakukan pencatatan dan perekapan data absensi. Proses absensi mahasiswa berhasil dilakukan pada kondisi pencahayaan yang bagus dan resolusi yang sama dengan kondisi foto wajah yang disimpan dalam basis data.

Kata kunci: Absensi, Mahasiswa, Pengenalan Wajah, Eigenface, Sistem Manajemen Basis Data Relasional

DESIGN AND IMPLEMENTATION OF CLASS ATTENDANCE USING FACE RECOGNITION

Student Name : Muhammad Yusuf
NRP : 51111000703
Major : Department of Informatics FTIf-ITS
Advisor 1 : Dr.tech. Ir. R. V. Hari Ginardi, M. Sc
Advisor 2 : Adhatus Solichah A., S. Kom., M. Sc.

ABSTRACT

Attendance and recapitulation process are done manually which considered less effective because it gives the opportunities to entrust of filling up the attendance and spent a long time to enter a lot of data. Attendance system with a technology can be applied to assist the attendance and recapitulation process to be more effective. In this study, the technology used is a face recognition system.

Construction of this attendance application with face recognition using Eigenface method that will be used in the process of face recognition. While datasets face images stored in a relational database management system.

Results of the application built to store all the data required and assist in attendance process. Attendance process of students successfully performed in the same as a condition of lighting and resolution of the face images stored in a database. This application can also show attendance data of students in a lecture so as to assist in recapitulating students' college.

Keywords: Attendance, Students, Face Recognition, Eigenface, Relational Database Management System

KATA PENGANTAR

Alhamdulillah rabbil'alamin, puji syukur kepada Allah Yang Maha Esa atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan tugas akhir yang berjudul

“Rancang Bangun Aplikasi Absensi Perkuliahan Mahasiswa dengan Pengenalan Wajah”

Harapan dari penulis semoga apa yang tertulis di dalam buku tugas akhir ini dapat bermanfaat bagi pengembangan ilmu pengetahuan saat ini, serta dapat memberikan kontribusi yang nyata.

Dalam pelaksanaan dan pembuatan tugas akhir ini tentunya sangat banyak bantuan yang penulis terima dari berbagai pihak, dengan hormat penulis ingin mengucapkan terima kasih sebesar-besarnya kepada:

1. Bapak Supadi, Ibu Muzayanah, Kakak Ismail dan Kakak Mohammad Sholeh beserta keluarga yang selalu memberikan do'a dan dukungan penuh untuk menyelesaikan tugas akhir ini.
2. Bapak Dr.tech. Ir. R. V. Hari Ginardi, M. Sc. dan Ibu Adhatus Solichah A., S. Kom., M. Sc. selaku dosen pembimbing yang telah bersedia meluangkan waktu untuk memberikan petunjuk selama proses pengerjaan tugas akhir ini.
3. Bapak Dr.Eng Darlis Herumurti, S. Kom., M. Kom. selaku Ketua Jurusan Teknik Informatika ITS dan Bapak Radityo Anggoro, S. Kom., M. Sc. selaku koordinator tugas akhir yang memberikan ruang diskusi untuk setiap kendala dalam pengerjaan tugas akhir.
4. Bapak dan Ibu dosen Jurusan Teknik Informatika ITS yang telah banyak memberikan ilmu dan bimbingan yang tak ternilai harganya bagi penulis.

5. Seluruh staf dan karyawan Teknik Informatika ITS yang banyak memberikan kelancaran administrasi akademik kepada penulis.
6. Kementerian Agama Republik Indonesia yang telah memberikan kesempatan kepada penulis untuk menuntut ilmu di Teknik Informatika ITS.
7. Bapak Dr. Agus Zainal Arifin S. Kom., M. Kom. selaku pembina CSS MoRA ITS yang selalu memberikan arahan, bimbingan, dan masukan pada setiap permasalahan penulis di ITS.
8. Sahabat-sahabat CSS MoRA ITS yang selalu siap mendorong dan memberikan bantuan ketika penulis mengalami kesulitan.
9. Sahabat-sahabat SATU MASA PBSB ITS 2011 yang telah memberikan banyak dukungan dan semangat kepada penulis.
10. Mahardhika Maulana, Bryan Alfadhori, dan Bahrul Halimi yang telah berbagi ilmu kepada penulis.
11. Teman-teman angkatan 2011 jurusan Teknik Informatika ITS yang telah menemani di kampus perjuangan ini atas saran, masukan, dan dukungan terhadap pengerjaan tugas akhir ini.
12. Serta semua pihak yang tidak dapat disebutkan disini yang telah banyak membantu penulis dalam penyusunan tugas akhir ini.

Penulis telah berusaha sebaik-baiknya dalam menyusun tugas akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan ke depannya.

Surabaya, Juni 2016

Muhammad Yusuf

DAFTAR ISI

LEMBAR PENGESAHAN.....	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxiii
DAFTAR KODE SUMBER	xxvii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Tujuan.....	2
1.3. Rumusan Permasalahan.....	2
1.4. Batasan Permasalahan	2
1.5. Metodologi	3
1.6. Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA	7
2.1. Aplikasi	7
2.2. Absensi	7
2.3. Pengenalan Wajah	8
2.4. Emgu CV.....	8
2.5. Metode <i>Eigenface</i>	12
2.6. Basis Data.....	19
2.7. MySQL.....	20
2.8. Penelitian Terkait	21
BAB III ANALISIS DAN PERANCANGAN SISTEM.....	25
3.1. Analisis.....	25
3.1.1. Analisis Permasalahan.....	25
3.1.2. Deskripsi Umum Sistem.....	26
3.1.3. Spesifikasi Kebutuhan Perangkat Lunak.....	27
3.1.4. Identifikasi Pengguna.....	82
3.2. Perancangan Sistem.....	83
3.2.1. Perancangan Arsitektur	84

3.2.2.	Proses Input Data Wajah	85
3.2.3.	Perancangan Basis Data	86
3.2.4.	Perancangan Antarmuka Pengguna.....	94
BAB IV IMPLEMENTASI.....		115
4.1.	Lingkungan Implementasi	115
4.1.1.	Lingkungan Implementasi Perangkat Keras.....	115
4.1.2.	Lingkungan Implementasi Perangkat Lunak.....	115
4.2.	Implementasi Proses Aplikasi	116
4.2.1.	Implementasi Proses Menambah Data <i>User</i> TU	116
4.2.2.	Implementasi Proses Mengubah Data <i>User</i> TU	116
4.2.3.	Implementasi Proses Menghapus Data <i>User</i> TU	116
4.2.4.	Implementasi Proses Menambah Data Mata Kuliah	116
4.2.5.	Implementasi Proses Mengubah Data Mata Kuliah	117
4.2.6.	Implementasi Proses Menghapus Data Mata Kuliah	117
4.2.7.	Implementasi Proses Menambah Data Mahasiswa	117
4.2.8.	Implementasi Proses Menambah Foto Mahasiswa	117
4.2.9.	Implementasi Proses Mengubah Data Mahasiswa	118
4.2.10.	Implementasi Proses Menghapus Data Mahasiswa	118
4.2.11.	Implementasi Proses Melihat Data Mahasiswa..	118
4.2.12.	Implementasi Proses Menambah Data Dosen	118
4.2.13.	Implementasi Proses Mengubah Data Dosen	118

4.2.14.	Implementasi Proses Menghapus Data Dosen ...	119
4.2.15.	Implementasi Proses Menambah Data Jadwal ...	119
4.2.16.	Implementasi Proses Mengubah Data Jadwal	119
4.2.17.	Implementasi Proses Menghapus Data Jadwal ..	119
4.2.18.	Implementasi Proses Menambah Data Mahasiswa Per Kelas	119
4.2.19.	Implementasi Proses Menghapus Data Mahasiswa Per Kelas	120
4.2.20.	Implementasi Proses Mengubah <i>Password</i>	120
4.2.21.	Implementasi Proses Melihat Absensi	120
4.2.22.	Implementasi Proses Mengekstrak Data Absensi	120
4.2.23.	Implementasi Proses Mengisi Berita Acara	120
4.2.24.	Implementasi Proses Melakukan Absensi.....	121
4.3.	Implementasi Metode <i>Eigenface</i>	121
4.3.1.	Implementasi <i>Camera Capture</i>	121
4.3.2.	Implementasi Deteksi Wajah	122
4.3.3.	Implementasi Pengenalan Wajah	123
4.4.	Implementasi Basis Data	124
4.4.1.	Implementasi Tabel foto.....	124
4.4.2.	Implementasi Tabel mahasiswa	124
4.4.3.	Implementasi Tabel kehadiran	125
4.4.4.	Implementasi Tabel beritaAcara	125
4.4.5.	Implementasi Tabel jadwal	126
4.4.6.	Implementasi Tabel kelas.....	127
4.4.7.	Implementasi Tabel mataKuliah	127
4.4.8.	Implementasi Tabel dosen.....	128

4.4.9.	Implementasi Tabel login.....	128
4.5.	Implementasi Antarmuka Pengguna.....	128
4.5.1.	Implementasi Antarmuka Halaman Login	129
4.5.2.	Implementasi Antarmuka Halaman Beranda untuk Administrator.....	129
4.5.3.	Implementasi Antarmuka Halaman Beranda untuk Tata Usaha.....	130
4.5.4.	Implementasi Antarmuka Ganti Password	131
4.5.5.	Implementasi Antarmuka Halaman Data Tata Usaha	132
4.5.6.	Implementasi Antarmuka Halaman Data Mahasiswa	133
4.5.7.	Implementasi Antarmuka Form Lihat Data Mahasiswa.....	134
4.5.8.	Implementasi Antarmuka Halaman Data Mata Kuliah	135
4.5.9.	Implementasi Antarmuka Halaman Data Jadwal Perkuliahan.....	136
4.5.10.	Implementasi Antarmuka Halaman Data Mahasiswa Per Kelas	137
4.5.11.	Implementasi Antarmuka Halaman Absensi	138
4.5.12.	Implementasi Antarmuka Halaman Data Absensi	139
BAB V PENGUJIAN DAN EVALUASI		141
5.1.	Lingkungan Pengujian.....	141
5.2.	Skenario Pengujian	141
5.2.1.	Data Pengujian	141
5.2.2.	Pengujian Fungsionalitas.....	159
5.3.	Evaluasi Pengujian	216

BAB VI KESIMPULAN DAN SARAN.....	219
6.1. Kesimpulan.....	219
6.2. Saran.....	220
DAFTAR PUSTAKA.....	221
LAMPIRAN A – GAMBAR TUTORIAL <i>CAMERA CAPTURE</i> ..	
.....	223
BIODATA PENULIS.....	229

DAFTAR GAMBAR

Gambar 2-1 Emgu CV <i>Platform Windows</i>	9
Gambar 2-2 Emgu CV <i>Platform Mobile Devices</i>	10
Gambar 2-3 Emgu CV <i>Platform OSX, Linux, Unix</i>	11
Gambar 2-4 Wajah 1	12
Gambar 2-5 Wajah 2	13
Gambar 2-6 Wajah Baru.....	17
Gambar 3-1 Diagram Kasus Penggunaan.....	33
Gambar 3-2 Diagram Aktivitas Kasus Penggunaan UC-0001	35
Gambar 3-3 Diagram Aktivitas Kasus Penggunaan UC-0002	37
Gambar 3-4 Diagram Aktivitas Kasus Penggunaan UC-0003	39
Gambar 3-5 Diagram Aktivitas Kasus Penggunaan UC-0004	41
Gambar 3-6 Diagram Aktivitas Kasus Penggunaan UC-0005	43
Gambar 3-7 Diagram Aktivitas Kasus Penggunaan UC-0006	45
Gambar 3-8 Diagram Aktivitas Kasus Penggunaan UC-0007	47
Gambar 3-9 Diagram Aktivitas Kasus Penggunaan UC-0008	50
Gambar 3-10 Diagram Aktivitas Kasus Penggunaan UC-0009 ..	52
Gambar 3-11 Diagram Aktivitas Kasus Penggunaan UC-0010 ..	54
Gambar 3-12 Diagram Aktivitas Kasus Penggunaan UC-0011 ..	55
Gambar 3-13 Diagram Aktivitas Kasus Penggunaan UC-0012 ..	57
Gambar 3-14 Diagram Aktivitas Kasus Penggunaan UC-0013 ..	59
Gambar 3-15 Diagram Aktivitas Kasus Penggunaan UC-0014 ..	61
Gambar 3-16 Diagram Aktivitas Kasus Penggunaan UC-0015 ..	63
Gambar 3-17 Diagram Aktivitas Kasus Penggunaan UC-0016 ..	65
Gambar 3-18 Diagram Aktivitas Kasus Penggunaan UC-0017 ..	67
Gambar 3-19 Diagram Aktivitas Kasus Penggunaan UC-0018 ..	69
Gambar 3-20 Diagram Aktivitas Kasus Penggunaan UC-0019 ..	71
Gambar 3-21 Diagram Aktivitas Kasus Penggunaan UC-0020 ..	74
Gambar 3-22 Diagram Aktivitas Kasus Penggunaan UC-0021 ..	76
Gambar 3-23 Diagram Aktivitas Penggunaan UC-0022.....	78
Gambar 3-24 Diagram Aktivitas Kasus Penggunaan UC-0023 ..	79
Gambar 3-25 Diagram Aktivitas Kasus Penggunaan UC-0024 ..	82
Gambar 3-26 Arsitektur Aplikasi	84
Gambar 3-27 Proses Input Data Wajah.....	85

Gambar 3-28 Diagram CDM.....	87
Gambar 3-29 Diagram PDM	88
Gambar 3-30 Rancangan Antarmuka Halaman Login	95
Gambar 3-31 Rancangan Antarmuka Halaman Beranda untuk Admin.....	96
Gambar 3-32 Rancangan Antarmuka Halaman Beranda untuk Tata Usaha	98
Gambar 3-33 Rancangan Antarmuka Ganti Password	100
Gambar 3-34 Rancangan Antarmuka Halaman Data Tata Usaha	101
Gambar 3-35 Rancangan Antarmuka Halaman Data Mahasiswa	103
Gambar 3-36 Rancangan Antarmuka Form Lihat Data Mahasiswa	105
Gambar 3-37 Rancangan Antarmuka Halaman Data Mata Kuliah	106
Gambar 3-38 Rancangan Antarmuka Halaman Data Jadwal Perkuliahan.....	108
Gambar 3-39 Rancangan Antarmuka Halaman Data Mahasiswa Per Kelas.....	110
Gambar 3-40 Rancangan Antarmuka Halaman Absensi	112
Gambar 3-41 Rancangan Antarmuka Halaman Data Absensi ..	113
Gambar 4-1 Implementasi Antarmuka Halaman Login	129
Gambar 4-2 Implementasi Antarmuka Halaman Beranda untuk Administrator.....	130
Gambar 4-3 Implementasi Antarmuka Halaman Beranda untuk Tata Usaha.....	131
Gambar 4-4 Implementasi Antarmuka Ganti Password	132
Gambar 4-5 Implementasi Antarmuka Halaman Data Tata Usaha	133
Gambar 4-6 Implementasi Antarmuka Halaman Data Mahasiswa	134
Gambar 4-7 Implementasi Antarmuka Form Lihat Data Mahasiswa	135

Gambar 4-8 Implementasi Antarmuka Halaman Data Mata Kuliah	136
Gambar 4-9 Implementasi Antarmuka Halaman Data Jadwal Perkuliahan.....	137
Gambar 4-10 Implementasi Antarmuka Halaman Data Mahasiswa Per Kelas	138
Gambar 4-11 Implementasi Antarmuka Halaman Absensi.....	139
Gambar 4-12 Implementasi Antarmuka Halaman Data Absensi	140
Gambar 5-1 Pengisian Data Mahasiswa.....	160
Gambar 5-2 Hasil Penambahan Data Mahasiswa	160
Gambar 5-3 <i>Form Input</i> Foto Mahasiswa	162
Gambar 5-4 <i>Input</i> Foto Metode <i>Browse</i>	163
Gambar 5-5 <i>Input</i> Foto Metode <i>Webcam</i>	164
Gambar 5-6 Hasil <i>Capture</i> Wajah.....	164
Gambar 5-7 Pengisian Berita Acara.....	166
Gambar 5-8 Keterangan Berita Acara Berhasil Disimpan dan Absensi Bisa Dimulai.....	167
Gambar 5-9 Fitur Absensi Mahasiswa	168
Gambar 5-10 Fitur Absensi Mahasiswa Diaktifkan	169
Gambar 5-11 Konfirmasi Kebenaran Identitas Mahasiswa.....	170
Gambar 5-12 Mahasiswa Telah Melakukan Absensi	171
Gambar 5-13 Daftar Data Kelas Berdasarkan ID Jadwal.....	174
Gambar 5-14 Data Absensi Mahasiswa	175
Gambar 5-15 Memilih Direktori Penyimpanan File	177
Gambar 5-16 Pesan Data Absensi Berhasil Diekstrak	177
Gambar 5-17 File Data Hasil Ekstraksi.....	178
Gambar 5-18 Pengisian Data Tata Usaha.....	180
Gambar 5-19 Hasil Penambahan Data User Tata Usaha.....	180
Gambar 5-20 Pengubahan Data <i>User</i> Tata Usaha	182
Gambar 5-21 Hasil Pengubahan Data <i>User</i> Tata Usaha	182
Gambar 5-22 Hasil Penghapusan Data <i>User</i> Tata Usaha	184
Gambar 5-23 Pengisian Data Mata Kuliah.....	186
Gambar 5-24 Hasil Penambahan Data Mata Kuliah	186
Gambar 5-25 Pengubahan Data Mata Kuliah.....	188

Gambar 5-26 Hasil Pengubahan Data Mata Kuliah	188
Gambar 5-27 Hasil Penghapusan Data Mata Kuliah	190
Gambar 5-28 Pengubahan Data Mahasiswa	191
Gambar 5-29 Hasil Pengubahan Data Mahasiswa.....	192
Gambar 5-30 Hasil Penghapusan Data Mahasiswa	193
Gambar 5-31 Hasil Lihat Data Mahasiswa.....	195
Gambar 5-32 Pengisian Data Dosen.....	196
Gambar 5-33 Hasil Penambahan Data Dosen	197
Gambar 5-34 Pengubahan Data Dosen.....	198
Gambar 5-35 Hasil Pengubahan Data Dosen	199
Gambar 5-36 Hasil Penghapusan Data Dosen.....	200
Gambar 5-37 Pengisian Data Jadwal.....	202
Gambar 5-38 Hasil Penambahan Data Jadwal	202
Gambar 5-39 Pengubahan Data Jadwal	204
Gambar 5-40 Hasil Pengubahan Data Jadwal	204
Gambar 5-41 Hasil Penghapusan Data Jadwal	206
Gambar 5-42 Daftar Data Kelas Berdasarkan ID Jadwal	208
Gambar 5-43 Daftar NRP Mahasiswa	209
Gambar 5-44 Hasil Penambahan Data Mahasiswa Per Kelas ...	210
Gambar 5-45 Hasil Penghapusan Data Mahasiswa Per Kelas...	212
Gambar 5-46 Pengisian Ganti Password	213
Gambar 5-47 Peringatan Password Lama Salah.....	214
Gambar 5-48 Peringatan <i>Password</i> Tidak Boleh Kosong.....	214
Gambar 5-49 Peringatan Konfirmasi Password Berbeda	215
Gambar 5-50 Pesan <i>Password</i> Berhasil Diubah.....	215
Gambar A-1 Buat Projek Baru pada Visual Studio	223
Gambar A-2 Pilih <i>Windows Form Application</i>	224
Gambar A-3 Tambahkan References Emgu CV	225
Gambar A-4 Tambahkan Image Box Emgu CV.....	226
Gambar A-5 Atur Border Style ImageBox Emgu CV menjadi Fixed Single.....	227
Gambar A-6 <i>Camera Capture</i> ketika Dijalankan.....	228

DAFTAR TABEL

Tabel 2-1 Penelitian Terkait	22
Tabel 3-1 Daftar Kebutuhan Fungsional	27
Tabel 3-2 Daftar Kasus Penggunaan	30
Tabel 3-3 Rincian Alur Kasus Penggunaan UC-0001.....	34
Tabel 3-4 Rincian Alur Kasus Penggunaan UC-0002.....	35
Tabel 3-5 Rincian Alur Kasus Penggunaan UC-0003.....	37
Tabel 3-6 Rincian Alur Kasus Penggunaan UC-0004.....	39
Tabel 3-7 Rincian Alur Kasus Penggunaan UC-0005.....	41
Tabel 3-8 Rincian Alur Kasus Penggunaan UC-0006.....	43
Tabel 3-9 Rincian Alur Kasus Penggunaan UC-0007.....	45
Tabel 3-10 Rincian Alur Kasus Penggunaan UC-0008.....	47
Tabel 3-11 Rincian Alur Kasus Penggunaan UC-0009.....	51
Tabel 3-12 Rincian Alur Kasus Penggunaan UC-0010.....	53
Tabel 3-13 Rincian Alur Kasus Penggunaan UC-0011.....	54
Tabel 3-14 Rincian Alur Kasus Penggunaan UC-0012.....	56
Tabel 3-15 Rincian Alur Kasus Penggunaan UC-0013.....	57
Tabel 3-16 Rincian Alur Kasus Penggunaan UC-0014.....	59
Tabel 3-17 Rincian Alur Kasus Penggunaan UC-0015.....	61
Tabel 3-18 Rincian Alur Kasus Penggunaan UC-0016.....	63
Tabel 3-19 Rincian Alur Kasus Penggunaan UC-0017.....	65
Tabel 3-20 Rincian Alur Kasus Penggunaan UC-0018.....	67
Tabel 3-21 Rincian Alur Kasus Penggunaan UC-0019.....	70
Tabel 3-22 Rincian Alur Kasus Penggunaan UC-0020.....	72
Tabel 3-23 Rincian Alur Kasus Penggunaan UC-0021.....	74
Tabel 3-24 Rincian Alur Kasus Penggunaan UC-0022.....	77
Tabel 3-25 Rincian Alur Kasus Penggunaan UC-0023.....	78
Tabel 3-26 Rincian Alur Kasus Penggunaan UC-0024.....	80
Tabel 3-27 Rancangan Tabel foto	89
Tabel 3-28 Rancangan Tabel mahasiswa	89
Tabel 3-29 Rancangan Tabel kehadiran	90
Tabel 3-30 Rancangan Tabel beritaAcara	90
Tabel 3-31 Rancangan Tabel jadwal	91
Tabel 3-32 Rancangan Tabel kelas.....	92

Tabel 3-33 Rancangan Tabel mataKuliah	93
Tabel 3-34 Rancangan Tabel dosen.....	93
Tabel 3-35 Rancangan Tabel login.....	94
Tabel 5-1 Data Uji Administrator dan Tata Usaha.....	142
Tabel 5-2 Data Uji Dosen.....	142
Tabel 5-3 Data Uji Mata Kuliah	142
Tabel 5-4 Data Uji Jadwal Perkuliahan	143
Tabel 5-5 Data Uji Foto Wajah Mahasiswa	143
Tabel 5-6 Skenario Pengujian Menambah Data Mahasiswa	159
Tabel 5-7 Skenario Pengujian Menambah Foto Mahasiswa	161
Tabel 5-8 Skenario Pengujian Mengisi Berita Acara	165
Tabel 5-9 Skenario Pengujian Melakukan Absensi.....	167
Tabel 5-10 Hasil Pengujian Melakukan Absensi	171
Tabel 5-11 Skenario Pengujian Melihat Absensi	173
Tabel 5-12 Skenario Pengujian Mengekstrak Data Absensi	176
Tabel 5-13 Skenario Pengujian Menambah Data <i>User</i> TU	179
Tabel 5-14 Skenario Pengujian Mengubah Data <i>User</i> TU.....	181
Tabel 5-15 Skenario Pengujian Menghapus Data <i>User</i> Tata Usaha	183
Tabel 5-16 Skenario Pengujian Menambah Data Mata Kuliah ..	185
Tabel 5-17 Skenario Pengujian Mengubah Data Mata Kuliah ..	187
Tabel 5-18 Skenario Pengujian Menghapus Data Mata Kuliah ..	189
Tabel 5-19 Skenario Pengujian Mengubah Data Mahasiswa	190
Tabel 5-20 Skenario Pengujian Menghapus Data Mahasiswa ..	192
Tabel 5-21 Skenario Pengujian Melihat Data Mahasiswa.....	194
Tabel 5-22 Skenario Pengujian Menambah Data Dosen.....	195
Tabel 5-23 Skenario Pengujian Mengubah Data Dosen.....	197
Tabel 5-24 Skenario Pengujian Menghapus Data Dosen	199
Tabel 5-25 Skenario Pengujian Menambah Data Jadwal	201
Tabel 5-26 Skenario Pengujian Mengubah Data Jadwal	203
Tabel 5-27 Skenario Pengujian Menghapus Data Jadwal	205
Tabel 5-28 Skenario Pengujian Menambah Data Mahasiswa Per Kelas	206
Tabel 5-29 Skenario Pengujian Menghapus Data Mahasiswa Per Kelas.....	211

Tabel 5-30 Skenario Pengujian Mengubah <i>Password</i>	212
Tabel 5-31 Rangkuman Hasil Pengujian Fungsionalitas.....	216

DAFTAR KODE SUMBER

Kode Sumber 4-1 Deklarasi HaarCascade	122
Kode Sumber 4-2 Memuat <i>File XML HaarCascade</i>	122
Kode Sumber 4-3 Deteksi Wajah	122
Kode Sumber 4-4 Kriteria Pengenalan Wajah	123
Kode Sumber 4-5 <i>Eigenface Recognizer</i>	123
Kode Sumber 4-6 Menampilkan Nama yang Dikenali	123
Kode Sumber 4-7 Implementasi Tabel foto	124
Kode Sumber 4-8 Implementasi Tabel mahasiswa	125
Kode Sumber 4-9 Implementasi Tabel kehadiran	125
Kode Sumber 4-10 Implementasi Tabel beritaAcara	126
Kode Sumber 4-11 Implementasi Tabel jadwal	126
Kode Sumber 4-12 Implementasi Tabel kelas.....	127
Kode Sumber 4-13 Implementasi Tabel mataKuliah	127
Kode Sumber 4-14 Implementasi Tabel dosen	128
Kode Sumber 4-15 Implementasi Tabel login.....	128

BAB I

PENDAHULUAN

Pada bab ini dibahas secara garis besar tugas akhir yang meliputi latar belakang, tujuan, rumusan dan batasan permasalahan, metodologi pembuatan tugas akhir, dan sistematika penulisan.

1.1. Latar Belakang

Sistem absensi adalah hal yang penting dalam sebuah perkuliahan sebagai pencatat kehadiran mahasiswa. Di kampus Institut Teknologi Sepuluh Nopember (ITS) Surabaya, kehadiran menentukan seorang mahasiswa berhak atau tidak untuk mengikuti Evaluasi Akhir Semester (EAS) dengan persentase 80% dari 16 pertemuan pada satu semester. Selama ini di ITS khususnya di jurusan Teknik Informatika, sistem absensi masih dilakukan secara manual dengan menandatangani lembar kertas absensi. Hal itu tentu kurang efektif karena terbukanya kesempatan untuk melakukan kecurangan. Selain itu, rekapitulasi oleh Tata Usaha (TU) juga dilakukan dengan cara manual yang membutuhkan waktu lama karena harus memasukkan banyak data.

Dengan semakin berkembangnya bidang teknologi, saat ini telah banyak dibangun sistem absensi dengan pengenalan pola yang mengenali ciri-ciri khusus fisik seseorang diantaranya pengenalan wajah, pengenalan iris, dan pengenalan sidik jari. Pencatatan absensi menggunakan pengenalan pola yang mengenali ciri-ciri fisik seseorang untuk perkuliahan mahasiswa sebelumnya pernah diterapkan di jurusan Sistem Informasi ITS yaitu dengan pengenalan sidik jari, akan tetapi sekarang sudah tidak berjalan dan alat *fingerprint* hanya terpasang di tiap kelas. Dengan memanfaatkan pola ciri-ciri khusus fisik pula tugas akhir ini akan mengenali seseorang, yaitu dengan pengenalan wajah.

Sistem yang dibangun memanfaatkan *web camera (webcam)* pada komputer untuk melakukan absensi. Tujuan

dikembangkannya sistem ini adalah untuk memberikan solusi yang efektif dalam pencatatan dan perekapan absensi perkuliahan.

1.2. Tujuan

Tujuan dari pembuatan tugas akhir ini adalah merancang dan membangun aplikasi absensi perkuliahan mahasiswa dengan pengenalan wajah.

1.3. Rumusan Permasalahan

Rumusan masalah yang diangkat dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana membangun aplikasi absensi perkuliahan mahasiswa dengan pengenalan wajah?
2. Bagaimana menerapkan metode *Eigenface* pada aplikasi absensi perkuliahan mahasiswa dengan pengenalan wajah?
3. Bagaimana merancang basis data yang optimal untuk aplikasi yang dibangun?

1.4. Batasan Permasalahan

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

1. Aplikasi yang dibangun menggunakan metode *Eigenface*, berbasis *desktop*, dan menggunakan bahasa pemrograman C#.
2. Wajah yang akan dideteksi dan dikenali adalah wajah yang menghadap ke depan, pada pencahayaan yang baik, dan wajah tidak terhalangi oleh kacamata, masker, ataupun objek lainnya.
3. Sistem yang dibangun berupa prototipe dan tidak terintegrasi dengan sistem informasi yang ada di ITS.
4. Sistem berjalan pada kondisi perkuliahan normal dan tidak termasuk untuk mata kuliah yang diundur atau ditunda.
5. Data jadwal perkuliahan diambil dari jadwal perkuliahan semester genap 2015/2016 Teknik Informatika ITS.

1.5. Metodologi

Langkah-langkah yang ditempuh dalam pengerjaan tugas akhir ini yaitu:

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. Subbab metodologi berisi penjelasan mengenai tahapan penyusunan tugas akhir mulai dari penyusunan proposal hingga penyusunan buku tugas akhir. Terdapat pula ringkasan tugas akhir yang berisi gambaran secara umum sistem yang dibangun dalam tugas akhir.

2. Studi Literatur

Pada tahap ini dilakukan studi literatur untuk mencari, mengumpulkan, mempelajari, dan memahami informasi yang berkaitan dengan rancang bangun aplikasi absensi dengan pengenalan wajah. Literatur yang digunakan diperoleh dari beberapa jurnal, buku, dan sumber dari internet yang terkait.

3. Analisis dan Perancangan Sistem

Pada tahap ini dilakukan analisis dan pendefinisian kebutuhan sistem untuk mengetahui hal-hal yang dibutuhkan pada sistem. Dianalisis juga aktor-aktor yang terlibat dalam sistem. Selanjutnya, melakukan perancangan sistem dari hasil analisis terhadap sistem. Langkah yang akan digunakan pada tahap ini sebagai berikut:

- a. Analisis sistem
- b. Perancangan tabel basis data
- c. Analisis kebutuhan fungsional
- d. Perancangan antarmuka.

4. Implementasi

Pada tahap ini dilakukan realisasi terhadap aplikasi sebagai bentuk implementasi dari analisis dan perancangan sistem yang telah dilakukan sebelumnya.

Perincian tahap ini adalah sebagai berikut:

- a. Implementasi proses aplikasi
- b. Implementasi metode yang digunakan pada aplikasi
- c. Implementasi basis data yang telah dirancang
- d. Implementasi pembuatan antarmuka aplikasi.

5. Pengujian dan Evaluasi

Pada tahap ini dilakukan pengujian terhadap elemen perangkat lunak dengan menggunakan skenario yang telah disiapkan sebelumnya. Uji coba dan evaluasi dilakukan untuk mencari masalah yang mungkin timbul, mengevaluasi jalannya program, dan mengadakan perbaikan jika ada kekurangan. Pengujian dilakukan menggunakan pengujian kotak hitam (*blackbox*). Pengujian kotak hitam adalah pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, penguji dapat mendefinisikan kumpulan kondisi masukan dan melakukan pengujian pada spesifikasi fungsional program. Pengujian ini dilakukan untuk memeriksa masing-masing fungsi yang ada pada sistem.

6. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan pendokumentasian dan pelaporan dari seluruh konsep, tinjauan pustaka, rancangan, implementasi, proses yang telah dilakukan, dan hasil-hasil yang telah didapatkan selama pengerjaan tugas akhir. Buku

tugas akhir ini bertujuan untuk memberikan gambaran dari pengerjaan tugas akhir dan dapat berguna bagi pembaca yang tertarik untuk pengembangan aplikasi ini lebih lanjut.

1.6. Sistematika Penulisan

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 ini berisi latar belakang masalah, tujuan dan manfaat pembuatan tugas akhir, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan tugas akhir.

Bab II Tinjauan Pustaka

Bab ini membahas beberapa teori penunjang yang berhubungan dengan pokok pembahasan dan mendasari pembuatan tugas akhir ini.

Bab III Analisis dan Perancangan Sistem

Bab ini membahas mengenai perancangan perangkat lunak meliputi perancangan data, antarmuka, arsitektur dan proses pada perangkat lunak yang dibangun.

Bab IV Implementasi

Bab ini membahas tentang implementasi dari analisis dan perancangan sistem dalam bentuk kode dan antarmuka perangkat lunak.

Bab V Pengujian dan Evaluasi

Bab ini membahas pengujian dengan metode pengujian berdasarkan skenario yang telah ditentukan. Melakukan

evaluasi fitur aplikasi apakah telah memenuhi kebutuhan fungsional.

Bab VI Kesimpulan

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan dan membahas saran-saran untuk pengembangan perangkat lunak kedepannya.

Daftar Pustaka

Merupakan daftar referensi yang digunakan untuk mengembangkan tugas akhir.

Lampiran

Merupakan bab tambahan yang berisi daftar kode sumber dan daftar gambar yang berkaitan dengan aplikasi ini.

BAB II

TINJAUAN PUSTAKA

Bab ini berisi mengenai penjelasan teori yang berkaitan dengan implementasi perangkat lunak. Penjelasan tersebut bertujuan untuk memberikan gambaran mengenai sistem yang dibangun dan berguna sebagai pendukung dalam pengembangan perangkat lunak.

2.1. Aplikasi

Aplikasi atau yang sering disebut perangkat lunak adalah suatu program yang dirancang dan dibangun dengan memiliki tujuan tertentu. Aplikasi memiliki beberapa basis penerapan, misalnya *Desktop*, *Web*, *Handphone*, dan lain-lain.

Kode-kode dari suatu aplikasi bisa bersifat *open-source* dan *close-source*. Aplikasi *open-source* adalah aplikasi yang kode sumbernya tersedia dengan lisensi dimana pemilik hak cipta memberikan hak untuk mengembangkan, pembelajaran, dan mendistribusikannya kepada siapa saja dan untuk tujuan apa saja [1]. Sedangkan *close-source* atau *proprietary* adalah aplikasi yang tidak gratis atau tidak *open-source*.

2.2. Absensi

Absensi adalah sebuah konsep orang sebagai individu maupun kelompok pada sebuah lokasi untuk acara yang telah dijadwalkan sebelumnya. Mengukur tingkat kehadiran dapat menghasilkan informasi yang dapat digunakan untuk mengukur efektivitas usaha seseorang.

Dalam pengaturan pembelajaran dan tempat kerja, kehadiran adalah sesuatu yang wajib. Minimnya kehadiran seseorang dapat mempengaruhi nilai ataupun evaluasi lainnya. Kehadiran yang buruk juga bisa mencerminkan masalah dalam situasi pribadi mahasiswa, dan merupakan indikator bahwa mahasiswa tidak mengembangkan pengetahuan dan keterampilan yang diperlukan untuk meraih kesuksesan [2].

2.3. Pengenalan Wajah

Pengenalan wajah merupakan salah satu pendekatan pengenalan pola untuk keperluan identifikasi wajah seseorang dengan pendekatan biometrik. Suatu biometrik bersifat unik sehingga dapat digunakan untuk mengenali identitas seseorang. Proses pengenalan biometrik dapat dibagi menjadi dua karakteristik, yaitu secara fisik dan secara perilaku. Biometrik fisik berasal dari pengukuran dan data yang ada langsung dari bagian manusia misalnya pengenalan sidik jari, pengenalan wajah, iris, retina, dan tangan. Sedangkan biometrik perilaku berasal dari pengukuran dan data yang berasal dari tindakan seperti suara, tanda tangan, dan *keystrokes*. Sistem biometrik mengacu pada terintegrasinya antara perangkat keras dan perangkat lunak untuk melakukan proses identifikasi dan verifikasi.

Sistem pengenalan wajah adalah sebuah aplikasi komputer yang mampu mengidentifikasi atau mengenali wajah seseorang dari gambar digital atau video. Salah satu cara yang digunakan dalam pengenalan wajah yaitu dengan membandingkan wajah dari gambar yang dipilih atau video dengan basis data wajah.

Pengenalan wajah manusia telah menarik banyak para peneliti. Sebuah sistem pengenalan wajah bisa ditemukan pada banyak aplikasi dari berbagai bidang seperti Interaksi Manusia dan Komputer (IMK), sistem keamanan, dan lain-lain [3].

2.4. Emgu CV

Emgu CV adalah sebuah jembatan lintas *platform* .Net untuk menghubungkan ke *library* pengolahan gambar Open CV. Emgu CV memungkinkan untuk memanggil fungsi Open CV ke berbagai bahasa .Net seperti C#, VB, VC++, IronPython dan lain-lain, yang dapat di-*compile* pada Visual Studio, Xamarin Studio, dan Unity, serta dapat dijalankan pada perangkat Windows, Linux, Mac OS X, iOS, Android, dan Windows Phone [4].

Emgu CV tersedia untuk beberapa *platform*, dan memiliki fitur yang berbeda. Berikut perbandingan berbagai versi Emgu CV untuk berbagai *platform*

- *Platform Windows*
Fitur-fitur *library* Emgu CV pada *platform Windows* dapat dilihat pada Gambar 2-1.

Name	Emgu CV (Open Source)	Emgu CV for Windows (Commercial Optimized)		Emgu CV for Unity
Development tools	Visual Studio 2010 and up	Visual Studio 2010 and up	Visual Studio 2013 and up	Unity 3D Pro v5.1 and up
Platform	Windows	Windows	Windows 8.1 Store app *	Windows Desktop Standalone
Supported CPU Architecture	i386, x64	i386, x64	i386, x64	i386 (Editor & Standalone), x64 (Standalone)
CUDA GPU Processing	✓	✓	X	X
OpenCL (GPU&CPU)	✓	✓	X	✓
Tesseract OCR	✓	✓	✓	✓
Compiled with Intel C++ Compiler, TBB & IPP	X	✓	X	X
Exception Handling	✓	✓	✓	✓
Debugger Visualizer	✓	✓	X	X
Emgu.CV.UI	✓	✓	X	X
License	GPL	Commercial License		Commercial License

Gambar 2-1 Emgu CV Platform Windows

- *Platform Mobile Devices*
Fitur-fitur *library* Emgu CV pada *platform mobile devices* dapat dilihat pada Gambar 2-2.

Name	Emgu CV for Xamarin iOS Bussiness Version+	Emgu CV for Xamarin Android Bussiness Version+	Emgu CV for Windows Phone	Emgu CV for Unity		
Requirement	Xamarin iOS Bussiness Version+	Xamarin Android Bussiness Version+	Visual Studio 2013 and Windows Phone 8.1	Unity 3D v5.1 Pro / Free version		
Platform	iOS (iPhone, iPad, iPod Touch)	Android	Windows Phone 8.1	iOS (iPhone, iPad, iPod Touch)	Android	Windows Phone 8.1
Supported Device	armv7, armv7a, arm64	armeabi, armeabi-v7a, arm64-v8a, x86, x86_64	arm	armv7, armv7a, arm64	armeabi-v7a, x86	arm
Supported Simulator	i386, x64	armeabi, armeabi-v7a, arm64-v8a, x86, x86_64	x86	i386	armeabi-v7a, x86	x86
CUDA GPU Processing	X	X	X	X	X	X
OpenCL 1.2 (GPU&CPU)	X	✓ (on supported devices ↗)	X	X	✓ (on supported devices ↗)	X
Tesseract OCR ↗	✓	✓	✓	✓	✓	✓
Exception Handling	✓	✓	✓	✓	✓	✓
License	Commercial License	Commercial License	Commercial License	Commercial License		

Gambar 2-2 Emgu CV Platform Mobile Devices

- *Platform* OSX, Linux, Unix
Fitur-fitur *library* Emgu CV pada *platform* OSX, Linux, dan Unix dapat dilihat pada Gambar 2-3.

Name	Emgu CV (Open Source)	Emgu CV for OSX, Linux Unix(Commercial License)	Emgu CV for Unity
Development Tools	Xamarin Studio or MonoDevelop	Xamarin Studio or MonoDevelop	Unity 3D Pro v5.1
OSX	✓	✓	x86 & x64 (Editor & Standalone)
Linux, Unix	✓	✓	X
CUDA GPU Processing	✓	✓	X
OpenCL (GPU&CPU)	✓	✓	✓
Tesseract OCR	✓	✓	✓
Exception Handling	✓	✓	✓
Emgu.CV.UI	✓	✓	X
License	GPL	Commercial License	Commercial License

Gambar 2-3 Emgu CV Platform OSX, Linux, Unix

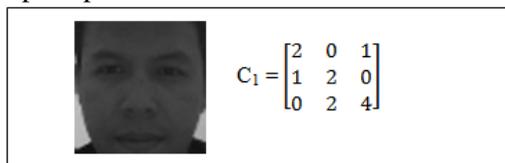
2.5. Metode *Eigenface*

Eigenface adalah nama yang diberikan untuk satu set *eigenvector* ketika digunakan dalam pengenalan wajah pada bidang visi komputer. Dalam istilah Layman, *Eigenface* adalah sekumpulan *standardized face ingredient* dari analisis statistik dari banyak gambar wajah [5]. Pendekatan *Eigenfaces* untuk pengenalan wajah dikembangkan oleh Sirovich dan Kirby (1987) dan digunakan oleh Matthew Turk dan Alex Pentland pada klasifikasi wajah [6]. Selain merancang sistem untuk pengenalan wajah, Matthew Turk dan Alex Pentland juga menunjukkan cara menghitung *eigenvector* untuk melakukan *eigendecomposition* pada sebagian besar gambar wajah. *Eigenvector* berasal dari kovarian matriks distribusi probabilitas pada ruang vektor dari gambar wajah [7]. ¹*Data set* wajah yang digunakan harus diambil dalam kondisi pencahayaan dan resolusi yang sama dengan saat melakukan pengenalan wajah baru.

Algoritma pengenalan wajah dengan metode *Eigenface* dilakukan melalui beberapa tahapan [8], yaitu:

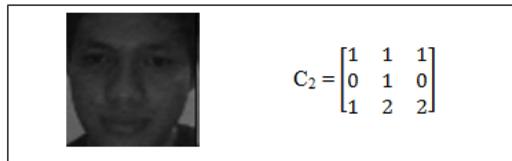
1. Menyusun *Flatvector* Matriks Citra

Langkah pertama yaitu menyusun suatu himpunan S matriks yang terdiri dari seluruh *training image* ($\Gamma_1, \Gamma_2, \dots, \Gamma_m$). Misal ada dua data wajah yang masing-masing memiliki nilai matriks seperti pada Gambar 2-4 dan 2-5.



Gambar 2-4 Wajah 1

¹ Wikipedia <https://en.wikipedia.org/wiki/Eigenface> (diakses pada 04/05/2016)



Gambar 2-5 Wajah 2

2. Menghitung Nilai Tengah atau *Mean* (Ψ)
Kemudian jumlahkan nilai matriks pada wajah 1 dan wajah 2, dan bagikan dengan jumlah data wajah yang ada pada basis data.

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n$$

$$\Psi = \frac{1}{2} \sum_{n=1}^2 \Gamma_n = \frac{1}{2} \left[\begin{array}{c} \begin{bmatrix} 2 & 0 & 1 \\ 1 & 2 & 0 \\ 0 & 2 & 4 \end{bmatrix} + \\ \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 2 & 2 \end{bmatrix} \end{array} \right]$$

$$\Psi = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 2 & 3 \end{bmatrix}$$

3. Menghitung Selisih antara *Training Image* dengan Nilai Tengah atau *Mean* (Ψ)

Kemudian kurangkan *training image* (Γ) dengan *mean* (Ψ)

$$\phi_i = \Gamma_i - \Psi$$

$$\begin{aligned} \phi_1 = \Gamma_1 - \Psi &= \begin{bmatrix} 2 & 0 & 1 \\ 1 & 2 & 0 \\ 0 & 2 & 4 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 2 & 3 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

$$\begin{aligned}\phi_2 = \Gamma_2 - \Psi &= \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 2 & 2 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 2 & 3 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}\end{aligned}$$

4. Menghitung Nilai Matriks Kovarian (C)

Nilai matriks kovarian (C) digunakan untuk menghitung *eigenvalue* (λ) dan *eigenvector* (v)

$$C = \frac{1}{M} \sum_{n=1}^M \phi_n \phi_n^T = AA^T$$

$$L = A^T A \quad L = \phi_m^T \phi_n$$

$$L = \begin{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} x \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{bmatrix}$$

$$L = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \rightarrow \text{matriks kovarian}$$

5. Menghitung Nilai *Eigenvalue* dan *Eigenvector*

Kemudian menghitung *eigenvalue* (λ) dan *eigenvector* (v) dari matriks kovarian (C)

$$C x v_i = \lambda_i x v_i$$

$$L x v = \lambda x v$$

$$(L - \lambda I) = 0 \text{ atau } (\lambda I - L) = 0$$

$$0 = \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

Maka *eigenvalue* yang dihasilkan adalah $\lambda_1 = 3, \lambda_2 = 1, \lambda_3 = 3$

$$v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

Sedangkan *eigenvector* dihasilkan dengan mensubstitusikan nilai *eigenvalue* (λ) ke dalam persamaan $(\lambda I - L)v = 0$

Eigenvector dari masing-masing *eigenvalue* didapat berdasarkan masing-masing kolom *eigenvalue* dan kemudian dihimpun kembali menjadi satu matriks.

a. Untuk $\lambda_1 = 3$

$$\begin{bmatrix} \lambda - 2 & -1 & 0 \\ -1 & \lambda - 2 & 0 \\ 0 & 0 & \lambda - 3 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Dihasilkan *eigenvector* v_1 adalah $\begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}$

b. Untuk $\lambda_2 = 1$

$$\begin{bmatrix} \lambda - 2 & -1 & 0 \\ -1 & \lambda - 2 & 0 \\ 0 & 0 & \lambda - 3 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Dihasilkan *eigenvector* v_2 adalah $\begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix}$

c. Untuk $\lambda_3 = 3$

$$\begin{bmatrix} \lambda - 2 & -1 & 0 \\ -1 & \lambda - 2 & 0 \\ 0 & 0 & \lambda - 3 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Dihasilkan *eigenvector* V_3 adalah $\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$

Setelah didapat *eigenvector* v_1, v_2, v_3 , maka *eigenvector* yang dihasilkan dari matriks L adalah

$$\begin{bmatrix} 1 & -1 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

6. Mencari Nilai *Eigenface*

Nilai *Eigenface* (μ) dapat dicari dengan

$$\mu_i = \sum_{k=1}^M v_{ik} \phi_k$$

$$i = 1, \dots, M$$

Cari nilai *Eigenface* (μ)

$$\mu_i = \sum_{k=1}^M v_{ik} \phi_k$$

$$\mu_1 = v \times \phi_1$$

$$= \begin{bmatrix} 1 & -1 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mu_1 = \begin{bmatrix} 0 & -1 & 0 \\ -2 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\mu_2 = v \times \phi_2$$

$$= \begin{bmatrix} 1 & -1 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

$$\mu_2 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

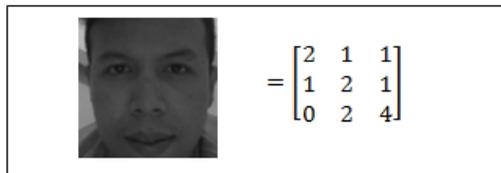
7. Proses Identifikasi

Proses identifikasi adalah proses dimana wajah baru masuk dan akan dicocokkan dengan wajah yang ada di dalam basis data. Untuk mengenali wajah baru dilakukan langkah yang sama dengan sebelumnya untuk mendapatkan nilai *Eigenface* dari wajah baru

$$\mu_{new} = v \times (\Gamma_{new} - \Psi)$$

$$\Omega = [\mu_1, \mu_2, \dots, \mu_M]$$

Nilai matriks wajah baru dapat dilihat pada gambar dibawah ini



Gambar 2-6 Wajah Baru

Pertama cari selisih (ϕ) antara wajah baru dengan *mean* (Ψ).

$$\phi_{new} = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 0 & 2 & 4 \\ 1 & 1 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 2 & 3 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Kemudian cari nilai *Eigenface*

$$\mu_{new} = v \times \phi_{new}$$

$$\mu_{new} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mu_{new} = \begin{bmatrix} -2 & -2 & -1 \\ 0 & 0 & 0 \end{bmatrix}$$

Gunakan *euclidean distance* untuk mencari selisih antara nilai *Eigenface* wajah di basis data (Γ_i) dengan wajah baru (Γ_{new}). Kemudian jumlahkan matriks dari masing-masing *euclidean distance*.

$$\varepsilon_k = \|\Omega - \Omega_k\|$$

$$\varepsilon_k = \|\Omega - \Omega_{new}\|$$

$$\begin{aligned} \varepsilon_1 &= \|\Omega - \Omega_{new}\| \\ &= \left\| \begin{bmatrix} 0 & 0 & -1 \\ -2 & -2 & -1 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & -1 & 0 \\ -2 & -2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\| \\ &= \sqrt{\frac{(0)^2 + (1)^2 + (1)^2 + (0)^2 + (1)^2 + (1)^2 + (0)^2 + (0)^2 + (0)^2 + (0)^2}{(1)^2 + (0)^2 + (0)^2 + (0)^2}} \\ &= \sqrt{4} = 2 \end{aligned}$$

$$\begin{aligned} \varepsilon_2 &= \|\Omega - \Omega_{new}\| \\ &= \left\| \begin{bmatrix} 0 & 0 & -1 \\ -2 & -2 & -1 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\| \\ &= \sqrt{\frac{(0)^2 + (1)^2 + (1)^2 + (4)^2 + (1)^2 + (1)^2 + (0)^2 + (0)^2 + (0)^2 + (0)^2}{(1)^2 + (0)^2 + (0)^2 + (0)^2}} \\ &= \sqrt{8} = 2,828 \end{aligned}$$

Dari hasil perhitungan didapatkan jarak *Eigenface* citra wajah 1 lebih kecil, maka dapat disimpulkan bahwa wajah baru lebih mirip dengan wajah 1 dibandingkan dengan wajah 2.

2.6. Basis Data

Chou mendefinisikan basis data sebagai kumpulan informasi bermanfaat yang diorganisasikan ke dalam tata cara khusus. Menurut Fabbri dan Schwab, basis data adalah sistem berkas terpadu yang dirancang terutama untuk meminimalkan pengulangan data. Menurut Date, basis data dapat dianggap sebagai tempat untuk sekumpulan berkas data terkomputerisasi yang tujuan utamanya adalah memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan [9]. Jadi basis data dapat didefinisikan dalam berbagai sudut pandang sebagai berikut [10]:

1. Himpunan kelompok data yang saling berhubungan yang diorganisasikan sedemikian rupa sehingga kelak dapat dimanfaatkan dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa tanpa pengulangan (*redundancy*) yang tidak perlu, untuk memenuhi kebutuhan.
3. Kumpulan *file*/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Sedangkan sistem manajemen basis data (DBMS) merupakan paket program (*software*) yang dibuat agar memudahkan dan mengefisienkan pemasukan, pengubahan, penghapusan, dan pengambilan informasi terhadap basis data. Penyimpanan data dalam DBMS akan mempunyai banyak manfaat dan kelebihan, seperti:

1. *Performance*
Jika data yang dikelola cukup besar, penggunaan DBMS akan menyebabkan efisiensi dalam hal media penyimpanan dan penggunaan memori.
2. Integritas

Integritas data akan lebih terjamin, seperti masalah *redundancy*. *Redundancy* adalah kejadian berulang data atau kumpulan data yang sama dalam sebuah basis data yang mengakibatkan pemborosan media penyimpanan.

3. Independensi
Perubahan struktur basis data memungkinkan terjadi tanpa harus mengubah aplikasi untuk mengaksesnya. Sehingga pembuatan antarmuka kedalam data akan lebih mudah.
4. Sentralisasi
Data yang terpusat akan mempermudah pengelolaan basis data. Kemudahan melakukan bagi pengguna dan konsistensi data yang diakses secara bersama-sama akan dapat lebih terjamin.
5. Security
DBMS memiliki sistem keamanan yang lebih fleksibel daripada pengamanan pada *file* sistem operasi. Keamanan dalam DBMS memberikan keluwesan untuk memberikan hak akses kepada pengguna daripada keamanan dalam sistem operasi.

Software yang tergolong dalam sistem manajemen basis data antara lain, Microsoft SQL, MySQL, Oracle, Microsoft Access, dan lain-lain [11].

2.7. MySQL

MySQL adalah sistem manajemen basis data *open-source* yang dibangun, didistribusikan, dan didukung oleh Oracle Corporation.

Dari pengertian tersebut maka dapat dijabarkan

1. MySQL adalah sebuah sistem manajemen basis data.
Basis data adalah kumpulan dari data yang terstruktur. Untuk menambahkan, mengakses, dan memproses data pada basis data komputer memerlukan sistem manajemen basis data. Sistem manajemen basis data memiliki peran yang penting dalam komputasi, bisa sebagai *standalone utilities* atau bagian dari aplikasi lainnya.

2. Basis Data MySQL adalah relasional.
Basis data relasional menyimpan data dalam tabel yang terpisah. Ada beberapa pengaturan dalam basis data relasional antara data yang berbeda bisa *one-to-one*, *one-to-many*, dan sebagainya. Dengan pengaturan seperti itu, maka basis data tidak akan *inconsistent*, *duplicate*, *orphan*, *out-of-date*, atau kehilangan data.
3. MySQL adalah *open-source*
Setiap orang bisa mengunduh MySQL dari internet dan dapat menggunakannya tanpa harus membayar.
4. *Server* basis MySQL bekerja pada *client/server* atau sistem *embedded*
Basis data MySQL adalah sistem *client/server* yang terdiri dari *multi-threaded* SQL server yang mendukung *backends* yang berbeda, beberapa program dan *library client* yang berbeda, dan berbagai *application programming interfaces*(API) [12].

2.8. Penelitian Terkait

Beberapa penelitian terkait dengan tugas akhir ini diantaranya penelitian yang dilakukan oleh Priyanka Wagh, Roshani Thakare, Jagruti Chaudhari, dan Shweta Patil dalam jurnal yang berjudul *Attendance System based on Face Recogniiton using Eigenface and PCA Algorithm*. Hasil penelitian menunjukkan bahwa sistem kehadiran yang otomatis dapat dibuktikan sebagai sistem yang efisien untuk pencatatan kehadiran mahasiswa. Dengan menggunakan sistem ini, kemungkinan kecurangan dapat dikurangi [13].

Muhammad Rizki Muliawan, Beni Irawan, dan Yulrio Brianorman dalam jurnal yang berjudul Implementasi Pengenalan Wajah dengan Metode Eigenface pada Sistem Absensi. Penelitian ini mendapatkan hasil yang berbeda-beda antara wajah satu dengan wajah yang lainnya, pada saat basis data berisi 10 data wajah, hasil rata-rata persentase kecocokan mencapai 88%, sedangkan pada saat basis data berjumlah 20 data wajah, hasil rata-rata persentase

kecocokan mencapai 52%. Penyebab dari perbedaan hasil tersebut adalah faktor pencahayaan, jarak, bentuk wajah, serta jumlah data yang tersedia [8].

Adapun penelitian-penelitian terdahulu dari tugas akhir ini dapat dilihat pada Tabel 2-1 berikut.

Tabel 2-1 Penelitian Terkait

No	Peneliti	Judul	Hasil
1.	Priyanka Wagh, Roshani Thakare, Jagruti Chaudhari, dan Shweta Patil	<i>Attendance System based on Face Recogniiton using using Eigenface and PCA Algorithm</i>	Sistem kehadiran yang otomatis dapat dibuktikan sebagai sistem yang efisien untuk pencatatan kehadiran mahasiswa
2.	Muhammad Rizki Muliawan, Beni Irawan, dan Yulrio Brianorman	Implementasi Pengenalan Wajah dengan Metode Eigenface	mendapatkan hasil yang didapat berbeda-beda antara wajah satu dengan wajah yang lainnya, pada saat basis data berisi 10 data wajah, hasil rata-rata persentase kecocokan mencapai 88%, sedangkan pada saat basis data berjumlah 20 data wajah, hasil rata-rata persentase kecocokan mencapai 52%.

No	Peneliti	Judul	Hasil
			Penyebab dari perbedaan hasil tersebut adalah faktor pencahayaan, jarak, bentuk wajah, serta jumlah data yang tersedia
3.	Indra	Sistem Pengenalan Wajah dengan Metode Eigenface untuk Absensi pada PT Florindo Lestari	Pengenalan wajah adalah salah satu sistem yang dapat dikembangkan untuk menjadi aplikasi absensi yang dapat diterapkan diperusahaan untuk mencegah manipulasi absen oleh karyawan
4.	Siswo Wardoyo, Romi Wiryadinata, Raya Sagita	Sistem Absensi Berbasis Algoritma Eigenface dengan Metode Principal Component Analysis	Sistem presensi menggunakan ekspresi wajah, aksesoris wajah, dan variasi jarak, didapat sensitivitas 100%, spesifisitas 55,55%, dan akurasi 69,33%.
5.	Riyan Syah Putra	Perancangan Aplikasi Absensi dengan	Pengenalan wajah ini salah satunya dapat

No	Peneliti	Judul	Hasil
		Deteksi Wajah Menggunakan Metode Eigenface	dikembangkan untuk menjadi aplikasi absensi yang dapat diterapkan diperusahaan untuk mencegah manipulasi absen oleh karyawan.

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini membahas analisis permasalahan, analisis kebutuhan, dan perancangan dari sistem yang dibangun. Analisis permasalahan membahas permasalahan yang diangkat dalam pengerjaan tugas akhir. Kemudian analisis kebutuhan mencantumkan kebutuhan-kebutuhan yang diperlukan sistem. Selanjutnya dibahas mengenai perancangan sistem yang dibuat untuk memberikan gambaran tentang sistem yang dibangun.

3.1. Analisis

Pada subbab ini membahas analisis pembuatan aplikasi absensi perkuliahan mahasiswa dengan pengenalan wajah. Analisis yang dilakukan meliputi analisis permasalahan, deskripsi umum sistem, identifikasi pengguna, dan kebutuhan sistem.

3.1.1. Analisis Permasalahan

Sistem absensi adalah hal yang penting dalam sebuah perkuliahan sebagai pencatat kehadiran mahasiswa. Di kampus Institut Teknologi Sepuluh Nopember (ITS) Surabaya, kehadiran menentukan seorang mahasiswa berhak atau tidak untuk mengikuti Evaluasi Akhir Semester (EAS) dengan persentase 80% dari 16 pertemuan pada satu semester. Selama ini di ITS khususnya di jurusan Teknik Informatika, sistem absensi masih dilakukan secara manual dengan menandatangani lembar kertas absensi. Hal itu tentu kurang efektif karena terbukanya kesempatan untuk melakukan kecurangan. Selain itu, rekapitulasi oleh Tata Usaha (TU) juga dilakukan dengan cara manual yang membutuhkan waktu lama karena harus memasukkan banyak data.

Dengan semakin berkembangnya bidang teknologi, saat ini telah banyak dibangun sistem absensi dengan pengenalan pola yang mengenali ciri-ciri khusus fisik seseorang diantaranya pengenalan wajah, pengenalan iris, dan pengenalan sidik jari. Pencatatan absensi menggunakan pengenalan pola yang mengenali

ciri-ciri fisik seseorang untuk perkuliahan mahasiswa sebelumnya pernah diterapkan di jurusan Sistem Informasi ITS yaitu dengan pengenalan sidik jari, akan tetapi sekarang sudah tidak berjalan dan alat *fingerprint* hanya terpasang di tiap kelas. Dengan memanfaatkan pola ciri-ciri khusus fisik pula tugas akhir ini akan mengenali seseorang, yaitu dengan pengenalan wajah.

Sistem yang dibangun memanfaatkan *web camera (webcam)* pada komputer untuk melakukan absensi. Tujuan dikembangkannya sistem ini adalah untuk memberikan solusi yang efektif dalam pencatatan dan perekapan absensi perkuliahan.

3.1.2. Deskripsi Umum Sistem

Aplikasi yang dibuat dalam tugas akhir ini merupakan aplikasi absensi perkuliahan mahasiswa dengan pengenalan wajah. Aplikasi perkuliahan mahasiswa ini merupakan aplikasi berbasis *desktop*. Aplikasi ini berupa prototipe dan tidak terintegrasi dengan sistem informasi yang ada di ITS. Aplikasi ini dibangun menggunakan metode *Eigenface* yang merupakan algoritma dari *library* Open CV yang dipanggil dalam bahasa pemrograman *C#* dengan menggunakan *library* Emgu CV. Metode ini nantinya digunakan untuk mengenali wajah mahasiswa saat melakukan absensi mahasiswa pada suatu perkuliahan. Semua data yang ada pada aplikasi disimpan dalam relasional manajemen basis data MySQL. Data-data yang dibutuhkan pada sistem dikelola oleh pengguna administrator dan tata usaha. Semua menu yang bisa diakses oleh pengguna tata usaha juga dapat diakses oleh pengguna administrator. Perbedaan antara pengguna administrator dan tata usaha terletak pada menu Tata Usaha, dimana administrator bisa mengelola data-data dari tata usaha.

Proses absensi diawali dengan dosen *login* ke sistem absensi, kemudian mengisi detail materi pada berita acara perkuliahan suatu mata kuliah dan menekan tombol Mulai Absensi. Proses absensi dilakukan dengan menggunakan *webcam* pada komputer. Aplikasi akan mendeteksi dan mengenali wajah mahasiswa jika *data set* wajah yang ada di basis data memiliki pencahayaan dan resolusi

yang sama pada saat melakukan absensi. Ketika wajah dikenali oleh aplikasi, maka aplikasi akan mengkonfirmasi apakah benar wajah yang dikenali oleh sistem adalah wajah mahasiswa yang bersangkutan. Jika mahasiswa mengkonfirmasi benar, maka sistem akan mencatat kehadiran mahasiswa dan menyimpannya ke basis data.

3.1.3. Spesifikasi Kebutuhan Perangkat Lunak

Berdasarkan uraian permasalahan yang telah dijelaskan sebelumnya, maka terdapat beberapa spesifikasi perangkat lunak yang harus dipenuhi dalam pembangunan aplikasi ini agar sistem mampu menyelesaikan permasalahan dalam tugas akhir ini. Bagian ini berisi semua kebutuhan perangkat lunak yang diuraikan seperti kebutuhan dalam sistem yang mencakup kebutuhan fungsional.

3.1.3.1. Kebutuhan Fungsional

Kebutuhan fungsional berisi semua kebutuhan pokok yang harus dipenuhi di dalam aplikasi agar sistem dapat berjalan dengan baik. Kebutuhan fungsional secara tidak langsung mendeskripsikan fungsionalitas dasar perangkat lunak. Daftar kebutuhan fungsional dapat dilihat pada Tabel 3-1.

Tabel 3-1 Daftar Kebutuhan Fungsional

No.	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
1.	F-0001	Menambah data <i>user</i> TU	Pengguna dapat menambah data <i>user</i> tata usaha, berupa <i>username</i> , nama, dan <i>password</i> yang digunakan tata usaha untuk masuk ke dalam sistem.
2.	F-0002	Mengubah data <i>user</i> TU	Pengguna dapat mengubah data nama <i>user</i> tata usaha, untuk <i>username</i> dan <i>password</i> tidak bisa diubah

No.	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
			karena menjadi hak pemilik akun.
3.	F-0003	Menghapus data <i>user</i> TU	Pengguna dapat menghapus data <i>user</i> tata usaha.
4.	F-0004	Menambah data mata kuliah	Pengguna dapat menambah data mata kuliah.
5.	F-0005	Mengubah data mata kuliah	Pengguna dapat mengubah data mata kuliah.
6.	F-0006	Menghapus data mata kuliah	Pengguna dapat menghapus data mata kuliah.
7.	F-0007	Menambah data mahasiswa	Pengguna dapat menambah data mahasiswa.
8.	F-0008	Menambah foto mahasiswa	Pengguna dapat menambah foto mahasiswa.
9.	F-0009	Mengubah data mahasiswa	Pengguna dapat mengubah data mahasiswa.
10.	F-0010	Menghapus data mahasiswa	Pengguna dapat menghapus data mahasiswa.
11.	F-0011	Melihat data mahasiswa	Pengguna dapat melihat data mahasiswa.
12.	F-0012	Menambah data dosen	Pengguna dapat menambah data dosen.
13.	F-0013	Mengubah data dosen	Pengguna dapat mengubah data dosen.

No.	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
14.	F-0014	Menghapus data dosen	Pengguna dapat menghapus data dosen.
15.	F-0015	Menambah data jadwal	Pengguna dapat menambah data jadwal perkuliahan.
16.	F-0016	Mengubah data jadwal	Pengguna dapat mengubah data jadwal perkuliahan.
17.	F-0017	Menghapus data jadwal	Pengguna dapat menghapus data jadwal perkuliahan.
18.	F-0018	Menambah data mahasiswa per kelas	Pengguna dapat menambah data mahasiswa per kelas.
19.	F-019	Menghapus data mahasiswa per kelas	Pengguna dapat menghapus data mahasiswa per kelas.
20.	F-020	Mengubah <i>password</i>	Pengguna dapat mengubah <i>password</i> .
21.	F-021	Melihat absensi	Pengguna dapat melihat rekapitulasi absensi
22.	F-022	Mengekstrak data absensi	Pengguna dapat mengekstrak <i>file</i> rekapitulasi absensi.
23.	F-023	Mengisi berita acara	Pengguna dapat mengisi berita acara berupa materi yang disampaikan per pertemuan.
24.	F-024	Melakukan absensi	Pengguna dapat mencatat kehadirannya pada perkuliahan.

3.1.3.2. Kasus Penggunaan

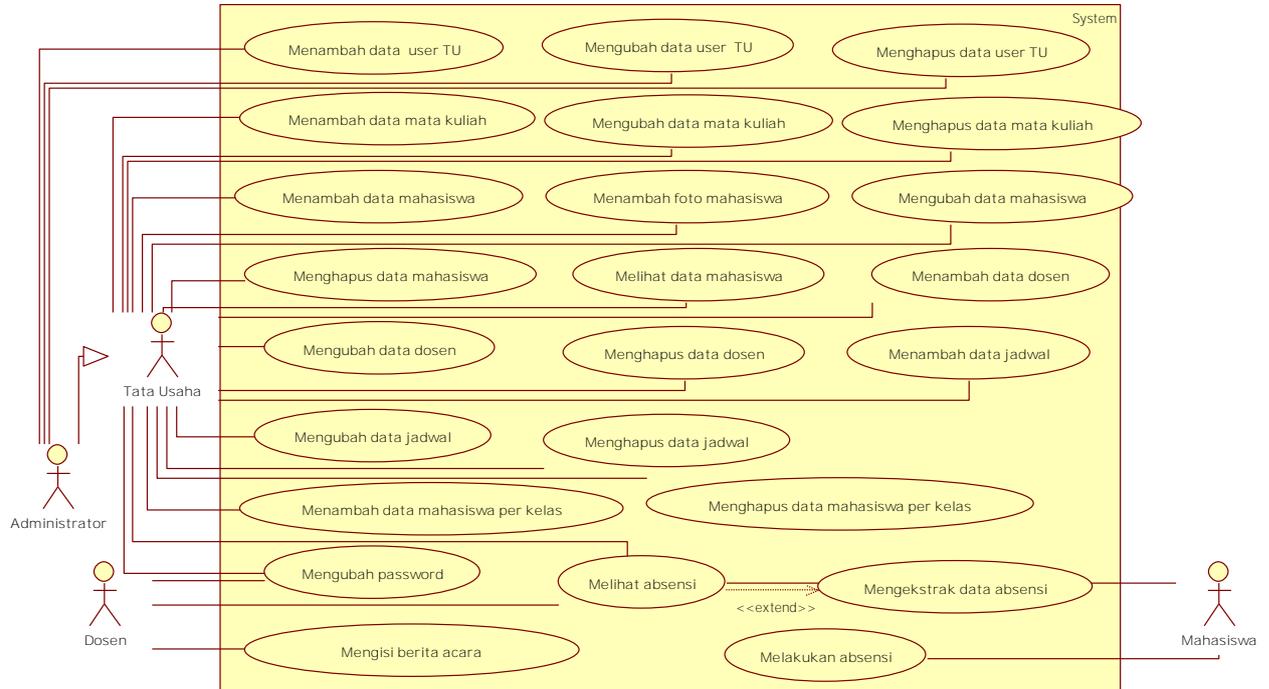
Berdasarkan analisis spesifikasi kebutuhan fungsional dan identifikasi pengguna dari sistem, maka dibuat kasus penggunaan sistem. Kasus penggunaan sistem digambarkan dalam bentuk diagram kasus penggunaan dan dalam tabel penjelasan kasus penggunaan. Terdapat dua puluh tiga kasus penggunaan, seperti yang akan dijelaskan pada Tabel 3-2 dan Gambar 3-1.

Tabel 3-2 Daftar Kasus Penggunaan

No.	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
1.	UC-0001	Menambah data <i>user</i> TU	Aktor dapat menambah data <i>user</i> tata usaha, berupa <i>username</i> , nama, dan <i>password</i> yang digunakan tata usaha untuk masuk ke dalam sistem.
2.	UC-0002	Mengubah data <i>user</i> TU	Aktor dapat mengubah data nama <i>user</i> tata usaha, untuk <i>username</i> dan <i>password</i> tidak bisa diubah karena menjadi hak pemilik akun.
3.	UC-0003	Menghapus data <i>user</i> TU	Aktor dapat menghapus data <i>user</i> tata usaha.
4.	UC-0004	Menambah data mata kuliah	Aktor dapat menambah data mata kuliah.
5.	UC-0005	Mengubah data mata kuliah	Aktor dapat mengubah data mata kuliah.
6.	UC-0006	Menghapus data mata kuliah	Aktor dapat menghapus data mata kuliah.

No.	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
7.	UC-0007	Menambah data mahasiswa	Aktor dapat menambah data mahasiswa.
8.	UC-0008	Menambah foto mahasiswa	Aktor dapat menambah foto mahasiswa.
9.	UC-0009	Mengubah data mahasiswa	Aktor dapat mengubah data mahasiswa.
10.	UC-0010	Menghapus data mahasiswa	Aktor dapat menghapus data mahasiswa.
11.	UC-0011	Melihat data mahasiswa	Aktor dapat melihat data mahasiswa.
12.	UC-0012	Menambah data dosen	Aktor dapat menambah data dosen.
13.	UC-0013	Mengubah data dosen	Aktor dapat mengubah data dosen.
14.	UC-0014	Menghapus data dosen	Aktor dapat menghapus data dosen.
15.	UC-0015	Menambah data jadwal	Aktor dapat menambah data jadwal perkuliahan.
16.	UC-0016	Mengubah data jadwal	Aktor dapat mengubah data jadwal perkuliahan.
17.	UC-0017	Menghapus data jadwal	Aktor dapat menghapus data jadwal perkuliahan.
18.	UC-0018	Menambah data mahasiswa per kelas	Aktor dapat menambah data mahasiswa per kelas.
19.	UC-019	Menghapus data	Aktor dapat menghapus data mahasiswa per kelas.

No.	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
		mahasiswa per kelas	
20.	UC-020	Mengubah <i>password</i>	Aktor dapat mengubah <i>password</i> .
21.	UC-021	Melihat absensi	Aktor dapat melihat rekapitulasi absensi.
22.	UC-022	Mengekstrak data absensi	Aktor dapat mengekstrak <i>file</i> rekapitulasi absensi.
23.	UC-023	Mengisi berita acara	Aktor dapat mengisi berita acara berupa materi yang disampaikan per pertemuan.
24.	UC-024	Melakukan absensi	Aktor dapat mencatat kehadirannya pada perkuliahan.



Gambar 3-1 Diagram Kasus Penggunaan

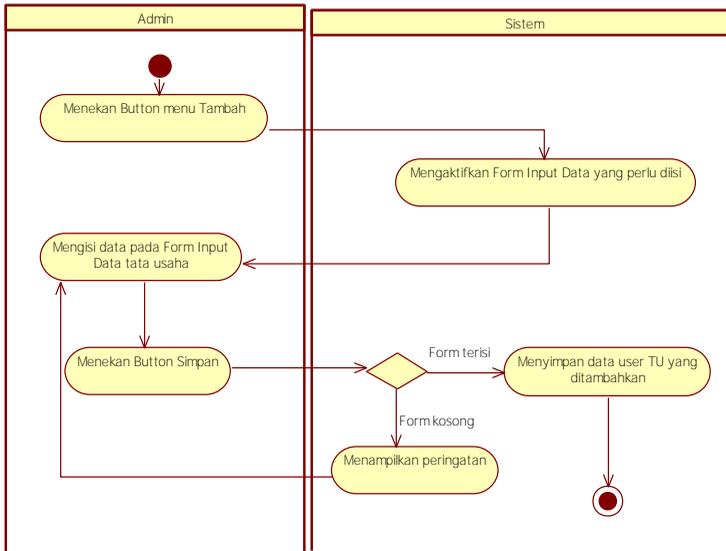
3.1.3.2.1. Kasus Penggunaan Menambah Data *User* TU

Pada kasus penggunaan UC-0001, pengguna dapat menambahkan data *user* tata usaha untuk mengakses halaman Beranda tata usaha. Rincian alur kasus penggunaan UC-0001 dijelaskan pada Tabel 3-3 dan diagram aktivitas kasus penggunaan UC-0001 dijelaskan pada Gambar 3-2

Tabel 3-3 Rincian Alur Kasus Penggunaan UC-0001

Nama	Menambah <i>User</i> TU
Kode	UC-0001
Aktor	Administrator
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Tata Usaha
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna menekan tombol menu Tambah 2. Sistem mengaktifkan <i>Form</i> Input Data yang perlu diisi 3. Pengguna mengisi data pada <i>form</i> Input Data tata usaha 4. Pengguna menekan tombol Simpan <ol style="list-style-type: none"> A1. <i>Form</i> data kosong 5. Sistem menyimpan data <i>user</i> tata usaha yang ditambahkan 6. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. <i>Form</i> data kosong</p> <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 3
Kondisi Akhir	Data <i>user</i> tata usaha disimpan ke dalam basis data

Kebutuhan Khusus	-
------------------	---



Gambar 3-2 Diagram Aktivitas Kasus Penggunaan UC-0001

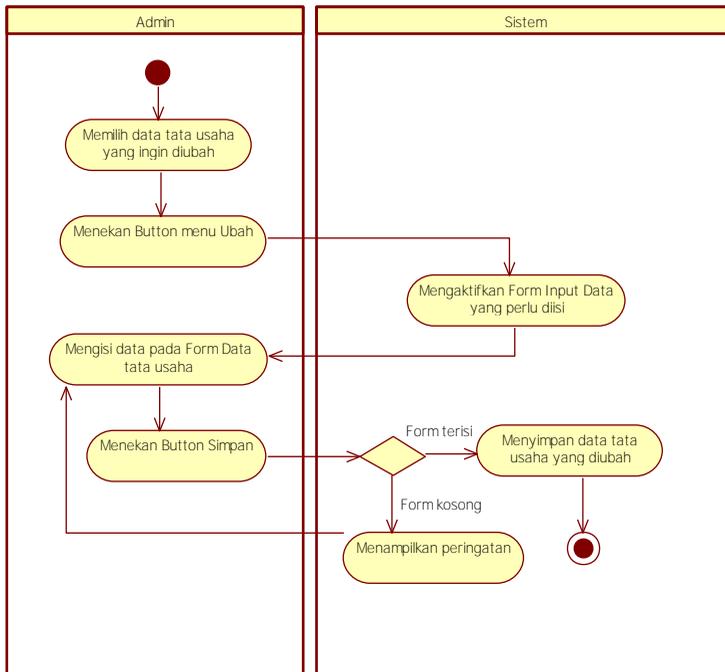
3.1.3.2.2. Kasus Penggunaan Mengubah Data *User* TU

Pada kasus penggunaan UC-0002, pengguna dapat mengubah data nama *user* tata usaha. Rincian alur kasus penggunaan UC-0002 dijelaskan pada Tabel 3-4 dan diagram aktivitas kasus penggunaan UC-0002 dijelaskan pada Gambar 3-3.

Tabel 3-4 Rincian Alur Kasus Penggunaan UC-0002

Nama	Mengubah Data <i>User</i> TU
Kode	UC-0002
Aktor	Administrator

Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Tata Usaha
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih data tata usaha yang ingin diubah pada daftar data tata usaha 2. Pengguna menekan tombol menu Ubah 3. Sistem mengaktifkan <i>form</i> Input Data yang perlu diisi 4. Pengguna mengisi data pada <i>form</i> Input Data tata usaha 5. Pengguna menekan tombol Simpan A1. <i>Form</i> data kosong 6. Sistem menyimpan data tata usaha yang diubah 7. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. <i>Form</i> data kosong</p> <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 4
Kondisi Akhir	Data tata usaha diubah dari basis data
Kebutuhan Khusus	-



Gambar 3-3 Diagram Aktivitas Kasus Penggunaan UC-0002

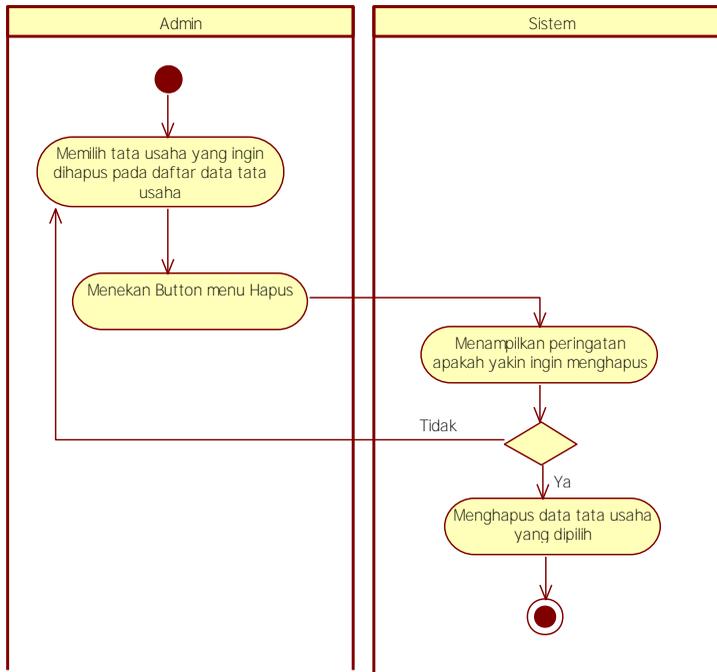
3.1.3.2.3. Kasus Penggunaan Menghapus Data *User TU*

Pada kasus penggunaan UC-0003, pengguna dapat menghapus data tata usaha. Rincian alur kasus penggunaan UC-0003 dijelaskan pada Tabel 3-5 dan diagram aktivitas kasus penggunaan UC-0003 dijelaskan pada Gambar 3-4.

Tabel 3-5 Rincian Alur Kasus Penggunaan UC-0003

Nama	Menghapus Data <i>User TU</i>
Kode	UC-0003
Aktor	Administrator

Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Tata Usaha
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih tata usaha yang ingin dihapus pada daftar data tata usaha 2. Pengguna menekan tombol menu Hapus 3. Sistem menampilkan peringatan apakah yakin ingin menghapus <ol style="list-style-type: none"> A1. Tidak A2. Ya 4. Sistem menghapus data tata usaha yang dipilih 5. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. Tidak</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 1 <p>A2. Ya</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 4
Kondisi Akhir	Data tata usaha dihapus dari basis data
Kebutuhan Khusus	-



Gambar 3-4 Diagram Aktivitas Kasus Penggunaan UC-0003

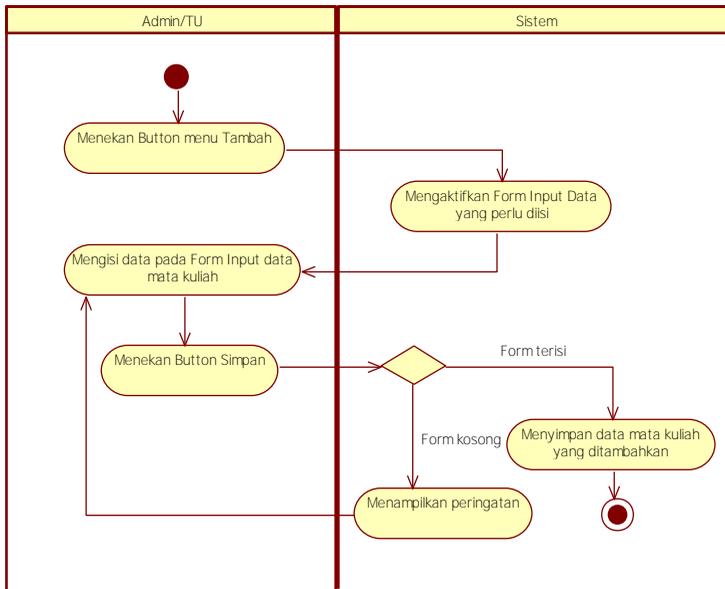
3.1.3.2.4. Kasus Penggunaan Menambah Data Mata Kuliah

Pada kasus penggunaan UC-0004, pengguna dapat menambah data mata kuliah. Rincian alur kasus penggunaan UC-0004 dijelaskan pada Tabel 3-6 dan diagram aktivitas kasus penggunaan UC-0004 dijelaskan pada Gambar 3-5.

Tabel 3-6 Rincian Alur Kasus Penggunaan UC-0004

Nama	Menambah Data Mata Kuliah
Kode	UC-0004
Aktor	Administrator dan Tata Usaha

Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Mata Kuliah
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna menekan tombol menu Tambah 2. Sistem mengaktifkan <i>form</i> Input Data yang perlu diisi 3. Pengguna mengisi data pada <i>form</i> Input Data mata kuliah 4. Pengguna menekan tombol Simpan <ol style="list-style-type: none"> A1. <i>Form</i> data kosong 5. Sistem menyimpan data mata kuliah yang ditambahkan 6. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. <i>Form</i> data kosong</p> <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 3
Kondisi Akhir	Data mata kuliah disimpan ke dalam basis data
Kebutuhan Khusus	-



Gambar 3-5 Diagram Aktivitas Kasus Penggunaan UC-0004

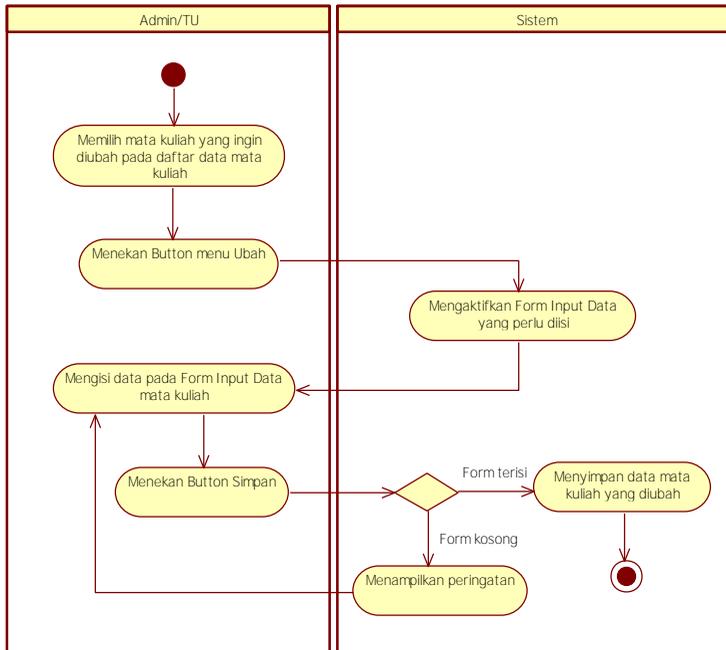
3.1.3.2.5. Kasus Penggunaan Mengubah Data Mata Kuliah

Pada kasus penggunaan UC-0005, pengguna dapat mengubah data mata kuliah. Rincian alur kasus penggunaan UC-0005 dijelaskan pada Tabel 3-7 dan diagram aktivitas kasus penggunaan UC-0005 dijelaskan pada Gambar 3-6.

Tabel 3-7 Rincian Alur Kasus Penggunaan UC-0005

Nama	Mengubah Data Mata Kuliah
Kode	UC-0005
Aktor	Administrator dan Tata Usaha

Kondisi Awal	Pengguna sudah masuk ke dalam sistem dana berada di halaman Data Mata Kuliah
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih mata kuliah yang ingin diubah pada daftar data mata kuliah 2. Pengguna menekan tombol menu Ubah 3. Sistem mengaktifkan <i>form</i> Input Data yang perlu diisi 4. Pengguna mengisi data pada <i>form</i> data mata kuliah 5. Pengguna menekan tombol Simpan A1. <i>Form</i> data kosong 6. Sistem menyimpan data mata kuliah yang diubah 7. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. <i>Form</i> data kosong</p> <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 4
Kondisi Akhir	Data mata kuliah diubah dari basis data
Kebutuhan Khusus	-



Gambar 3-6 Diagram Aktivitas Kasus Penggunaan UC-0005

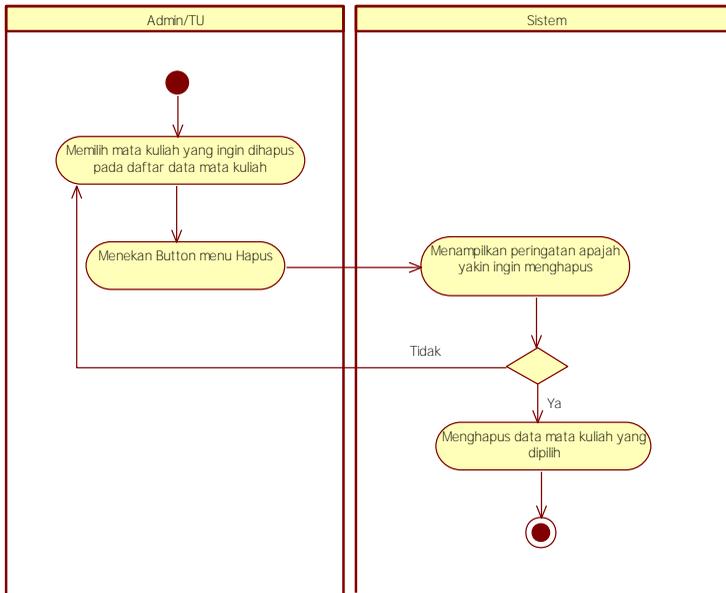
3.1.3.2.6. Kasus Penggunaan Menghapus Data Mata Kuliah

Pada kasus penggunaan UC-0006, pengguna dapat menghapus data mata kuliah. Rincian alur kasus penggunaan UC-0006 dijelaskan pada Tabel 3-8 dan diagram aktivitas kasus penggunaan UC-0006 dijelaskan pada Gambar 3-7.

Tabel 3-8 Rincian Alur Kasus Penggunaan UC-0006

Nama	Menghapus Data Mata Kuliah
Kode	UC-0006
Aktor	Administrator dan Tata Usaha

Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Mata Kuliah
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih mata kuliah yang ingin dihapus pada daftar data mata kuliah 2. Pengguna menekan tombol menu Hapus 3. Sistem menampilkan peringatan apakah yakin ingin menghapus <ol style="list-style-type: none"> A1. Tidak A2. Ya 4. Sistem menghapus data mata kuliah yang dipilih 5. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. Tidak</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 1 <p>A2. Ya</p> <ol style="list-style-type: none"> 2. Berlanjut ke aliran kejadian normal nomor 4
Kondisi Akhir	Data mata kuliah dihapus dari basis data mata kuliah
Kebutuhan Khusus	-



Gambar 3-7 Diagram Aktivitas Kasus Penggunaan UC-0006

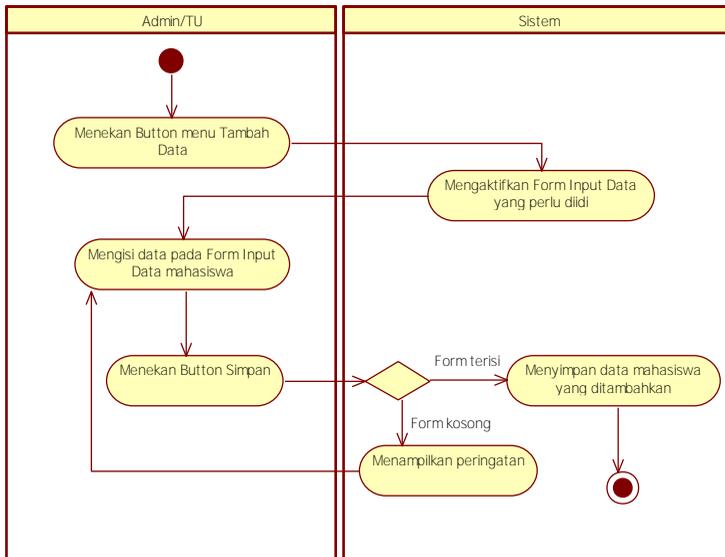
3.1.3.2.7. Kasus Penggunaan Menambah Data Mahasiswa

Pada kasus penggunaan UC-0007, pengguna dapat menambah data mahasiswa. Rincian alur kasus penggunaan UC-0007 dijelaskan pada Tabel 3-9 dan diagram aktivitas kasus penggunaan UC-0007 dijelaskan pada Gambar 3-8.

Tabel 3-9 Rincian Alur Kasus Penggunaan UC-0007

Nama	Menambah Data Mahasiswa
Kode	UC-0007
Aktor	Administrator dan Tata Usaha

Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Mahasiswa
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna menekan tombol menu Tambah Data 2. Sistem mengaktifkan <i>form</i> Input Data yang perlu diisi 3. Pengguna mengisi data pada <i>form</i> Input Data mahasiswa 4. Pengguna menekan tombol Simpan A1. <i>Form</i> data kosong 5. Sistem menyimpan data mahasiswa yang ditambahkan 6. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	A1. <i>Form</i> data kosong <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 3
Kondisi Akhir	Data mahasiswa disimpan ke dalam basis data
Kebutuhan Khusus	-



Gambar 3-8 Diagram Aktivitas Kasus Penggunaan UC-0007

3.1.3.2.8. Kasus Penggunaan Menambah Foto Mahasiswa

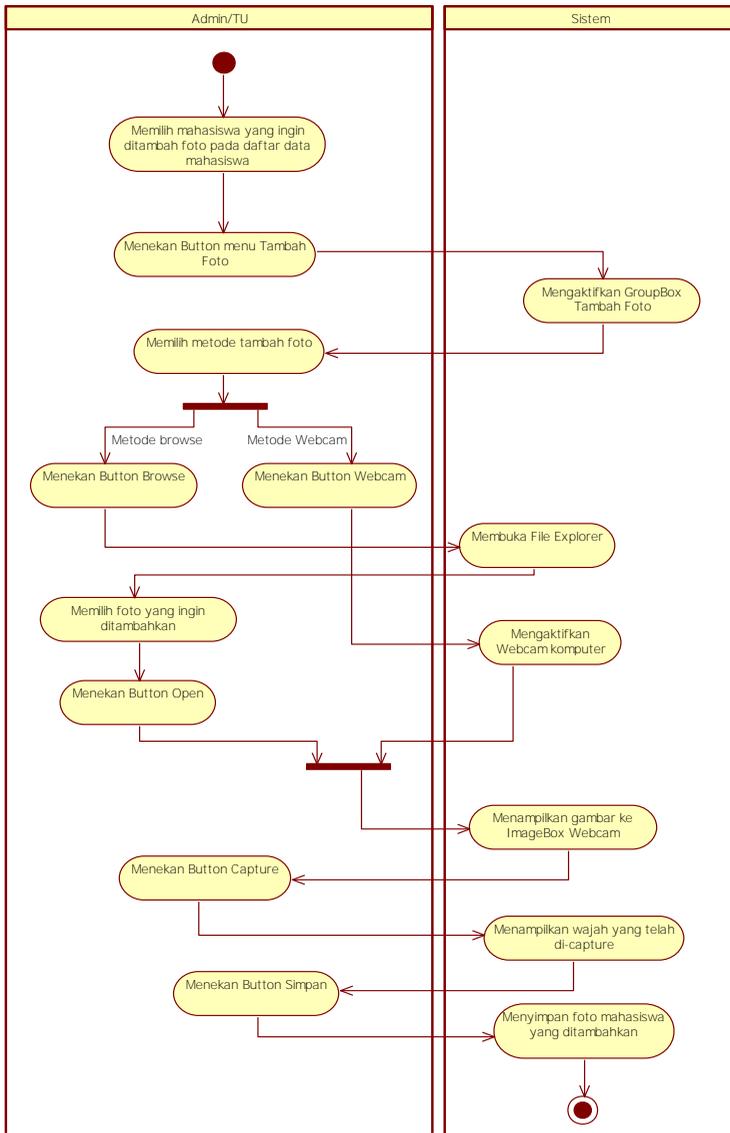
Pada kasus penggunaan UC-0008, pengguna dapat menambah foto mahasiswa. Rincian alur kasus penggunaan UC-0008 dijelaskan pada Tabel 3-10 dan diagram aktivitas kasus penggunaan UC-0008 dijelaskan pada Gambar 3-9.

Tabel 3-10 Rincian Alur Kasus Penggunaan UC-0008

Nama	Menambah Foto Mahasiswa
Kode	UC-0008
Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Mahasiswa

<p>Aliran Kejadian Normal</p>	<ol style="list-style-type: none"> 1. Pengguna memilih mahasiswa yang ingin ditambah foto pada daftar data mahasiswa 2. Pengguna menekan tombol menu Tambah Foto 3. Sistem mengaktifkan <i>GroupBox</i> Tambah Foto 4. Pengguna memilih metode tambah foto <ul style="list-style-type: none"> A1. Metode <i>Browse</i> A2. Metode <i>Webcam</i> 5. Sistem menampilkan gambar ke <i>ImageBox</i> Webcam 6. Pengguna menekan tombol Capture 7. Sistem menampilkan gambar wajah yang telah di-<i>capture</i> 8. Pengguna menekan tombol Simpan 9. Sistem menyimpan foto mahasiswa yang ditambahkan. 10. Kasus penggunaan berakhir
<p>Aliran Kejadian Alternatif</p>	<p>A1. Metode <i>Browse</i></p> <ol style="list-style-type: none"> 1. Pengguna menekan tombol Browse 2. Sistem membuka <i>browse File Explorer</i> 3. Pengguna memilih Foto yang ingin ditambahkan 4. Pengguna menekan tombol Open 5. Kembali ke aliran kejadian normal nomor 5 <p>A2. Metode <i>Webcam</i></p>

	<ol style="list-style-type: none">1. Pengguna menekan tombol Webcam2. Sistem mengaktifkan <i>webcam</i> komputer3. Berlanjut ke aliran kejadian normal nomor 5
Kondisi Akhir	Foto mahasiswa disimpan ke dalam basis data
Kebutuhan Khusus	-



Gambar 3-9 Diagram Aktivitas Kasus Penggunaan UC-0008

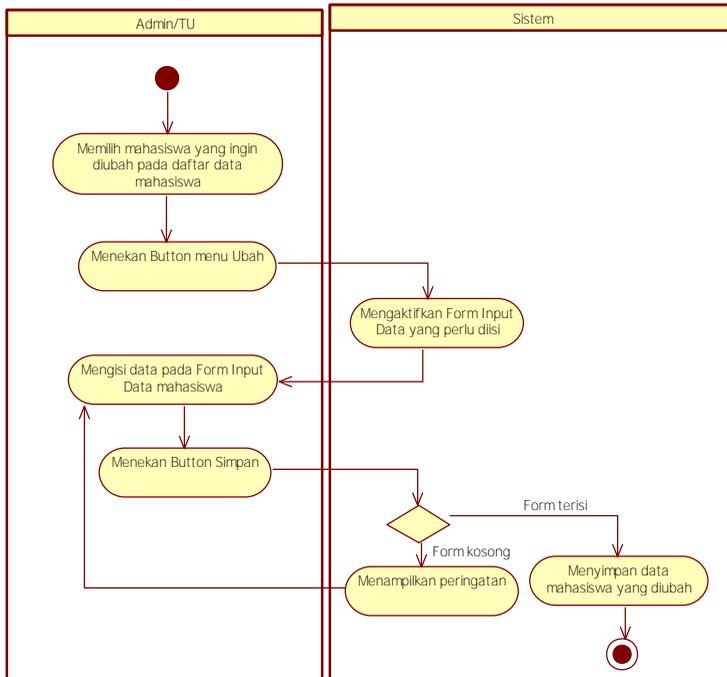
3.1.3.2.9. Kasus Penggunaan Mengubah Data Mahasiswa

Pada kasus penggunaan UC-0009, pengguna dapat mengubah data mahasiswa. Rincian alur kasus penggunaan UC-0009 dijelaskan pada Tabel 3-11 dan diagram aktivitas kasus penggunaan UC-0009 dijelaskan pada Gambar 3-10.

Tabel 3-11 Rincian Alur Kasus Penggunaan UC-0009

Nama	Mengubah Data Mahasiswa
Kode	UC-0009
Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Mahasiswa
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih mahasiswa yang ingin diubah pada daftar data mahasiswa 2. Pengguna menekan tombol menu Ubah 3. Sistem mengaktifkan <i>form</i> Input Data yang perlu diisi 4. Pengguna mengisi data pada <i>form</i> Input Data mahasiswa 5. Pengguna menekan tombol Simpan <ol style="list-style-type: none"> A1. <i>Form</i> data kosong 6. Sistem menyimpan data mahasiswa yang diubah 7. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<ol style="list-style-type: none"> A1. <i>Form</i> data kosong <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 4

Kondisi Akhir	Data mahasiswa diubah dari basis data
Kebutuhan Khusus	-



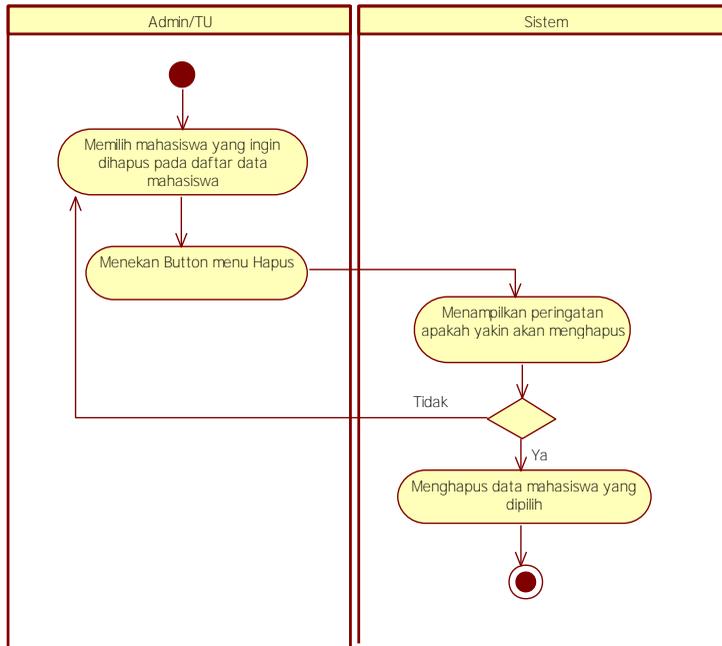
Gambar 3-10 Diagram Aktivitas Kasus Penggunaan UC-0009

3.1.3.2.10. Kasus Penggunaan Menghapus Data Mahasiswa

Pada kasus penggunaan UC-0009, pengguna dapat menghapus data mahasiswa. Rincian alur kasus penggunaan UC-0009 dijelaskan pada Tabel 3-12 dan diagram aktivitas kasus penggunaan UC-0009 dijelaskan pada Gambar 3-11.

Tabel 3-12 Rincian Alur Kasus Penggunaan UC-0010

Nama	Manghapus Data Mahasiswa
Kode	UC-0009
Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Mahasiswa
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih mahasiswa yang ingin dihapus pada daftar data mahasiswa 2. Pengguna menekan tombol menu Hapus 3. Sistem menampilkan peringatan apakah yakin akan menghapus data mahasiswa <ul style="list-style-type: none"> A1. Tidak A2. Ya 4. Sistem menghapus data mahasiswa yang dipilih 5. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. Tidak</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 1 <p>A2. Ya</p> <ol style="list-style-type: none"> 2. Berlanjut ke aliran kejadian normal nomor 4
Kondisi Akhir	Data mahasiswa dihapus dari basis data
Kebutuhan Khusus	-



Gambar 3-11 Diagram Aktivitas Kasus Penggunaan UC-0010

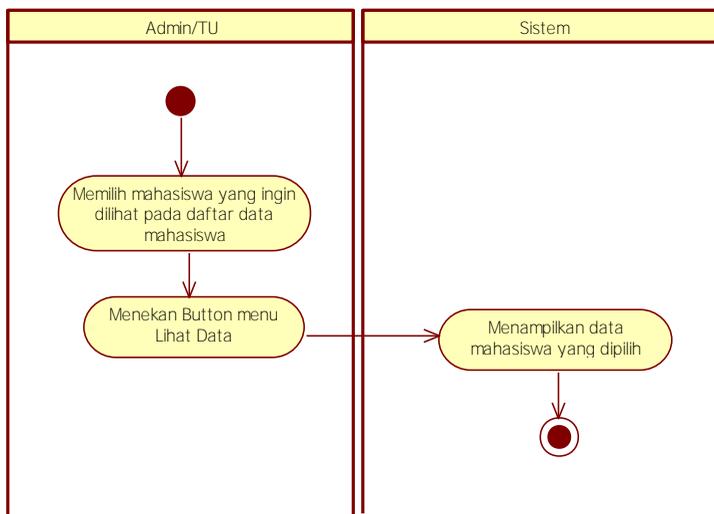
3.1.3.2.11. Kasus Penggunaan Melihat Data Mahasiswa

Pada kasus penggunaan UC-0011, pengguna dapat melihat data mahasiswa. Rincian alur kasus penggunaan UC-0011 dijelaskan pada Tabel 3-13 dan diagram aktivitas kasus penggunaan UC-0011 dijelaskan pada Gambar 3-12.

Tabel 3-13 Rincian Alur Kasus Penggunaan UC-0011

Nama	Melihat Data Mahasiswa
Kode	UC-0011
Aktor	Administrator dan Tata Usaha

Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Mahasiswa
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih mahasiswa yang ingin dilihat pada daftar data mahasiswa 2. Pengguna menekan tombol menu Lihat Data 3. Sistem menampilkan data mahasiswa yang dipilih 4. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	-
Kondisi Akhir	Daftar data mata kuliah ditampilkan
Kebutuhan Khusus	-



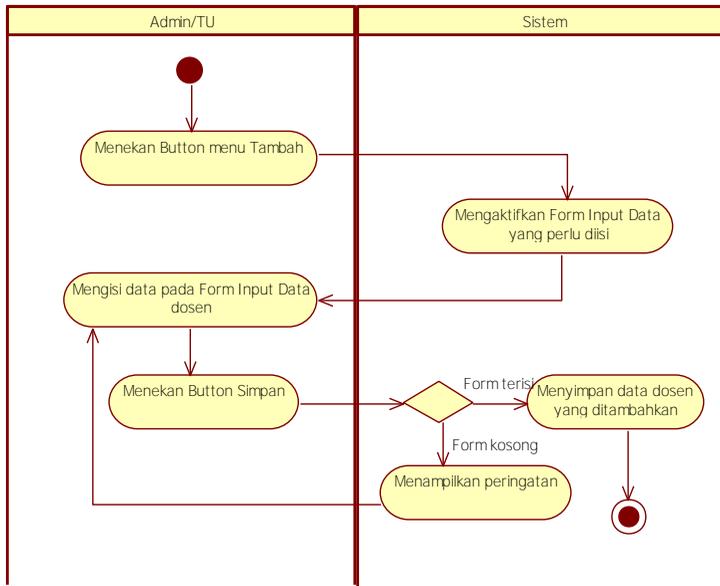
Gambar 3-12 Diagram Aktivitas Kasus Penggunaan UC-0011

3.1.3.2.12. Kasus Penggunaan Menambah Data Dosen

Pada kasus penggunaan UC-0012, pengguna dapat menambahkan data dosen. Rincian alur kasus penggunaan UC-0012 dijelaskan pada Tabel 3-14 dan diagram aktivitas kasus penggunaan UC-0012 dijelaskan pada Gambar 3-13.

Tabel 3-14 Rincian Alur Kasus Penggunaan UC-0012

Nama	Menambah Data Dosen
Kode	UC-0012
Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Dosen
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna menekan tombol menu Tambah 2. Sistem mengaktifkan <i>form</i> Input Data yang perlu diisi 3. Pengguna mengisi data pada <i>form</i> Input Data dosen 4. Pengguna menekan tombol Simpan A1. <i>Form</i> data kosong 5. Sistem menyimpan data dosen yang ditambahkan 6. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	A1. <i>Form</i> data kosong <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 3
Kondisi Akhir	Data dosen disimpan ke dalam basis data
Kebutuhan Khusus	-



Gambar 3-13 Diagram Aktivitas Kasus Penggunaan UC-0012

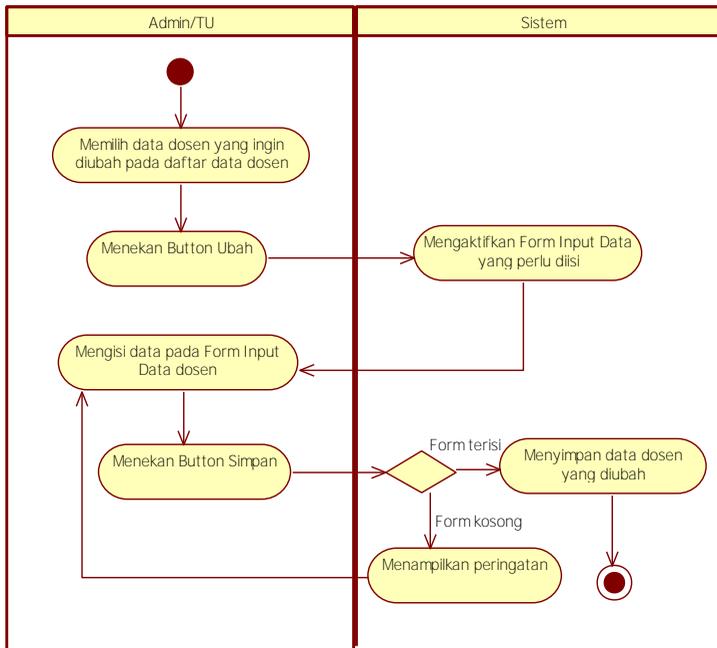
3.1.3.2.13. Kasus Penggunaan Mengubah Data Dosen

Pada kasus penggunaan UC-0013, pengguna dapat mengubah data dosen. Rincian alur kasus penggunaan UC-0013 dijelaskan pada Tabel 3-15 dan diagram aktivitas kasus penggunaan UC-0013 dijelaskan pada Gambar 3-14.

Tabel 3-15 Rincian Alur Kasus Penggunaan UC-0013

Nama	Mengubah Data Dosen
Kode	UC-0013
Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Dosen

Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih data dosen yang ingin diubah pada daftar data dosen 2. Pengguna menekan tombol menu Ubah 3. Sistem mengaktifkan <i>form</i> Input Data yang perlu diisi 4. Pengguna mengisi data pada <i>form</i> Input Data dosen 5. Pengguna menekan tombol Simpan A1. <i>Form</i> data kosong 6. Sistem menyimpan data dosen yang diubah 7. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	A1. <i>Form</i> data kosong <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 4
Kondisi Akhir	Data dosen diubah dari basis data
Kebutuhan Khusus	-



Gambar 3-14 Diagram Aktivitas Kasus Penggunaan UC-0013

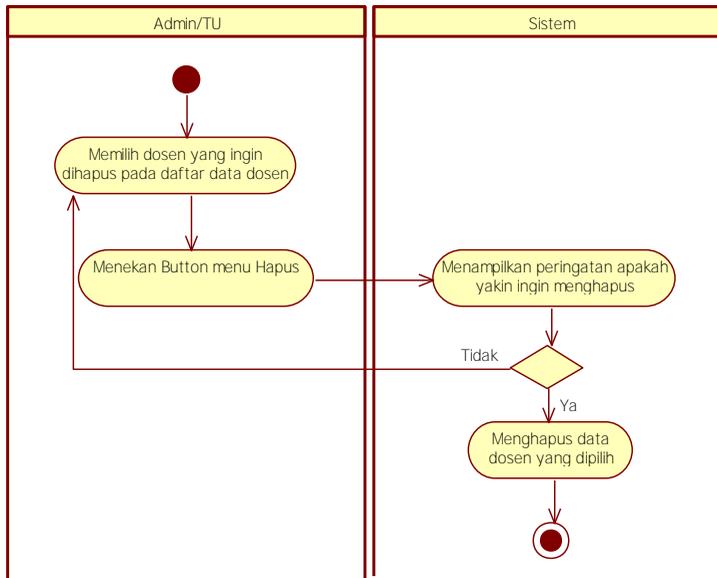
3.1.3.2.14. Kasus Penggunaan Menghapus Data Dosen

Pada kasus penggunaan UC-0014, pengguna dapat menghapus data dosen. Rincian alur kasus penggunaan UC-0014 dijelaskan pada Tabel 3-16 dan diagram aktivitas kasus penggunaan UC-0014 dijelaskan pada Gambar 3-15.

Tabel 3-16 Rincian Alur Kasus Penggunaan UC-0014

Nama	Menghapus Data Dosen
Kode	UC-0014
Aktor	Administrator dan Tata Usaha

Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Dosen
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih dosen yang ingin dihapus pada daftar data dosen 2. Pengguna menekan tombol menu Hapus 3. Sistem menampilkan peringatan apakah yakin ingin menghapus <ol style="list-style-type: none"> A1. Tidak A2. Ya 4. Sistem menghapus data dosen yang dipilih 5. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. Tidak</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 1 <p>A2. Ya</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 4
Kondisi Akhir	Data dosen dihapus dari basis data
Kebutuhan Khusus	-



Gambar 3-15 Diagram Aktivitas Kasus Penggunaan UC-0014

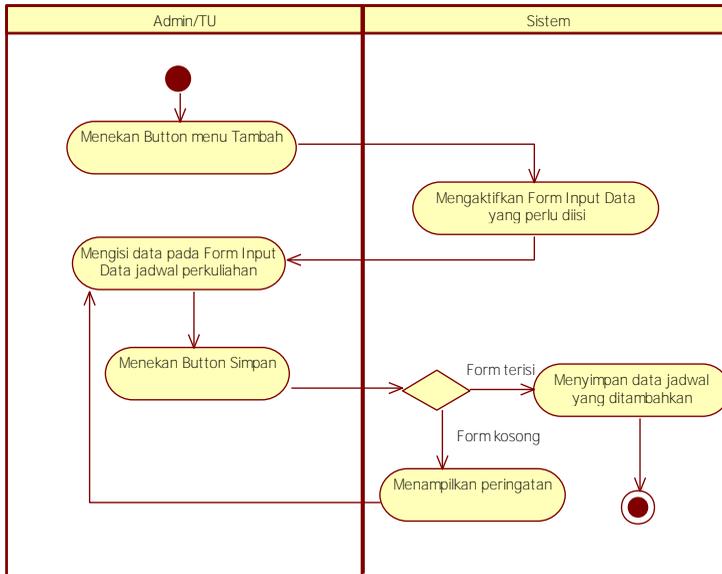
3.1.3.2.15. Kasus Penggunaan Menambah Data Jadwal

Pada kasus penggunaan UC-0015, pengguna dapat menambah data jadwal. Rincian alur kasus penggunaan UC-0015 dijelaskan pada Tabel 3-17 dan diagram aktivitas kasus penggunaan UC-0015 dijelaskan pada Gambar 3-16.

Tabel 3-17 Rincian Alur Kasus Penggunaan UC-0015

Nama	Menambah Data Jadwal
Kode	UC-0015
Aktor	Administrator dan Tata Usaha

Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Jadwal Perkuliahan
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna menekan tombol menu Tambah 2. Sistem mengaktifkan <i>form</i> Input Data yang perlu diisi 3. Pengguna mengisi data pada <i>form</i> Input Data jadwal perkuliahan 4. Pengguna menekan tombol Simpan A1. <i>Form</i> data kosong 5. Sistem menyimpan data jadwal yang ditambahkan 6. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. <i>Form</i> data kosong</p> <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 3
Kondisi Akhir	Data jadwal perkuliahan disimpan ke basis data
Kebutuhan Khusus	-



Gambar 3-16 Diagram Aktivitas Kasus Penggunaan UC-0015

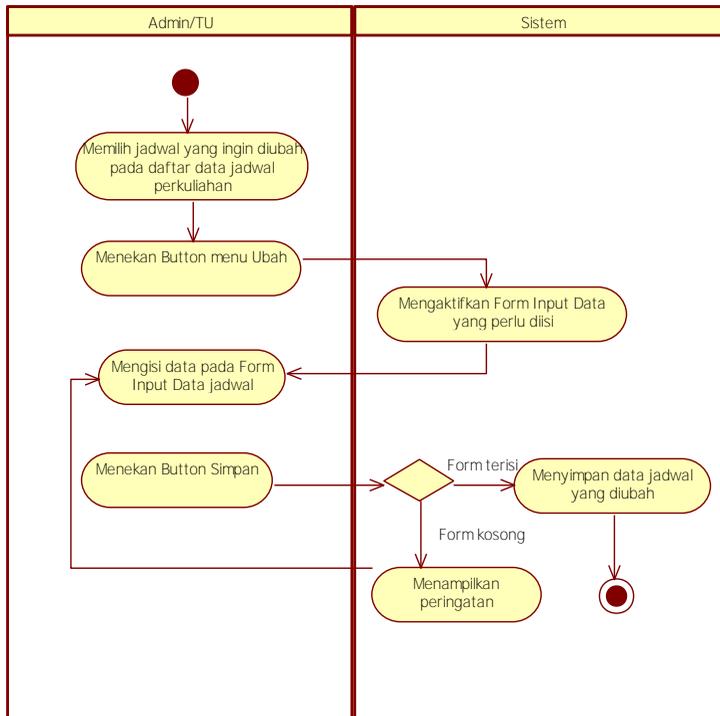
3.1.3.2.16. Kasus Penggunaan Mengubah Data Jadwal

Pada kasus penggunaan UC-0016, pengguna dapat mengubah data jadwal. Rincian alur kasus penggunaan UC-0016 dijelaskan pada Tabel 3-18 dan diagram aktivitas kasus penggunaan UC-0016 dijelaskan pada Gambar 3-17.

Tabel 3-18 Rincian Alur Kasus Penggunaan UC-0016

Nama	Mengubah Data Jadwal
Kode	UC-0016
Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Jadwal Perkuliahan

Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih jadwal yang ingin diubah pada daftar data jadwal perkuliahan 2. Pengguna menekan tombol menu Ubah 3. Sistem mengaktifkan <i>form</i> Input Data yang perlu diisi 4. Pengguna mengisi data pada <i>form</i> Input Data jadwal 5. Pengguna menekan tombol Simpan A1. <i>Form</i> data kosong 6. Sistem menyimpan data jadwal yang diubah 7. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. <i>Form</i> data kosong</p> <ol style="list-style-type: none"> 1. Sistem menampilkan peringatan 2. Berlanjut ke aliran kejadian normal nomor 4
Kondisi Akhir	Data jadwal diubah dari basis data
Kebutuhan Khusus	-



Gambar 3-17 Diagram Aktivitas Kasus Penggunaan UC-0016

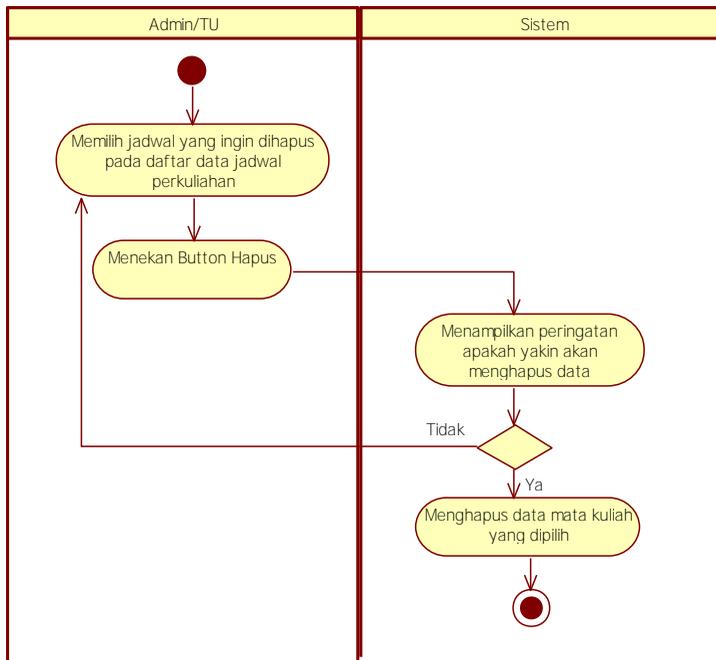
3.1.3.2.17. Kasus Penggunaan Menghapus Data Jadwal

Pada kasus penggunaan UC-0017, pengguna dapat menghapus data jadwal. Rincian alur kasus penggunaan UC-0017 dijelaskan pada Tabel 3-19 dan diagram aktivitas kasus penggunaan UC-0017 dijelaskan pada Gambar 3-18.

Tabel 3-19 Rincian Alur Kasus Penggunaan UC-0017

Nama	Menghapus Data Jadwal
Kode	UC-0017

Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Jadwal Perkuliahan
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih jadwal yang ingin dihapus pada daftar data jadwal perkuliahan 2. Pengguna menekan tombol menu Hapus 3. Sistem menampilkan peringatan apakah yakin akan menghapus data jadwal perkuliahan A1. Tidak A2. Ya 4. Sistem menghapus data mata kuliah yang dipilih. 5. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>B1. Tidak</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 1 <p>B2. Ya</p> <ol style="list-style-type: none"> 2. Berlanjut ke aliran kejadian normal nomor 4
Kondisi Akhir	Data jadwal dihapus dari basis data
Kebutuhan Khusus	-



Gambar 3-18 Diagram Aktivitas Kasus Penggunaan UC-0017

3.1.3.2.18. Kasus Penggunaan Menambah Data Mahasiswa Per Kelas

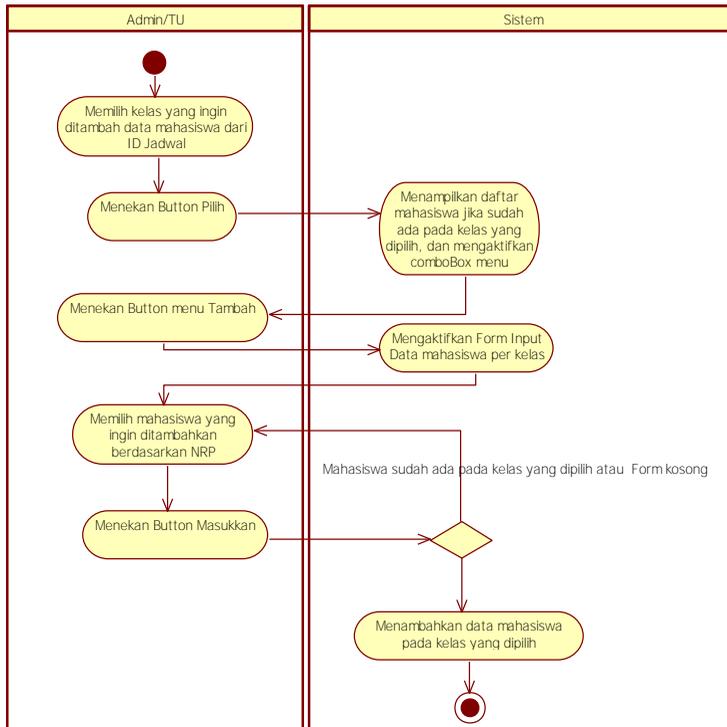
Pada kasus penggunaan UC-0018, pengguna dapat menambah data mahasiswa per kelas. Rincian alur kasus penggunaan UC-0018 dijelaskan pada Tabel 3-20 dan diagram aktivitas kasus penggunaan UC-0018 dijelaskan pada Gambar 3-19.

Tabel 3-20 Rincian Alur Kasus Penggunaan UC-0018

Nama	Menambah Data Mahasiswa Per Kelas
Kode	UC-0018

Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di Data Mahasiswa Per Kelas
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih kelas yang ingin ditambah data mahasiswa dari ID Jadwal 2. Pengguna menekan tombol Pilih 3. Sistem menampilkan daftar mahasiswa jika sudah ada pada kelas yang dipilih, dan mengaktifkan <i>groupBox</i> menu 4. Pengguna menekan tombol menu Tambah 5. Sistem mengaktifkan <i>form</i> Input Data mahasiswa pe kelas 6. Pengguna memilih mahasiswa yang ingin ditambahkan berdasarkan NRP 7. Pengguna menekan tombol Masukkan A1. Mahasiswa sudah ada pada kelas tersebut atau data kosong 8. Sistem menambahkan data mahasiswa pada kelas yang dipilih 9. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. Mahasiswa sudah ada pada kelas tersebut atau data kosong</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 6
Kondisi Akhir	Data mahasiswa ditambahkan pada kelas yang dipilih dan disimpan di basis data

Kebutuhan Khusus	-
------------------	---



Gambar 3-19 Diagram Aktivitas Kasus Penggunaan UC-0018

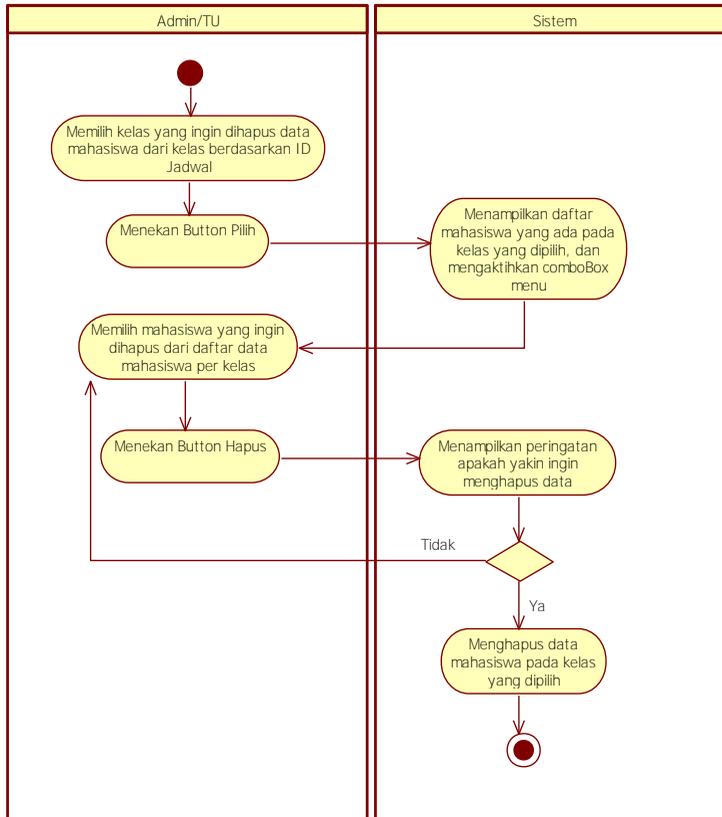
3.1.3.2.19. Kasus Penggunaan Menghapus Data Mahasiswa Per Kelas

Pada kasus penggunaan UC-0019, pengguna dapat menghapus data mahasiswa per kelas. Rincian alur kasus penggunaan UC-0019 dijelaskan pada Tabel 3-21 dan diagram aktivitas kasus penggunaan UC-0019 dijelaskan pada Gambar. 3-20

Tabel 3-21 Rincian Alur Kasus Penggunaan UC-0019

Nama	Menghapus Data Mahasiswa Per Kelas
Kode	UC-0019
Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem dan berada di halaman Data Mahasiswa Per Kelas
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih kelas yang ingin dihapus data mahasiswa dari kelas berdasarkan ID Jadwal 2. Pengguna menekan tombol Pilih 3. Sistem menampilkan daftar mahasiswa yang ada pada kelas tersebut jika sudah ada, dan mengaktifkan <i>groupBox</i> menu 4. Pengguna memilih mahasiswa yang ingin dihapus dari daftar data mahasiswa per kelas 5. Pengguna menekan tombol Hapus 6. Sistem menampilkan peringatan apakah yakin akan menghapus mahasiswa dari daftar mahasiswa per kelas <ol style="list-style-type: none"> A1. Tidak A2. Ya 7. Sistem menghapus data mahasiswa pada kelas yang dipilih 8. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. Tidak</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 4 <p>A2. Ya</p>

	2. Berlanjut ke aliran kejadian normal nomor 7
Kondisi Akhir	Data mahasiswa dihapus dari basis data mahasiswa per kelas
Kebutuhan Khusus	-



Gambar 3-20 Diagram Aktivitas Kasus Penggunaan UC-0019

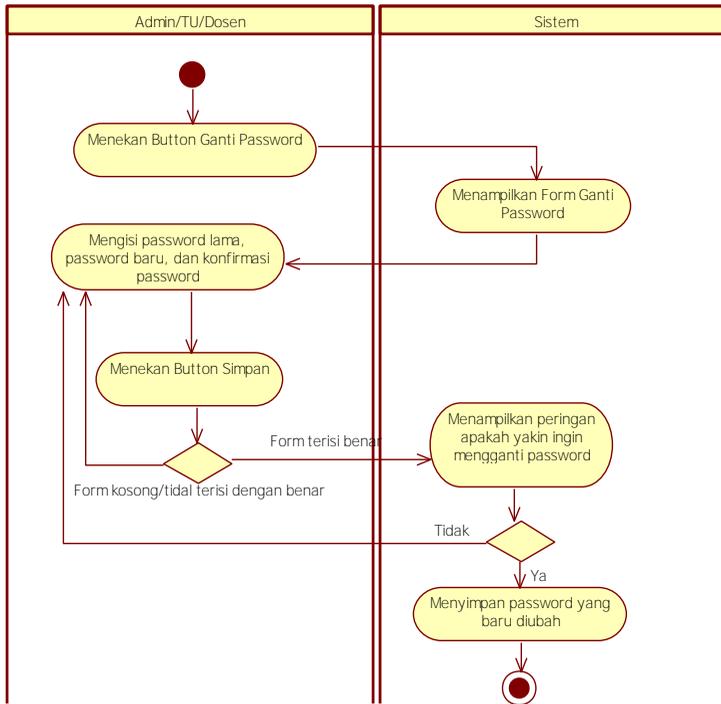
3.1.3.2.20. Kasus Penggunaan Mengubah *Password*

Pada kasus penggunaan UC-0020, pengguna dapat mengubah *password*. Rincian alur kasus penggunaan UC-0020 dijelaskan pada Tabel 3-22 dan diagram aktivitas kasus penggunaan UC-0020 dijelaskan pada Gambar 3-21.

Tabel 3-22 Rincian Alur Kasus Penggunaan UC-0020

Nama	Mengubah <i>Password</i>
Kode	UC-0020
Aktor	Administrator, Tata Usaha, dan Dosen
Kondisi Awal	Pengguna sudah masuk ke dalam sistem
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna menekan tombol Ganti <i>Password</i> 2. Sistem menampilkan Form Ganti <i>Password</i> 3. Pengguna mengisi <i>password</i> lama, <i>password</i> baru, dan konfirmasi <i>password</i> 4. Pengguna menekan tombol Simpan <ol style="list-style-type: none"> A1. <i>Form</i> kosong, <i>password</i> lama salah, <i>password</i> baru sama dengan <i>password</i> lama, atau konfirmasi <i>password</i> berbeda 5. Sistem menampilkan peringatan apakah yakin ingin mengganti <i>password</i> <ol style="list-style-type: none"> A2. Tidak A3. Ya 6. Sistem menyimpan <i>password</i> yang baru diubah 7. Kasus penggunaan berakhir

Aliran Kejadian Alternatif	<p>A1. Jika <i>Form</i> kosong, <i>password</i> lama salah, <i>password</i> baru sama dengan <i>password</i> lama atau konfirmasi <i>password</i> berbeda</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 3 <p>A2. Tidak</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 3 <p>A3. Ya</p> <ol style="list-style-type: none"> 1. Berlanjut ke aliran kejadian normal nomor 6
Kondisi Akhir	<i>Password</i> baru disimpan ke basis data
Kebutuhan Khusus	-



Gambar 3-21 Diagram Aktivitas Kasus Penggunaan UC-0020

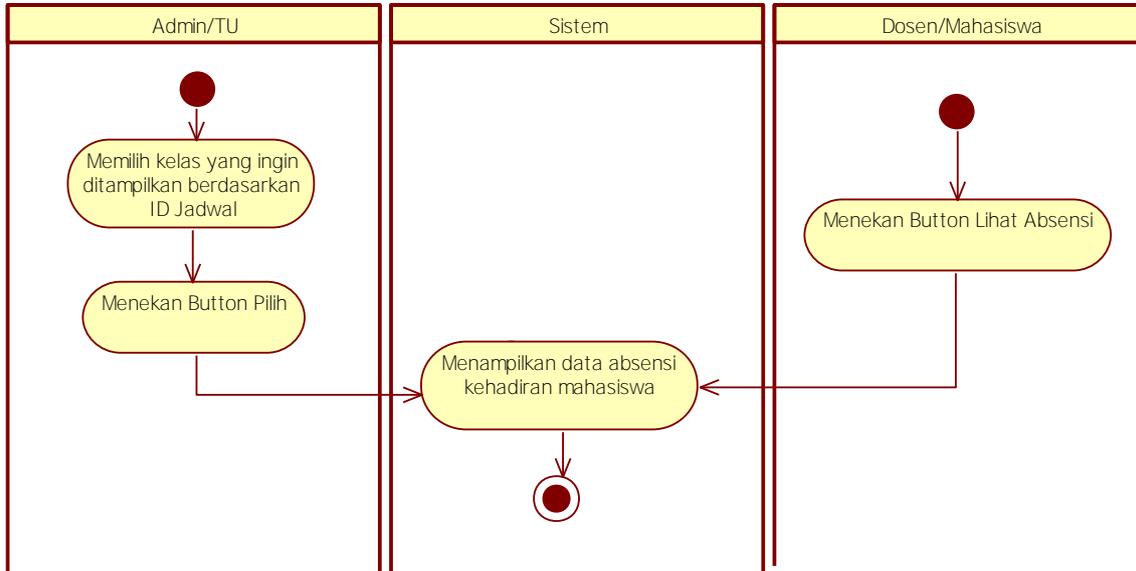
3.1.3.2.21. Kasus Penggunaan Melihat Absensi

Pada kasus penggunaan UC-0021, pengguna dapat melihat absensi. Rincian alur kasus penggunaan UC-0021 dijelaskan pada Tabel 3-23 dan diagram aktivitas kasus penggunaan UC-0021 dijelaskan pada Gambar 3-22.

Tabel 3-23 Rincian Alur Kasus Penggunaan UC-0021

Nama	Melihat Absensi
Kode	UC-0021
Aktor	Semua Aktor

Kondisi Awal	Pengguna sudah masuk ke dalam sistem
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih kelas yang ingin ditampilkan berdasarkan ID Jadwal 2. Administrator/tata usaha menekan tombol Pilih, sedangkan dosen dan mahasiswa menekan tombol Lihat Absensi 3. Sistem menampilkan data absensi mahasiswa 4. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	Untuk mahasiswa dan dosen langsung berlanjut ke aliran kejadian nomor 2
Kondisi Akhir	Data absensi perkuliahan mahasiswa ditampilkan
Kebutuhan Khusus	-



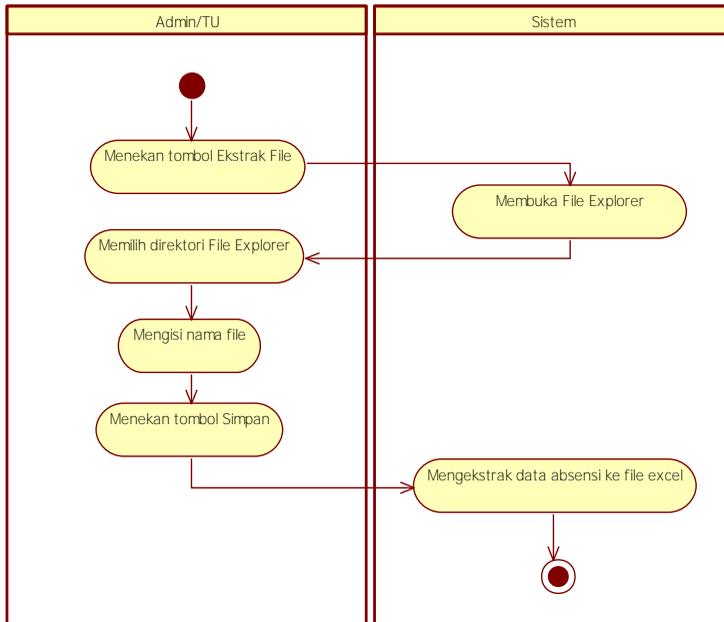
Gambar 3-22 Diagram Aktivitas Kasus Penggunaan UC-0021

3.1.3.2.22. Kasus Penggunaan Mengekstrak Data Absensi

Pada kasus penggunaan UC-0022, pengguna dapat mengekstrak data absensi. Rincian alur kasus penggunaan UC-0022 dijelaskan pada Tabel 3-24 dan diagram aktivitas kasus penggunaan UC-0022 dijelaskan pada Gambar 3-23.

Tabel 3-24 Rincian Alur Kasus Penggunaan UC-0022

Nama	Mengekstrak Data Absensi
Kode	UC-0022
Aktor	Administrator dan Tata Usaha
Kondisi Awal	Pengguna sudah masuk ke dalam sistem
Aliran Kejadian Normal	<ol style="list-style-type: none">1. Pengguna menekan tombol Ekstrak File2. Sistem membuka File Explorer3. Pengguna memilih direktore File Explorer4. Pengguna mengisi nama file5. Pengguna menekan tombol Simpan6. Sistem mengekstrak data absensi ke file Excel
Aliran Kejadian Alternatif	-
Kondisi Akhir	Data absensi diekstrak ke file Excel
Kebutuhan Khusus	-



Gambar 3-23 Diagram Aktivitas Penggunaan UC-0022

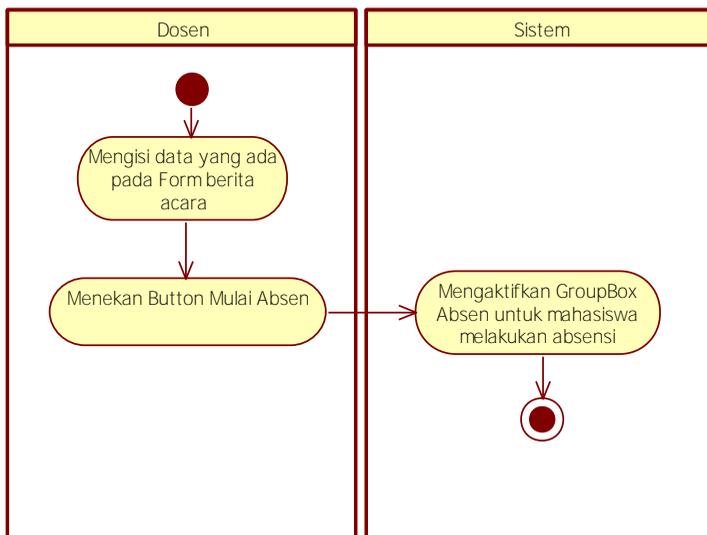
3.1.3.2.23. Kasus Penggunaan Mengisi Berita Acara

Pada kasus penggunaan UC-0023, pengguna dapat mengisi berita acara. Rincian alur kasus penggunaan UC-0023 dijelaskan pada Tabel 3-24 dan diagram aktivitas kasus penggunaan UC-0022 dijelaskan pada Gambar 3-23.

Tabel 3-25 Rincian Alur Kasus Penggunaan UC-0023

Nama	Mengisi Berita Acara
Kode	UC-0023
Aktor	Dosen
Kondisi Awal	Pengguna sudah masuk ke dalam sistem

Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna mengisi data yang ada pada <i>Form</i> berita acara 2. Pengguna menekan tombol Mulai Absen 3. Sistem mengaktifkan <i>groupBox</i> Absen untuk mahasiswa melakukan absensi perkuliahan 4. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	-
Kondisi Akhir	Data berita acara disimpan ke basis data
Kebutuhan Khusus	-



Gambar 3-24 Diagram Aktivitas Kasus Penggunaan UC-0023

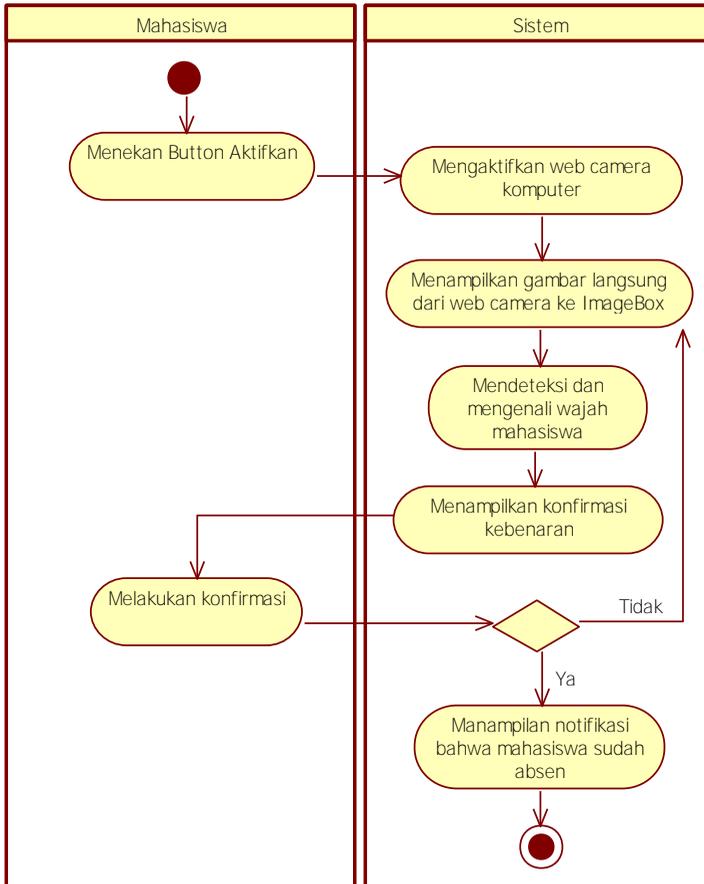
3.1.3.2.24. Kasus Penggunaan Melakukan Absensi

Pada kasus penggunaan UC-0024, pengguna dapat melakukan absensi. Rincian alur kasus penggunaan UC-0024 dijelaskan pada Tabel 3-26 dan diagram aktivitas kasus penggunaan UC-0023 dijelaskan pada Gambar 3-25.

Tabel 3-26 Rincian Alur Kasus Penggunaan UC-0024

Nama	Melakukan Absensi
Kode	UC-0024
Aktor	Mahasiswa
Kondisi Awal	Fitur absensi pada sistem sudah diaktifkan oleh aktor dosen.
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna menekan tombol Aktifkan 2. Sistem mengaktifkan <i>webcam</i> pada komputer 3. Sistem menampilkan gambar langsung dari <i>webcam</i> ke <i>ImageBox</i> 4. Sistem mendeteksi dan mengenali wajah mahasiswa 5. Sistem menampilkan peringatan konfirmasi kebenaran 6. Pengguna melakukan konfirmasi <ol style="list-style-type: none"> A1. Tidak A2. Ya 7. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<ol style="list-style-type: none"> A1. Tidak <ol style="list-style-type: none"> 1. Kembali ke aliran kejadian normal nomor 3 A2. Ya <ol style="list-style-type: none"> 1. Sistem menampilkan notifikasi bahwa mahasiswa sudah melakukan absensi

	2. Kembali ke aliran kejadian normal nomor 6
Kondisi Akhir	Data absensi mahasiswa disimpan ke basis data
Kebutuhan Khusus	<i>Data set</i> wajah yang digunakan harus digunakan dalam kondisi pencahayaan dan resolusi yang sama dengan saat melakukan absensi perkuliahan



Gambar 3-25 Diagram Aktivitas Kasus Penggunaan UC-0024

3.1.4. Identifikasi Pengguna

Identifikasi pengguna mendefinisikan entitas-entitas yang terlibat dan berinteraksi langsung dengan sistem. Berdasarkan deskripsi umum yang telah dijelaskan pada bagian sebelumnya,

maka pengguna yang akan menggunakan aplikasi ini ada empat, yaitu administrator, tata usaha, dosen, dan mahasiswa.

3.1.4.1. Administrator

Administrator merupakan pengguna yang bertanggungjawab terkait dengan data tata usaha, dosen, mahasiswa, mata kuliah, kelas, dan jadwal perkuliahan.

3.1.4.2. Tata Usaha

Tata Usaha merupakan pengguna yang bertanggungjawab dalam terkait dengan data dosen, mahasiswa, mata kuliah, kelas, dan jadwal perkuliahan

3.1.4.3. Dosen

Dosen merupakan pengguna yang mengatur fitur pada saat perkuliahan, seperti pengisian berita acara perkuliahan dan mengaktifkan fitur absensi pada *groupBox* Absen yang digunakan mahasiswa untuk melakukan absensi perkuliahan.

3.1.4.4. Mahasiswa

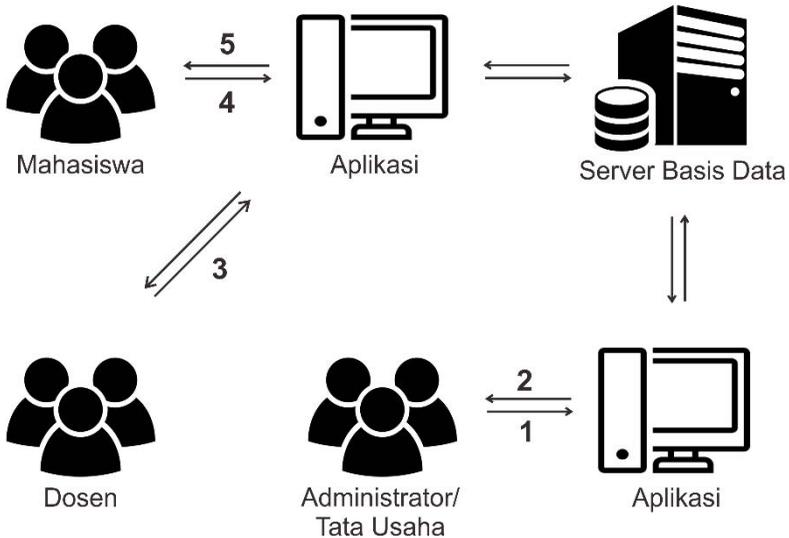
Mahasiswa merupakan pengguna yang hanya memiliki akses untuk melakukan dan melihat absensi.

3.2. Perancangan Sistem

Pada subbab ini membahas perancangan dari aplikasi absensi perkuliahan mahasiswa dengan pengenalan wajah. Tahap perancangan dibagi menjadi beberapa bagian yaitu perancangan arsitektur, perancangan basis data, dan perancangan antarmuka.

3.2.1. Perancangan Arsitektur

Pada subbab ini membahas tentang perancangan arsitektur aplikasi yang dibangun. Arsitektur aplikasi dapat dilihat pada Gambar 3-26 di bawah ini



Gambar 3-26 Arsitektur Aplikasi

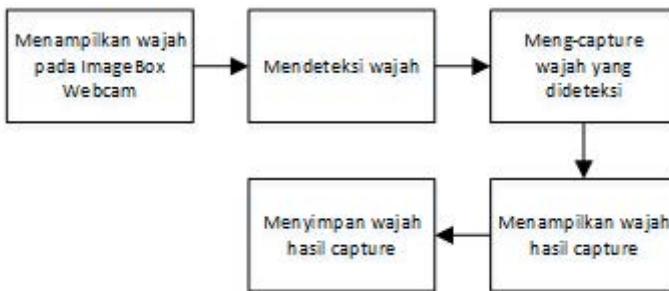
Berikut penjelasan arsitektur sistem pada Gambar 3-26.

1. Administrator atau tata usaha mengelola data-data yang dibutuhkan sistem.
2. Administrator atau tata usaha mendapatkan data rekapitulasi kehadiran, file ekstraksi data kehadiran, dan data yang dikelola.
3. Dosen mengisi berita acara pada suatu perkuliahan.
4. Setelah dosen mengisi berita acara, mahasiswa memulai absensi dengan menghadapkan wajah ke depan komputer sehingga webcam dapat menangkap wajah mahasiswa.

5. Ketika aplikasi mengenali wajah mahasiswa maka aplikasi akan memberikan umpan balik berupa konfirmasi apakah benar data yang dikenali oleh sistem adalah mahasiswa yang bersangkutan. Jika tidak maka aplikasi akan kembali menangkap wajah mahasiswa.

3.2.2. Proses Masukan Data Wajah

Pada subbab ini membahas tentang proses masukan data wajah. Proses masukan data wajah pada sistem dapat dilihat pada Gambar 3-27 di bawah ini

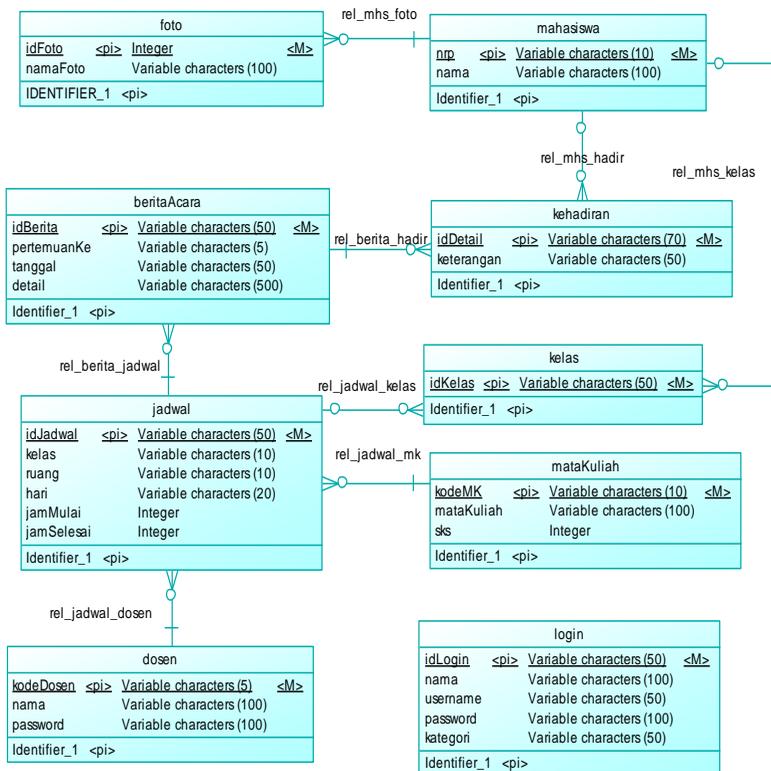


Gambar 3-27 Proses Masukan Data Wajah

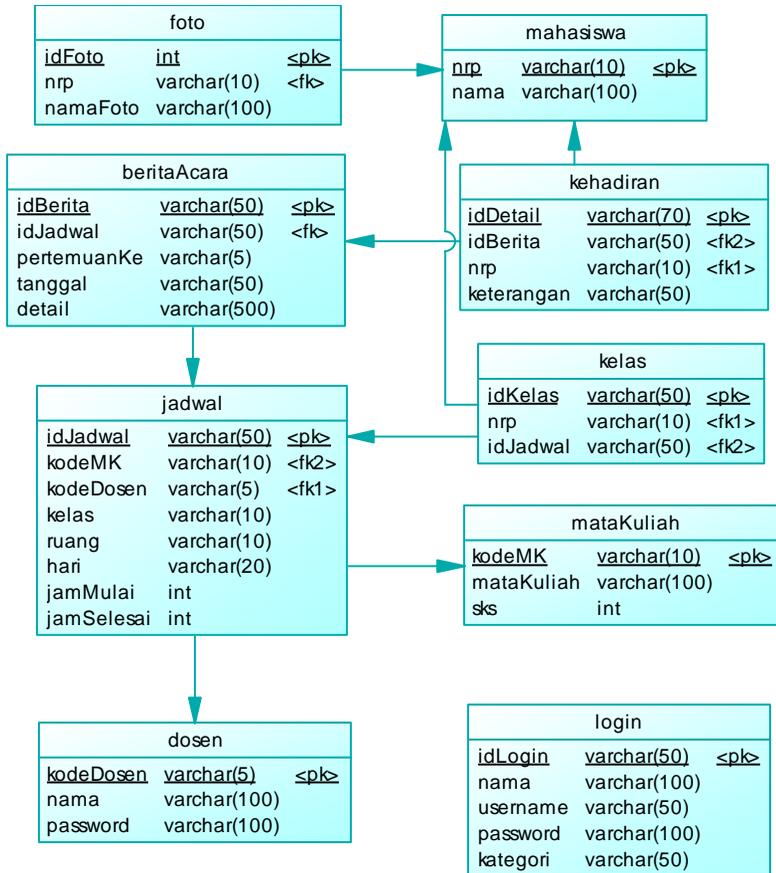
Pada proses masukan data wajah, pengguna pertama memilih metode masukan data wajah yaitu dari *file* foto atau secara langsung melalui *webcam*. Kemudian akan ditampilkan pada *ImageBox Webcam*. Sistem akan mendeteksi wajah dengan metode *Haarcascade* dari *library* Emgu CV. Wajah yang dideteksi adalah wajah yang menghadap ke depan, pada pencahayaan yang baik, dan wajah yang tidak terhalangi oleh kacamata, masker, ataupun objek lainnya. Kemudian pengguna meng-*capture* wajah yang dideteksi dan sistem akan menampilkan wajah hasil *capture*. Setelah itu, data wajah akan disimpan pada basis data. Data wajah yang disimpan pada basis data akan digunakan untuk mencocokkan wajah mahasiswa saat melakukan pencatatan absensi.

3.2.3. Perancangan Basis Data

Pada subbab ini akan dijelaskan bagaimana rancangan basis data pada aplikasi absensi perkuliahan mahasiswa dengan pengenalan wajah. Basis data yang dibangun pada aplikasi ini menggunakan sistem manajemen basis data relasional MySQL. Data-data yang disimpan dalam MySQL diantaranya data absensi mahasiswa, data mata kuliah, data jadwal perkuliahan, data mahasiswa, data foto mahasiswa, data mahasiswa per kelas, data dosen, serta data *login* yang digunakan untuk administrator dan tata usaha. Perancangan basis data dimodelkan dalam diagram model konsep basis data atau *Conceptual Data Model* (CDM) dan diagram model fisik basis data atau *Physical Data Model* (PDM). Diagram-diagram tersebut dapat dilihat pada Gambar 3-28 dan Gambar 3-29.



Gambar 3-28 Diagram CDM



Gambar 3-29 Diagram PDM

3.2.3.1. Rancangan Tabel foto

Tabel ini digunakan untuk menyimpan data foto mahasiswa. Satu mahasiswa bisa memiliki lebih dari satu foto. Tabel ini berhubungan dengan tabel mahasiswa. Penjelasan tabel mata kuliah dapat dilihat pada Tabel 3-27.

Tabel 3-27 Rancangan Tabel foto

Nama Atribut	Tipe Data	Keterangan
idFoto	Integer	Merupakan <i>primary key</i> dari tabel foto, yang menyimpan foto-foto mahasiswa.
nrp	Varchar (10)	Merupakan <i>foreign key</i> yang berasal dari tabel mahasiswa.
namaFoto	Varchar (100)	Merupakan atribut yang menyimpan nama foto dari mahasiswa.

3.2.3.2. Rancangan Tabel mahasiswa

Tabel ini digunakan untuk menyimpan data mahasiswa. Tabel ini berhubungan dengan tabel foto dan kehadiran. Penjelasan tabel mata kuliah dapat dilihat pada Tabel 3-28.

Tabel 3-28 Rancangan Tabel mahasiswa

Nama Atribut	Tipe Data	Keterangan
nrp	Varchar (10)	Merupakan <i>primary key</i> dari tabel mahasiswa, yang menyimpan data mahasiswa.
nama	Varchar (100)	Merupakan atribut yang menyimpan nama mahasiswa.

3.2.3.3. Rancangan Tabel kehadiran

Tabel ini digunakan untuk menyimpan data absensi mahasiswa. Tabel ini berhubungan dengan tabel mahasiswa dan beritaAcara. Penjelasan tabel mata kuliah dapat dilihat pada Tabel 3-29.

Tabel 3-29 Rancangan Tabel kehadiran

Nama Atribut	Tipe Data	Keterangan
idDetail	Varchar (70)	Merupakan <i>primary key</i> dari tabel kehadiran, yang menyimpan data mahasiswa.
idBerita	Varchar (50)	Merupakan <i>foreign key 2</i> yang berasal dari tabel beritaAcara
nrp	Varchar (10)	Merupakan <i>foreign key 1</i> yang berasal dari tabel mahasiswa.
keterangan	Varchar (50)	Merupakan atribut yang menyimpan status hadir mahasiswa.

3.2.3.4. Rancangan Tabel beritaAcara

Tabel ini digunakan untuk menyimpan data berita acara perkuliahan mahasiswa. Tabel ini berhubungan dengan tabel jadwal dan tabel kehadiran. Penjelasan tabel mata kuliah dapat dilihat pada Tabel 3-30.

Tabel 3-30 Rancangan Tabel beritaAcara

Nama Atribut	Tipe Data	Keterangan
idBerita	Varchar (50)	Merupakan <i>primary key</i> dari tabel beritaAcara, yang menyimpan data berita acara perkuliahan mahasiswa.
idJadwal	Varchar (50)	Merupakan <i>foreign key</i> yang berasal dari tabel jadwal.
pertemuanKe	Varchar (5)	Merupakan atribut yang menyimpan data pertemuan perkuliahan.

Nama Atribut	Tipe Data	Keterangan
tanggal	Varchar (50)	Merupakan atribut yang menyimpan tanggal perkuliahan pada setiap satu pertemuan.
detail	Varchar (500)	Merupakan atribut yang menyimpan materi perkuliahan yang disampaikan pada setiap pertemuan.

3.2.3.5. Rancangan Tabel jadwal

Tabel ini digunakan untuk menyimpan daftar mata kuliah. Tabel ini berhubungan dengan tabel jadwal perkuliahan. Penjelasan tabel mata kuliah dapat dilihat pada Tabel 3-31.

Tabel 3-31 Rancangan Tabel jadwal

Nama Atribut	Tipe Data	Keterangan
idJadwal	Varchar (50)	Merupakan <i>primary key</i> dari tabel jadwal, yang menyimpan data jadwal per mata kuliah perkuliahan mahasiswa.
kodeDosen	Varchar (5)	Merupakan <i>foreign key 1</i> yang berasal dari tabel dosen.
kodeMK	Varchar (10)	Merupakan <i>foreign key 2</i> yang berasal dari tabel mata kuliah.
kelas	Varchar (10)	Merupakan atribut yang menyimpan kelas dari setiap mata kuliah.

Nama Atribut	Tipe Data	Keterangan
ruang	Varchar (10)	Merupakan atribut yang menyimpan ruangan dari setiap mata kuliah perkelas.
hari	Varchar (20)	Merupakan atribut yang menyimpan hari perkuliahan dari setiap mata kuliah.
jamMulai	Int	Merupakan atribut yang menyimpan waktu mulai dari setiap jadwal per mata kuliah.
jamSelesai	Int	Merupakan atribut yang menyimpan waktu selesai dari setiap jadwal per mata kuliah.

3.2.3.6. Rancangan Tabel kelas

Tabel ini digunakan untuk menyimpan data mahasiswa per kelas. Tabel ini berhubungan dengan tabel jadwal dan tabel mahasiswa. Penjelasan tabel mata kuliah dapat dilihat pada Tabel 3-32.

Tabel 3-32 Rancangan Tabel kelas

Nama Atribut	Tipe Data	Keterangan
idKelas	Varchar (50)	Merupakan <i>primary key</i> dari tabel kelas, yang menyimpan data mahasiswa per kelas.
nrp	Varchar (10)	Merupakan <i>foreign key 1</i> yang berasal dari tabel mahasiswa
idJadwal	Varchar (50)	Merupakan <i>foreign key 2</i> yang berasal dari tabel jadwal

3.2.3.7. Rancangan Tabel mataKuliah

Tabel ini digunakan untuk menyimpan data mata kuliah. Tabel ini berhubungan dengan tabel jadwal. Penjelasan tabel mata kuliah dapat dilihat pada Tabel 3-33.

Tabel 3-33 Rancangan Tabel mataKuliah

Nama Atribut	Tipe Data	Keterangan
kodeMK	Varchar (10)	Merupakan <i>primary key</i> dari tabel mataKuliah, yang menyimpan data mata kuliah.
mataKuliah	Varchar (100)	Merupakan atribut yang menyimpan nama-nama mata kuliah.
Sks	Integer	Merupakan atribut yang menyimpan jumlah sks dari setiap mata kuliah.

3.2.3.8. Rancangan Tabel dosen

Tabel ini digunakan untuk menyimpan data dosen. Tabel ini berhubungan dengan tabel jadwal. Penjelasan tabel mata kuliah dapat dilihat pada Tabel 3-34.

Tabel 3-34 Rancangan Tabel dosen

Nama Atribut	Tipe Data	Keterangan
kodeDosen	Varchar (5)	Merupakan <i>primary key</i> dari tabel dosen, yang menyimpan data dosen.
Nama	Varchar (100)	Merupakan atribut yang menyimpan nama dosen.
password	Varchar (100)	Merupakan atribut yang menyimpan <i>password</i> dosen

Nama Atribut	Tipe Data	Keterangan
		untuk <i>login</i> ke halaman absensi.

3.2.3.9. Rancangan Tabel login

Tabel ini digunakan untuk menyimpan data *login* dari administrator dan tata usaha. Penjelasan tabel mata kuliah dapat dilihat pada Tabel 3-35.

Tabel 3-35 Rancangan Tabel login

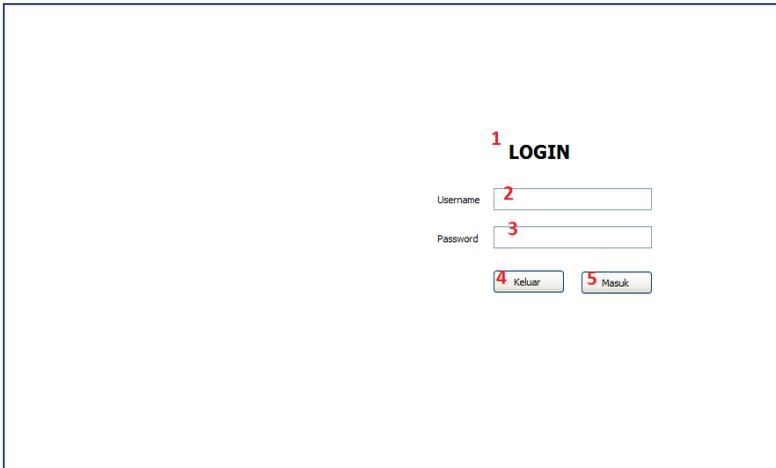
Nama Atribut	Tipe Data	Keterangan
IdLogin	Varchar (50)	Merupakan <i>primary key</i> dari tabel login.
Nama	Varchar (100)	Merupakan atribut yang menyimpan nama administrator dan tata usaha.
username	Varchar (50)	Merupakan atribut yang menyimpan <i>username</i> administrator dan tata usaha untuk <i>login</i> .
password	Varchar (100)	Merupakan atribut yang menyimpan <i>password</i> administrator dan tata usaha.
Kategori	Varchar (50)	Merupakan atribut yang menyimpan kategori dari <i>user login</i> . Dibedakan menjadi dua kategori, yaitu administrator dan tata usaha.

3.2.4. Perancangan Antarmuka Pengguna

Pada subbab ini membahas mengenai perancangan antarmuka yang akan dibuat pada aplikasi. Rancangan antarmuka disesuaikan dengan kebutuhan fungsional.

3.2.4.1. Rancangan Antarmuka Halaman Login

Gambar 3-30 merupakan gambar rancangan antarmuka halaman Login untuk administrator, tata usaha, dan dosen. Pada halaman ini terdapat dua *TextBox* Username dan Password serta tombol Keluar dan Masuk.



The image shows a login form with the following elements:

- 1. A red number '1' next to the word 'LOGIN'.
- 2. A red number '2' next to the 'Username' label and its corresponding text box.
- 3. A red number '3' next to the 'Password' label and its corresponding text box.
- 4. A red number '4' next to the 'Keluar' button.
- 5. A red number '5' next to the 'Masuk' button.

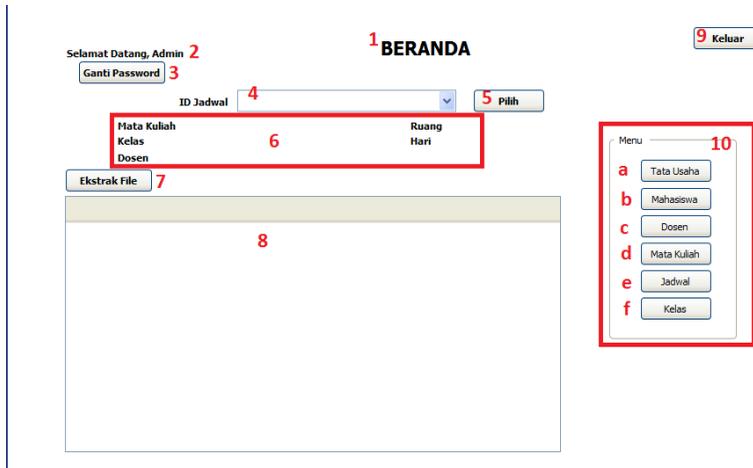
Gambar 3-30 Rancangan Antarmuka Halaman Login

Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-30.

1. *Label* Login yang menandakan bahwa pengguna sedang berada di halaman Login.
2. *TextBox* Username yang digunakan untuk mengisi *username* administrator, tata usaha, atau dosen.
3. *TextBox* Password yang digunakan untuk mengisi *password* administrator atau tata usaha.
4. Tombol Keluar yang digunakan untuk keluar dari aplikasi.
5. Tombol Masuk yang digunakan untuk masuk ke dalam sistem.

3.2.4.2. Rancangan Antarmuka Halaman Beranda untuk Administrator

Gambar 3-31 merupakan gambar rancangan antarmuka halaman Beranda untuk administrator. Pada halaman ini terdapat *ComboBox* ID Jadwal. Satu kotak *DataGridView* untuk melihat data absensi perkuliahan mahasiswa pada suatu kelas berdasarkan ID Jadwal yang dipilih dari *ComboBox*. Satu *GroupBox* yang berisi enam tombol, yaitu Tata Usaha, Mahasiswa, Dosen, Mata Kuliah, Jadwal, dan Kelas. Serta ada tombol Ganti Password, Keluar, Ekstrak File, dan Pilih.



Gambar 3-31 Rancangan Antarmuka Halaman Beranda untuk Admin

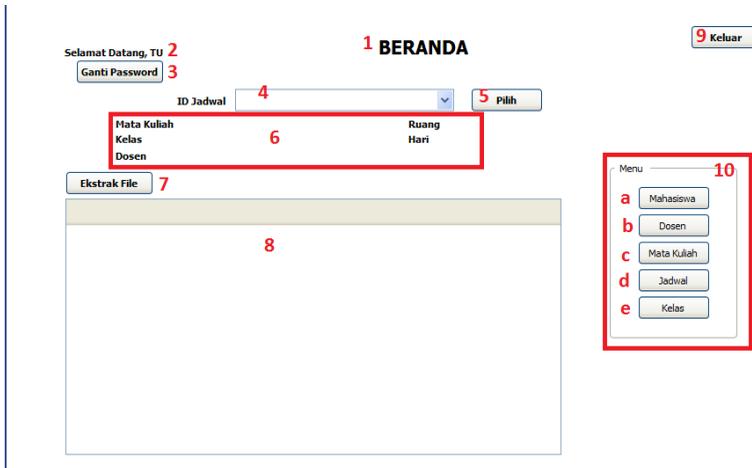
Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-31.

1. *Label* Beranda yang menandakan bahwa pengguna sedang berada di halaman Beranda.
2. *Label* Admin merupakan nama akun yang sedang mengakses sistem.

3. Tombol Ganti Password yang digunakan untuk masuk ke Form Ganti Password.
4. *ComboBox* ID Jadwal yang digunakan untuk memilih Kelas perkuliahan yang ingin ditampilkan berdasarkan ID Jadwal.
5. Tombol Pilih yang digunakan untuk mengeksekusi dari *ComboBox* ID Jadwal yang telah dipilih.
6. Label Mata Kuliah, Kelas, Dosen, Ruang, dan Hari digunakan untuk menampilkan detail data perkuliahan yang dipilih dari ID Jadwal.
7. Tombol Ekstrak File yang digunakan untuk mengekstrak data yang ditampilkan pada *DataGridView*.
8. *DataGridView* yang digunakan untuk menampilkan data absensi perkuliahan yang telah dilakukan pada satu mata kuliah.
9. Tombol Keluar untuk keluar dari aplikasi dan kembali ke halaman Login.
10. *GroupBox* menu yang berisi tombol beberapa menu yang dapat dipilih, yaitu:
 - a. Tombol Tata Usaha yang digunakan untuk masuk ke halaman Data Tata Usaha
 - b. Tombol Mahasiswa yang digunakan untuk masuk ke halaman Data Mahasiswa.
 - c. Tombol Dosen yang digunakan untuk masuk ke halaman Data Dosen
 - d. Tombol Mata Kuliah yang digunakan untuk masuk ke halaman Data Mata Kuliah
 - e. Tombol Jadwal yang digunakan untuk masuk ke halaman Data Jadwal Perkuliahan
 - f. Tombol Kelas yang digunakan untuk masuk ke halaman Data Mahasiswa Per Kelas

3.2.4.3. Rancangan Antarmuka Halaman Beranda untuk Tata Usaha

Gambar 3-32 merupakan gambar rancangan antarmuka halaman Beranda untuk tata usaha. Pada halaman ini terdapat *ComboBox* ID Jadwal. Satu kotak *DataGridView* untuk melihat data absensi perkuliahan mahasiswa pada suatu kelas berdasarkan ID Jadwal yang dipilih dari *ComboBox*. Satu *GroupBox* yang berisi lima tombol, yaitu Mahasiswa, Dosen, Mata Kuliah, Jadwal, dan Kelas. Serta ada tombol Ganti Password, Keluar, Ekstrak File, dan Pilih.



Gambar 3-32 Rancangan Antarmuka Halaman Beranda untuk Tata Usaha

Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-32.

1. *Label* Beranda yang menandakan bahwa pengguna sedang berada di halaman Beranda.
2. *Label* TU merupakan nama akun yang sedang mengakses sistem.

3. Tombol Ganti Password yang digunakan untuk masuk ke *form* Ganti Password.
4. *ComboBox* ID Jadwal yang digunakan untuk memilih Kelas perkuliahan yang ingin ditampilkan berdasarkan ID Jadwal
5. Tombol Pilih yang digunakan untuk mengeksekusi dari *ComboBox* ID Jadwal yang telah dipilih.
6. *Label* Mata Kuliah, Kelas, Dosen, Ruang, dan Hari digunakan untuk menampilkan detail data perkuliahan yang dipilih dari ID Jadwal.
7. Tombol Ekstrak File yang digunakan untuk mengekstrak data yang ditampilkan pada *DataGridView*.
8. *DataGridView* yang digunakan untuk menampilkan data absensi perkuliahan yang telah dilakukan pada satu mata kuliah.
9. Tombol Keluar untuk keluar dari aplikasi dan kembali ke halaman Login.
10. *GroupBox* menu yang berisi tombol beberapa menu yang dapat dipilih, yaitu:
 - a. Tombol Mahasiswa yang digunakan untuk masuk ke halaman Data Mahasiswa.
 - b. Tombol Dosen yang digunakan untuk masuk ke halaman Data Dosen.
 - c. Tombol Mata Kuliah yang digunakan untuk masuk ke halaman Data Mata Kuliah.
 - d. Tombol Jadwal yang digunakan untuk masuk ke halaman Data Jadwal Perkuliahan.
 - e. Tombol Kelas yang digunakan untuk masuk ke halaman Data Mahasiswa Per Kelas.

3.2.4.4. Rancangan Antarmuka Ganti Password

Gambar 3-33 merupakan gambar rancangan antarmuka halaman Ganti Password. Pada halaman ini terdapat *Label* nama pengguna yang sedang *login* dalam sistem, tiga kotak *TextBox*

untuk melihat mengisi *password* lama, *password* baru, dan konfirmasi *password* baru. Serta ada tombol Batal dan Simpan.

The image shows a user interface for changing a password. It is enclosed in a blue rectangular border. At the top left, the text "Admin/TU" is followed by a red number "1". Below this, there are three text input fields. The first is labeled "Password Lama" and has a red number "2" inside it. The second is labeled "Password Baru" and has a red number "3" inside it. The third is labeled "Konfirmasi Password" and has a red number "4" inside it. At the bottom, there are two buttons: "5 Batal" and "6 Simpan", both with red numbers "5" and "6" respectively.

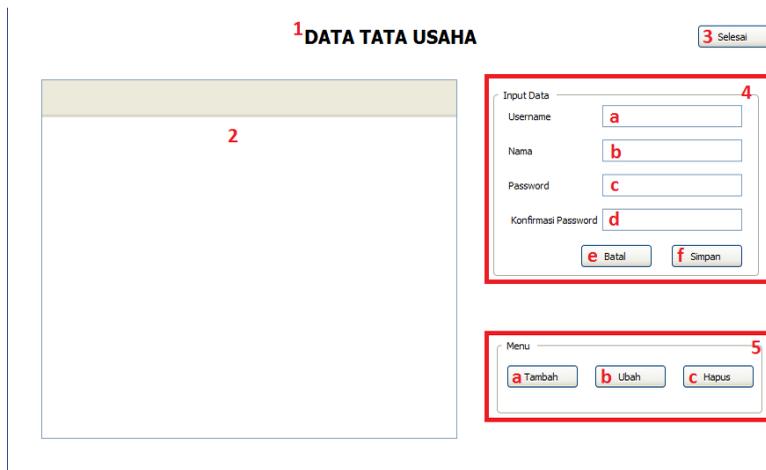
Gambar 3-33 Rancangan Antarmuka Ganti Password

Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-33.

1. *Label* nama pengguna yang sedang *login* dalam sistem.
2. *TextBox* Password Lama yang digunakan untuk mengisi *password* lama pengguna.
3. *TextBox* Password Baru yang digunakan untuk mengisi *password* baru pengguna.
4. *TextBox* Konfirmasi *Password* yang digunakan untuk memastikan bahwa *password* baru yang diisi benar.
5. Tombol Batal yang digunakan untuk membatalkan penggantian *password*.
6. Tombol Simpan yang digunakan untuk menyimpan *password* baru pengguna.

3.2.4.5. Rancangan Antarmuka Halaman Tata Usaha

Gambar 3-34 merupakan gambar rancangan antarmuka halaman Tata Usaha. Pada halaman ini terdapat satu kotak *DataGridView* untuk melihat data Tata Usaha. Dua *GroupBox* yaitu *GroupBox* Input Data dan *GroupBox* Menu. Pada *GroupBox* Input Data berisi tiga *TextBox* Username, Nama, dan Password, Sedangkan *GroupBox* menu berisi tiga tombol, yaitu Tambah, Ubah, dan Hapus. Serta ada tombol Keluar.



Gambar 3-34 Rancangan Antarmuka Halaman Data Tata Usaha

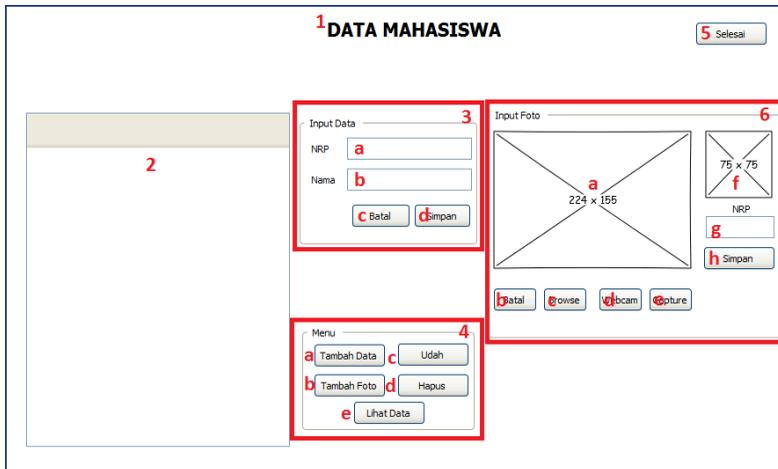
Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-34.

1. *Label* Data Tata Usaha yang menandakan bahwa pengguna sedang berada di halaman Data Tata Usaha.
2. *DataGridView* yang digunakan untuk menampilkan data tata usaha.
3. Tombol Selesai untuk keluar dari Halaman Tata Usaha dan kembali ke halaman Beranda.
4. *GroupBox* Input Data yang berisi

- a. *TextBox* Username yang digunakan untuk mengisi data *username* tata usaha.
 - b. *TextBox* Nama yang digunakan untuk mengisi data nama tata usaha.
 - c. *TextBox* Password yang digunakan untuk mengisi data *password* tata usaha
 - d. *TextBox* Konfirmasi Password yang digunakan untuk memastikan bahwa *password* baru yang diisi benar
 - e. Tombol Batal yang digunakan untuk membatalkan perubahan *password*
 - f. Tombol Simpan yang digunakan untuk menyimpan perubahan *password*
5. *GroupBox* Menu yang berisi
- a. Tombol Tambah yang digunakan untuk menambah data tata usaha.
 - b. Tombol Ubah yang digunakan untuk mengubah data tata usaha.
 - c. Tombol Hapus yang digunakan untuk menghapus data tata usaha.

3.2.4.6. Rancangan Antarmuka Halaman Mahasiswa

Gambar 3-35 merupakan gambar rancangan antarmuka halaman Mahasiswa. Pada halaman ini satu kotak *DataGridView* untuk melihat data Mahasiswa. Tiga *GroupBox* yaitu *GroupBox* Input Data, *GroupBox* Input Foto, dan *GroupBox* Menu. Pada *GroupBox* Input Data berisi dua *TextBox* yaitu *TextBox* NRP dan Nama. Dan pada *GroupBox* Input Foto berisi dua *ImageBox*, tombol Batal, tombol Browse, tombol Webcam, tombol Capture, tombol Simpan dan *TextBoxt* NRP. Sedangkan *GroupBox* menu berisi lima tombol, yaitu Tambah Data, Tambah Foto, Lihat Data, Ubah, dan Hapus. Serta ada tombol Selesai.



Gambar 3-35 Rancangan Antarmuka Halaman Data Mahasiswa

Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-35.

1. *Label* Data Mahasiswa yang menandakan bahwa pengguna sedang berada di halaman Data Mahasiswa.
2. *DataGridView* yang digunakan untuk menampilkan data mahasiswa.
3. *GroupBox* Input Data yang berisi
 - a. *TextBox* NRP yang digunakan untuk mengisi data NRP mahasiswa.
 - b. *TextBox* Nama yang digunakan untuk mengisi data nama mahasiswa.
 - c. Tombol Batal yang digunakan untuk membatalkan perintah Tambah Data.
 - d. Tombol Simpan yang digunakan untuk menyimpan data nama dan NRP mahasiswa yang telah diisikan pada *TextBox* Nama dan NRP.
4. *GroupBox* Menu yang berisi
 - a. Tombol Tambah Data yang digunakan untuk menambah data Mahasiswa

- b. Tombol Tambah Foto yang digunakan untuk menambah foto mahasiswa yang dipilih.
 - c. Tombol Ubah yang digunakan untuk mengubah data mahasiswa yang dipilih.
 - d. Tombol Hapus yang digunakan untuk menghapus data mahasiswa dipilih.
 - e. Tombol Lihat Data yang digunakan untuk melihat data mahasiswa yang dipilih.
5. Tombol Selesai untuk keluar dari Halaman Data Mahasiswa dan kembali ke halaman Beranda.
 6. *GroupBox Input* Foto yang berisi
 - a. *ImageBox* Webcam yang digunakan untuk mengambil foto dari *webcam* komputer.
 - b. Tombol Batal yang digunakan untuk membatalkan perintah Tambah Foto.
 - c. Tombol Browse yang digunakan untuk mengunggah *file* foto dari *file explorer* Windows.
 - d. Tombol Webcam yang digunakan untuk menambahkan foto dari *webcam*.
 - e. Tombol Capture yang digunakan untuk mengambil gambar wajah sesuai dengan ukuran wajah yang terdeteksi.
 - f. *ImageBox* Wajah yang digunakan untuk menampilkan wajah yang telah di-*capture*.
 - g. *TextBox* NRP berisi NRP dari mahasiswa yang ingin ditambahkan foto.
 - h. Tombol Simpan yang digunakan untuk menyimpan wajah dalam *ImageBox* Wajah dan *TextBox* NRP.

3.2.4.7. Rancangan Antarmuka Form Lihat Data Mahasiswa

Gambar 3-36 merupakan gambar rancangan antarmuka Lihat Data Mahasiswa. Pada halaman ini terdapat *Label* NRP dan

Nama. Dan ada *ImageBox* untuk melihat foto mahasiswa, serta ada tombol *previous*, *next*, Hapus, dan OK

Gambar 3-36 Rancangan Antarmuka Form Lihat Data Mahasiswa

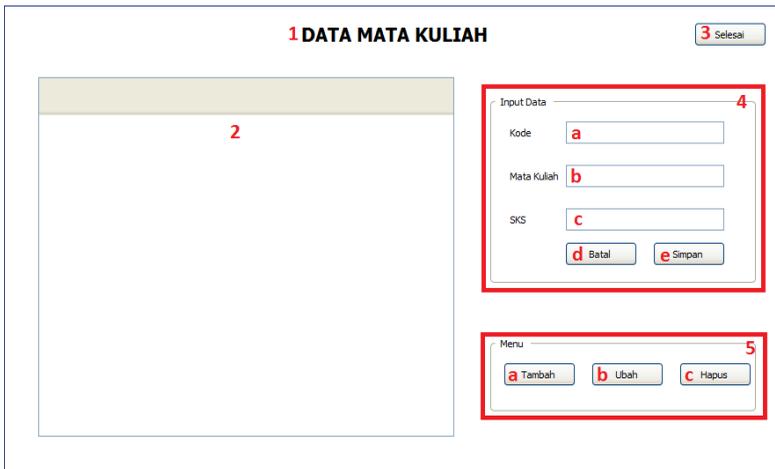
Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-36.

1. *Label* NRP merupakan NRP mahasiswa yang sedang dilihat.
2. *Label* Nama merupakan nama mahasiswa yang sedang dilihat.
3. *ImageBox* yang digunakan untuk menampilkan foto mahasiswa yang sedang dilihat.
4. Tombol *previous* yang digunakan untuk melihat foto sebelumnya dari data mahasiswa yang sedang dilihat
5. Tombol Hapus yang digunakan untuk menghapus foto yang sedang ditampilkan pada *ImageBox*.

6. Tombol *next* yang digunakan untuk melihat foto selanjutnya dari data mahasiswa yang sedang dilihat.
7. Tombol OK yang digunakan untuk menutup *Form* Lihat Data Mahasiswa.

3.2.4.8. Rancangan Antarmuka Halaman Mata Kuliah

Gambar 3-37 merupakan gambar rancangan antarmuka halaman Data Mata Kuliah. Pada halaman ini terdapat satu kotak *DataGridView* untuk melihat data mata kuliah. Dua *GroupBox* yaitu *GroupBox* Input Data dan *GroupBox* Menu. Pada *GroupBox* Input Data berisi tiga *TextBox* yaitu Kode, Mata Kuliah, dan SKS. Sedangkan *GroupBox* menu berisi tiga tombol, yaitu Tambah, Ubah, dan Hapus. Serta ada tombol Selesai.



Gambar 3-37 Rancangan Antarmuka Halaman Data Mata Kuliah

Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-37.

1. *Label* Data Mata Kuliah yang menandakan bahwa pengguna sedang berada di halaman Data Mata Kuliah.

2. *DataGridView* yang digunakan untuk menampilkan data mata kuliah.
3. Tombol Selesai untuk keluar dari Halaman Data Mata Kuliah dan kembali ke halaman Beranda.
4. *GroupBox* Input Data yang berisi
 - a. *TextBox* Kode yang digunakan untuk mengisi data kode mata kuliah.
 - b. *TextBox* Mata Kuliah yang digunakan untuk mengisi data nama mata kuliah.
 - c. *TextBox* SKS yang digunakan untuk mengisi data SKS mata kuliah.
 - d. Tombol Batal yang digunakan untuk membatalkan perintah Tambah Data.
 - e. Tombol Simpan yang digunakan untuk menyimpan data yang telah diisikan.
5. *GroupBox* Menu yang berisi
 - a. Tombol Tambah yang digunakan untuk menambah data mata kuliah.
 - b. Tombol Ubah yang digunakan untuk mengubah data mata kuliah.
 - c. Tombol Hapus yang digunakan untuk menghapus data mata kuliah.

3.2.4.9. Rancangan Antarmuka Halaman Jadwal Perkuliahan

Gambar 3-38 merupakan gambar rancangan antarmuka halaman Data Jadwal Perkuliahan. Pada halaman ini terdapat satu kotak *DataGridView* untuk melihat data Jadwal Perkuliahan. Dua *GroupBox* yaitu *GroupBox* Input Data dan *GroupBox* Menu. Pada *GroupBox* Input Data berisi lima *ComboBox* yaitu Dosen, Mata Kuliah, Kelas, Ruang, Hari, dan *DateTimePicker* Jam, serta ada tombol Batal dan Simpan. Sedangkan *GroupBox* menu berisi tiga tombol, yaitu Tambah, Ubah, dan Hapus. Serta ada tombol Keluar.

1 DATA JADWAL PERKULIAHAN 3 Selesai

2

4

Input Data

Dosen

Mata Kuliah

Kelas

Ruang

Hari

Jam -

5

Menu

Gambar 3-38 Rancangan Antarmuka Halaman Data Jadwal Perkuliahan

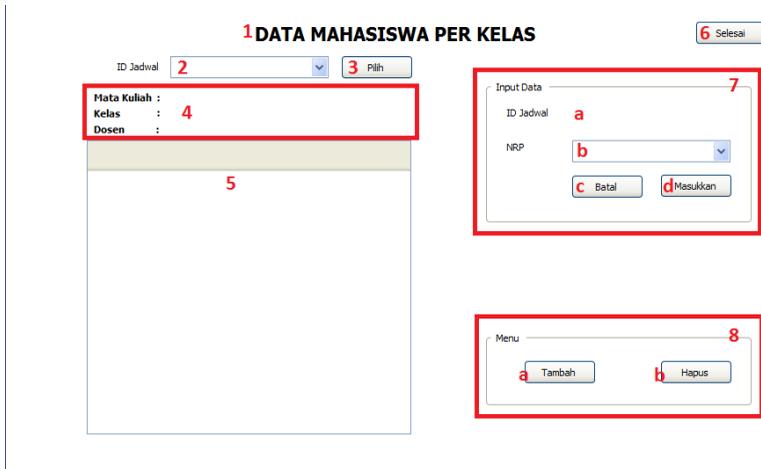
Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-38.

1. *Label* Data Jadwal Perkuliahan yang menandakan bahwa pengguna sedang berada di halaman Data Jadwal Perkuliahan
2. *DataGridView* yang digunakan untuk menampilkan data jadwal.
3. Tombol Selesai untuk keluar dari Halaman Data Jadwal Perkuliahan dan kembali ke halaman Beranda.
4. *GroupBox* Input Data yang berisi
 - a. *ComboBox* Dosen yang digunakan untuk mengisi data dosen pengajar pada suatu mata kuliah.
 - b. *ComboBox* Mata Kuliah yang digunakan untuk mengisi data Mata Kuliah.
 - c. *ComboBox* Kelas yang digunakan untuk mengisi data kelas dari mata kuliah.
 - d. *ComboBox* Ruang yang digunakan untuk mengisi data ruangan tempat mengajar pada suatu mata kuliah.
 - e. *ComboBox* Hari yang digunakan untuk mengisi data hari perkuliahan pada suatu mata kuliah.

- f. *DateTimePicker* Jam yang digunakan untuk mengisi data durasi perkuliahan pada suatu mata kuliah
 - g. Tombol Batal yang digunakan untuk membatalkan perintah Tambah Data.
 - h. Tombol Simpan yang digunakan untuk menyimpan data yang telah diisikan.
5. *GroupBox* Menu yang berisi
- a. Tombol Tambah yang digunakan untuk menambah data Jadwal per mata kuliah.
 - b. Tombol Ubah yang digunakan untuk mengubah data jadwal per mata kuliah.
 - c. Tombol Hapus yang digunakan untuk menghapus data jadwal per mata kuliah yang dipilih

3.2.4.10. Rancangan Antarmuka Halaman Kelas

Gambar 3-39 merupakan gambar rancangan antarmuka halaman Data Mahasiswa Per Kelas. Pada halaman ini terdapat *ComboBox* ID Jadwal untuk menampilkan data mahasiswa per kelas berdasarkan ID Jadwal. tombol Pilih untuk mengeksekusi dari ID Jadwal yang dipilih. Satu kotak *DataGridView* untuk melihat mahasiswa pada suatu kelas. Dua *GroupBox* yaitu *GroupBox* Input Data dan *GroupBox* Menu. Pada *GroupBox* Input Data berisi *Label* ID Jadwal dan *ComboBox* NRP. Sedangkan *GroupBox* menu berisi dua tombol, yaitu Tambah dan Hapus. Serta ada tombol Keluar.



Gambar 3-39 Rancangan Antarmuka Halaman Data Mahasiswa Per Kelas

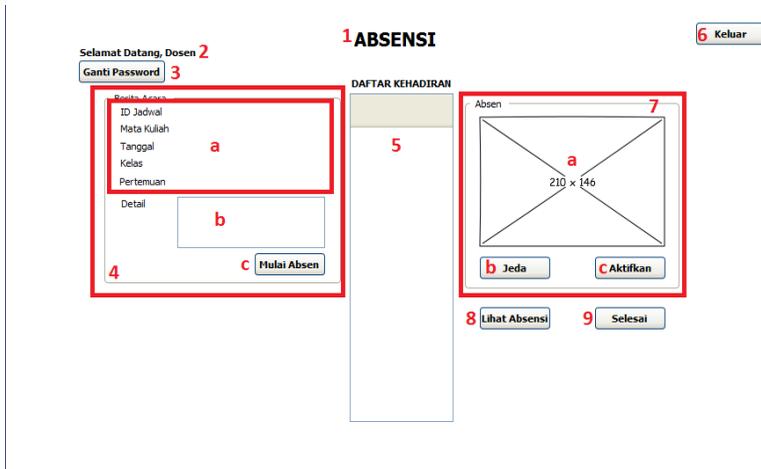
Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-39.

1. *Label* Data Mahasiswa Per Kelas yang menandakan bahwa pengguna sedang berada di halaman Data Mahasiswa Per Kelas
2. *ComboBox* ID Jadwal digunakan untuk menampilkan data mahasiswa per kelas berdasarkan ID Jadwal.
3. Tombol Pilih untuk mengeksekusi dari ID Jadwal yang dipilih
4. *Label* Mata Kuliah, Kelas, dan Dosen merupakan informasi dari ID Jadwal yang dipilih.
5. *DataGridView* yang digunakan untuk menampilkan data mahasiswa per kelas.
6. Tombol Selesai untuk keluar dari Halaman Tata Usaha dan kembali ke halaman Beranda.
7. *GroupBox* Input Data yang berisi
 - a. *Label* ID Jadwal yang menunjukkan data mahasiswa per kelas yang dipilih berdasarkan ID Jadwal

- b. *ComboBox* NRP yang digunakan untuk memilih mahasiswa yang akan dimasukkan ke dalam kelas berdasarkan NRP.
 - c. Tombol Masukkan yang digunakan untuk mengeksekusi dari *ComboBox* NRP yang telah dipilih
 - d. Tombol Batal yang digunakan untuk membatalkan penambahan data mahasiswa per kelas
8. *GroupBox* Menu yang berisi
- a. Tombol Tambah yang digunakan untuk menambah data mahasiswa per kelas.
 - b. Tombol Hapus yang digunakan untuk menghapus data mahasiswa per kelas

3.2.4.11. Rancangan Antarmuka Halaman Absensi

Gambar 3-40 merupakan gambar rancangan antarmuka halaman Absensi. Pada halaman ini terdapat satu kotak *DataGridView* untuk menampilkan data mahasiswa yang telah melakukan absensi. Dua *GroupBox* yaitu *GroupBox* Berita Acara dan *GroupBox* Absen. Pada *GroupBox* Berita Acara berisi ID Jadwal, Mata Kuliah, Tanggal, Kelas, Pertemuan dan Detail, serta ada tombol Mulai Absen. Sedangkan *GroupBox* Absen berisi *ImageBox* Absen dan tombol Aktifkan dan Jeda. Serta ada tombol Keluar, Lihat Absensi, dan Selesai.



Gambar 3-40 Rancangan Antarmuka Halaman Absensi

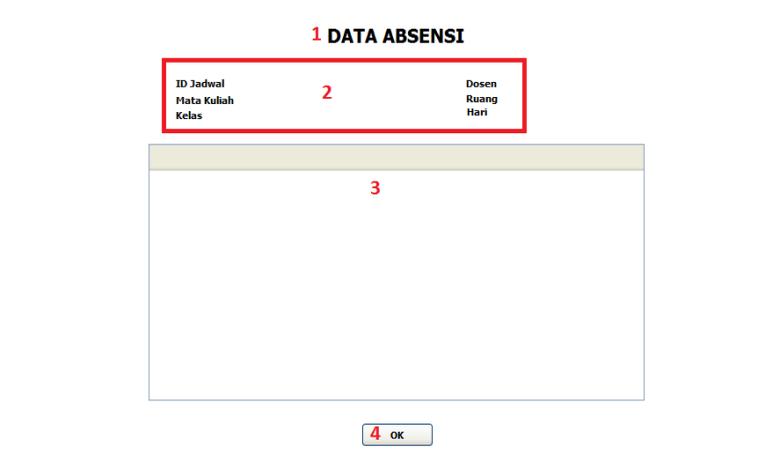
Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-40.

1. *Label* Absensi yang menandakan bahwa pengguna sedang berada di halaman Absensi.
2. *Label* Dosen merupakan nama akun yang sedang mengakses sistem.
3. Tombol Ganti Password yang digunakan untuk masuk ke *form* Ganti Password.
4. *GroupBox* Berita Acara yang berisi
 - a. *Label* ID Jadwal, Mata Kuliah, Tanggal, Kelas, dan Pertemuan merupakan informasi terkait mata kuliah pada saat dosen melakukan login.
 - b. *RichBox* Detail yang digunakan untuk mengisi data detail materi perkuliahan pada suatu pertemuan.
 - c. Tombol Mulai Absen yang digunakan untuk memulai proses absensi mahasiswa.
5. *DataGridView* Daftar Kehadiran yang digunakan untuk menampilkan data mahasiswa yang telah melakukan absensi.
6. Tombol Selesai yang digunakan untuk menyudahi proses absensi mahasiswa.

7. *GroupBox* Absen yang berisi
 - a. Tombol Aktifkan yang digunakan untuk mengaktifkan *webcam* pada *ImageBox* Absen.
 - b. Tombol Jeda untuk menjeda *webcam* pada *ImageBox* Absen
 - c. *ImageBox* Absen untuk mendeteksi dan mengenali wajah mahasiswa yang kemudian akan muncul *DialogBox* konfirmasi apakah benar yang dikenali sistem adalah wajah mahasiswa yang bersangkutan.
8. Tombol Lihat Absensi yang digunakan untuk melihat data absensi pada per kuliahhan tersebut.
9. Tombol Keluar untuk keluar dari Halaman Absensi dan kembali ke halaman Login

3.2.4.12. Rancangan Antarmuka Halaman Data Absensi

Gambar 3-41 merupakan gambar rancangan antarmuka halaman Data Absensi. Pada halaman ini terdapat kotak *DataGridView* untuk melihat data absensi mahasiswa. Serta ada tombol OK .



Gambar 3-41 Rancangan Antarmuka Halaman Data Absensi

Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3-41.

1. *Label* Data Absensi yang menandakan bahwa pengguna sedang berada di halaman Data Absensi
2. *Label* ID Jadwal, Mata Kuliah, Kelas, Dosen, Ruang, dan Hari digunakan untuk menampilkan detail data absensi kehadiran perkuliahan mahasiswa.
3. *DataGridView* Daftar Kehadiran yang digunakan untuk menampilkan data mahasiswa yang telah melakukan absensi.
4. Tombol OK yang digunakan untuk menutup Halaman Data Absensi.

BAB IV IMPLEMENTASI

Pada bab ini membahas tentang implementasi yang dilakukan berdasarkan rancangan sistem yang telah dijabarkan pada bab sebelumnya. Di dalamnya mencakup proses penerapan dan pengimplementasian proses dan antarmuka. Bahasa pemrograman yang digunakan adalah C# dan menggunakan *Library* Emgu CV. Sebelumnya akan ditunjukkan terlebih dahulu lingkungan untuk melakukan implementasi.

4.1. Lingkungan Implementasi

Dalam merancang aplikasi ini digunakan beberapa perangkat pendukung yang terdiri dari perangkat keras dan perangkat lunak.

4.1.1. Lingkungan Implementasi Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan aplikasi adalah komputer laptop. Spesifikasi dari perangkat tersebut adalah prosesor Inter(R) Core(TM) i3-3217U CPU @1.80GHz dan RAM 4.00 GB.

4.1.2. Lingkungan Implementasi Perangkat Lunak

Perangkat lunak yang digunakan dalam pengembangan aplikasi ini adalah sebagai berikut:

- Microsoft Windows 10 Home Single Language sebagai sistem operasi.
- Microsoft Visual Studio 2013 Profesional untuk implementasi aplikasi
- *Library* Emgu CV untuk implementasi metode *Eigenface*
- MySQL Server sebagai server basis data
- Power Designer 16.5 untuk merancang basis data
- Star UML 5.0 untuk desain aplikasi

- Evolus Pencil 2.0.5 untuk desain antarmuka

4.2. Implementasi Proses Aplikasi

Pada subbab ini akan membahas mengenai proses yang berjalan pada aplikasi. Proses yang terjadi karena ada permintaan dari pengguna aplikasi. Proses-proses yang diimplementasikan merupakan implementasi dari keseluruhan kasus penggunaan yang telah dibahas pada bab sebelumnya.

4.2.1. Implementasi Proses Menambah Data *User* TU

Berikut ini merupakan implementasi dari proses menambah data *user* tata usaha. Proses ini dilakukan oleh administrator. *User* tata usaha merupakan pengguna yang dapat mengakses sistem untuk mengelola data yang dibutuhkan dalam proses absensi perkuliahan mahasiswa selain *user* administrator.

4.2.2. Implementasi Proses Mengubah Data *User* TU

Berikut ini merupakan implementasi dari proses mengubah data *user* tata usaha. Proses ini dilakukan oleh administrator untuk mengubah data *user* tata usaha. Data *user* tata usaha yang hanya dapat diubah adalah nama *user* tata usaha, sedangkan untuk mengubah data *password* adalah hak pemilik akun *user* tata usaha.

4.2.3. Implementasi Proses Menghapus Data *User* TU

Berikut ini merupakan implementasi dari proses menghapus *user* tata usaha. Proses ini dilakukan oleh administrator untuk menghapus data *user* tata usaha.

4.2.4. Implementasi Proses Menambah Data Mata Kuliah

Berikut ini merupakan implementasi dari proses menambah data Mata Kuliah. Proses ini bisa dilakukan oleh administrator dan

tata usaha. Data mata kuliah diambil dari data sistem informasi integra ITS jurusan Teknik Informatika ITS kurikulum 2014 yang berisi kode mata kuliah, nama mata kuliah, dan SKS.

4.2.5. Implementasi Proses Mengubah Data Mata Kuliah

Berikut ini merupakan implementasi dari proses mengubah data mata kuliah. Proses ini bisa dilakukan oleh administrator dan tata usaha.

4.2.6. Implementasi Proses Menghapus Data Mata Kuliah

Berikut ini merupakan impementasi dari proses menghapus data mata kuliah. Proses ini bisa dilakukan oleh administrator dan tata usaha.

4.2.7. Implementasi Proses Menambah Data Mahasiswa

Berikut ini merupakan implementasi dari proses menambah data mahasiswa. Proses ini bisa dilakukan oleh administrator dan tata usaha. Data mahasiswa yang dimasukkan adalah data *dummy* yang berupa NRP dan nama mahasiswa.

4.2.8. Implementasi Proses Menambah Foto Mahasiswa

Berikut ini merupakan implementasi dari proses menambah foto mahasiswa. Proses ini bisa dilakukan oleh administrator dan tata usaha.

4.2.9. Implementasi Proses Mengubah Data Mahasiswa

Berikut ini merupakan implementasi dari proses mengubah data mahasiswa. Proses ini bisa dilakukan oleh administrator dan tata usaha. Data mahasiswa yang bisa diubah hanya nama mahasiswa saja.

4.2.10. Implementasi Proses Menghapus Data Mahasiswa

Berikut ini merupakan implementasi dari proses menghapus data mahasiswa. Proses ini bisa dilakukan oleh administrator dan tata usaha.

4.2.11. Implementasi Proses Melihat Data Mahasiswa

Berikut ini merupakan implementasi dari proses melihat data mahasiswa. Proses ini bisa dilakukan oleh administrator dan tata usaha. Data mahasiswa yang bisa dilihat ada Nama, NRP, dan foto mahasiswa yang telah ditambahkan sebelumnya.

4.2.12. Implementasi Proses Menambah Data Dosen

Berikut ini merupakan implementasi dari proses menambah data dosen. Proses ini bisa dilakukan oleh administrator dan tata usaha. Data yang ditambahkan adalah kode dosen, nama dosen, dan *password*.

4.2.13. Implementasi Proses Mengubah Data Dosen

Berikut ini merupakan implementasi dari proses mengubah data dosen. Proses ini bisa dilakukan oleh administrator dan tata usaha. Data dosen yang bisa diubah hanya nama dosen, sedangkan mengubah *password* adalah hak pemilik akun dosen.

4.2.14. Implementasi Proses Menghapus Data Dosen

Berikut ini merupakan implementasi dari proses menghapus data dosen. Proses ini bisa dilakukan oleh administrator dan tata usaha.

4.2.15. Implementasi Proses Menambah Data Jadwal

Berikut ini merupakan implementasi dari proses menambah data jadwal. Proses ini bisa dilakukan oleh administrator dan tata usaha. Data jadwal yang dimasukkan adalah jadwal perkuliahan semester genap jurusan Teknik Informatika ITS tahun 2005/2016.

4.2.16. Implementasi Proses Mengubah Data Jadwal

Berikut ini merupakan implementasi dari proses mengubah data jadwal. Proses ini bisa dilakukan oleh administrator dan tata usaha.

4.2.17. Implementasi Proses Menghapus Data Jadwal

Berikut ini merupakan implementasi dari proses menghapus data dosen. Proses ini bisa dilakukan oleh administrator dan tata usaha.

4.2.18. Implementasi Proses Menambah Data Mahasiswa Per Kelas

Berikut ini merupakan implementasi dari proses menambah data mahasiswa per kelas. Proses ini bisa dilakukan oleh administrator dan tata usaha. Data yang dimasukkan adalah NRP mahasiswa yang diambil dari data mahasiswa dan ID Jadwal yang diambil dari data jadwal.

4.2.19. Implementasi Proses Menghapus Data Mahasiswa Per Kelas

Berikut ini merupakan implementasi dari proses menghapus data mahasiswa per kelas. Proses ini bisa dilakukan oleh administrator dan tata usaha.

4.2.20. Implementasi Proses Mengubah *Password*

Berikut ini merupakan implementasi dari proses mengubah *password*. Proses ini bisa dilakukan oleh administrator, tata usaha, dan dosen.

4.2.21. Implementasi Proses Melihat Absensi

Berikut ini merupakan implementasi dari proses melihat absensi. Proses ini bisa dilakukan oleh semua pengguna, yaitu administrator, tata usaha, dosen, dan mahasiswa. Data absensi yang bisa dilihat yaitu pertemuan yang telah dilakukan sebelumnya, NRP mahasiswa yang hadir pada pertemuan tersebut, dan materi perkuliahannya.

4.2.22. Implementasi Proses Mengekstrak Data Absensi

Berikut merupakan implementasi dari proses mengekstrak data absensi. Proses ini bisa dilakukan oleh administrator dan tata usaha. Data yang diekstrak adalah data yang ditampilkan pada halaman Beranda administrator atau tata usaha.

4.2.23. Implementasi Proses Mengisi Berita Acara

Berikut ini merupakan implementasi dari proses mengisi berita acara. Proses ini dilakukan oleh dosen. Berita acara perkuliahan yang diisi adalah pertemuan dan materi perkuliahan.

4.2.24. Implementasi Proses Melakukan Absensi

Berikut ini merupakan implementasi dari proses melakukan absensi. Proses ini dilakukan oleh mahasiswa. Proses dilakukan dengan pengenalan wajah menggunakan metode *Eigenface*.

4.3. Implementasi Metode *Eigenface*

Pada subbab ini akan membahas mengenai implementasi dari metode *Eigenface* yang digunakan untuk mendeteksi dan mengenali wajah mahasiswa pada saat melakukan absensi perkuliahan.

4.3.1. Implementasi *Camera Capture*

Berikut ini merupakan implementasi *camera capture*. Yang dimaksud dengan *camera capture* di sini adalah mengaktifkan *webcam* pada komputer dan menampilkan gambar yang ditangkap oleh *webcam* pada *ImageBox* dari *library* Emgu CV. *Camera capture* pada aplikasi ini digunakan pada saat proses menambah foto mahasiswa dan proses absensi. *Camera capture* diimplementasikan dengan beberapa langkah, yaitu:

1. Membuka visual studio dan membuat proyek baru, seperti pada Gambar B-1.
2. Memilih *Windows Form Application*, seperti pada Gambar B-2.
3. Menambahkan beberapa *references* Emgu CV (*Emgu.CV.dll*, *Emgu.CV.UI.dll*, dan *Emgu.Util.dll*) pada proyek, seperti pada Gambar B-3.
4. Menambahkan *tools ImageBox* Emgu CV pada proyek dan atur Border Style menjadi Fixed Single, seperti pada Gambar B-4 dan Gambar B-5.

4.3.2. Implementasi Deteksi Wajah

Berikut ini merupakan implementasi deteksi wajah. Deteksi wajah pada aplikasi ini digunakan pada proses menambah foto mahasiswa dan proses absensi. Deteksi wajah diimplementasikan dengan beberapa langkah, yaitu:

1. Deklarasi *classifier*.

Mendeklarasikan objek kelas *HaarCascade* dapat dilihat pada Kode Sumber 4-1.

```
private HaarCascade haar;
```

Kode Sumber 4-1 Deklarasi HaarCascade

2. Memuat *file XML* dari *HaarCascade*.

Open CV telah berisi banyak pengklasifikasian untuk objek wajah, mata, senyum, dan lain-lain. Semua itu terdapat pada *file XML* dari Open CV. Sistem yang dibangun hanya perlu untuk mendeteksi wajah. Untuk memuat *file XML* dari *HaarCascade* untuk pendeteksian wajah dapat dilihat pada Kode Sumber 4-1.

```
haar = new  
HaarCascade("haarcascade_frontalface_default.xml");
```

Kode Sumber 4-2 Memuat File XML HaarCascade

3. Memasukkan kode untuk pendeteksian wajah

Pada *HaarCascade* yang digunakan untuk pendeteksian wajah pada sistem yang dibangun menggunakan *scale factor* 1.4, *min neighbors* 4, dan ukuran wajah 25 yang dapat dilihat pada Kode Sumber 4-3.

```
MCvAvgComp[]  
var faces = grayframe.DetectHaarCascade(haar, 1.4,  
4, HAAR_DETECTION_TYPE.DO_CANNY_PRUNING, new Size(25,  
25))[0];
```

Kode Sumber 4-3 Deteksi Wajah

4.3.3. Implementasi Pengenalan Wajah

Berikut ini merupakan implementasi pengenalan wajah. Pengenalan wajah pada aplikasi ini digunakan pada proses absensi menggunakan metode *Eigenface*. Pengenalan wajah diimplementasikan dengan beberapa langkah, yaitu

1. Memasukkan kriteria untuk pengenalan wajah.
Variabel *McvTermCriteria* adalah kriteria terminasi yang digunakan untuk training *eigen recognizer*. Kriteria yang digunakan pada sistem dapat dilihat pada Kode Sumber 4-4.

```
McvTermCriteria termCrit = new
McvTermCriteria(ContTrain, 0.001);
```

Kode Sumber 4-4 Kriteria Pengenalan Wajah

2. Menambah kode *Eigenface Recognizer*
Variabel *EigenObjectRecognizer* digunakan untuk sistem mengenali wajah pada saat melakukan absensi. Kode *Eigenface Recognizer* dapat dilihat pada Kode Sumber 4-5.

```
EigenObjectRecognizer recognizer = new
EigenObjectRecognizer (trainingImages.ToArray(),
labels.ToArray(), thrs, ref termCrit);
```

Kode Sumber 4-5 Eigenface Recognizer

3. Menampilkan nama dari wajah yang terdeteksi dan dikenali oleh sistem
Jika wajah dikenali sistem, maka aplikasi akan menampilkan NRP mahasiswa yang sedang melakukan absensi. Kode yang digunakan untuk menampilkan nama yang dikenali dapat dilihat pada Kode Sumber 4-6.

```
currentFrame.Draw(name, ref font, new Point(f.rect.X
- 2, f.rect.Y - 2), new Bgr(Color.LightGreen)); if
(name != "")
```

Kode Sumber 4-6 Menampilkan Nama yang Dikenali

4.4. Implementasi Basis Data

Pada subbab ini membahas implementasi dari perancangan basis data yang telah dijelaskan pada subbab sebelumnya. Basis yang digunakan adalah sistem manajemen basis data relasional MySQL yang digunakan untuk menyimpan data-data yang dibutuhkan oleh sistem. Implementasi basis data ini mengacu pada model data fisik yang telah dijelaskan pada subbab sebelumnya.

4.4.1. Implementasi Tabel foto

Tabel ini digunakan untuk menyimpan seluruh atribut data foto mahasiswa. Detail implementasi pembuatan tabel foto dapat dilihat pada Kode Sumber 4-7.

```
create table FOTO
(
  IDFOTO          int not null,
  NRP             varchar(10),
  NAMA FOTO      varchar(100),
  primary key (IDFOTO)
);

alter table FOTO add constraint FK_RELATIONSHIP_1 foreign
key (NRP)
references MAHASISWA (NRP) on delete restrict on
update restrict;
```

Kode Sumber 4-7 Implementasi Tabel foto

4.4.2. Implementasi Tabel mahasiswa

Tabel ini digunakan untuk menyimpan seluruh atribut data mahasiswa. Detail implementasi pembuatan tabel mahasiswa dapat dilihat pada Kode Sumber 4-8.

```
create table MAHASISWA
(
  NRP             varchar(10) not null,
  NAMA           varchar(100),
```

```
primary key (NRP)
);
```

Kode Sumber 4-8 Implementasi Tabel mahasiswa

4.4.3. Implementasi Tabel kehadiran

Tabel ini digunakan untuk menyimpan seluruh atribut data absensi perkuliahan mahasiswa. Detail implementasi pembuatan tabel kehadiran dapat dilihat pada Kode Sumber 4-9.

```
create table KEHADIRAN
(
  IDDETAIL          varchar(100) not null,
  IDBERITA          varchar(100),
  NRP               varchar(10),
  KETERANGAN       varchar(50),
  primary key (IDDETAIL)
);

alter table KEHADIRAN add constraint FK_RELATIONSHIP_2
foreign key (NRP)
references MAHASISWA (NRP) on delete restrict on
update restrict;

alter table KEHADIRAN add constraint FK_RELATIONSHIP_8
foreign key (IDBERITA)
references BERITAACARA (IDBERITA) on delete restrict
on update restrict;
```

Kode Sumber 4-9 Implementasi Tabel kehadiran

4.4.4. Implementasi Tabel beritaAcara

Tabel ini digunakan untuk menyimpan seluruh atribut data berita acara perkuliahan. Detail implementasi pembuatan tabel beritaAcara dapat dilihat pada Kode Sumber 4-10.

```
create table BERITAACARA
(
  IDBERITA          varchar(100) not null,
  IDJADWAL          varchar(100),
```

```

PERTEMUANKE      varchar(5),
TANGGAL          varchar(50),
DETAIL           varchar(500),
primary key (IDBERITA)
);

alter table BERITAACARA add constraint FK_RELATIONSHIP_5
foreign key (IDJADWAL)
references JADWAL (IDJADWAL) on delete restrict on
update restrict;

```

Kode Sumber 4-10 Implementasi Tabel beritaAcara

4.4.5. Implementasi Tabel jadwal

Tabel ini digunakan untuk menyimpan seluruh atribut data jadwal perkuliahan. Detail implementasi pembuatan tabel jadwal dapat dilihat pada Kode Sumber 4-11.

```

create table JADWAL
(
  IDJADWAL      varchar(100) not null,
  KODEMK       varchar(10),
  KODEDOSEN    varchar(5),
  KELAS        varchar(10),
  RUANG        varchar(10),
  HARI         varchar(20),
  JAMMULAI     int,
  JAMSELESAI  int,
  primary key (IDJADWAL)
);

alter table JADWAL add constraint FK_RELATIONSHIP_12
foreign key (KODEDOSEN)
references DOSEN (KODEDOSEN) on delete restrict on
update restrict;

alter table JADWAL add constraint FK_RELATIONSHIP_9
foreign key (KODEMK)
references MATAKULIAH (KODEMK) on delete restrict
on update restrict;

```

Kode Sumber 4-11 Implementasi Tabel jadwal

4.4.6. Implementasi Tabel kelas

Tabel ini digunakan untuk menyimpan seluruh atribut data mahasiswa per kelas. Detail implementasi pembuatan tabel kelas dapat dilihat pada Kode Sumber 4-12.

```
create table KELAS
(
  IDKELAS          varchar(100) not null,
  NRP              varchar(10),
  IDJADWAL        varchar(100),
  primary key (IDKELAS)
);

alter table KELAS add constraint FK_RELATIONSHIP_10
foreign key (NRP)
references MAHASISWA (NRP) on delete restrict on
update restrict;

alter table KELAS add constraint FK_RELATIONSHIP_11
foreign key (IDJADWAL)
references JADWAL (IDJADWAL) on delete restrict on
update restrict;
```

Kode Sumber 4-12 Implementasi Tabel kelas

4.4.7. Implementasi Tabel mataKuliah

Tabel ini digunakan untuk menyimpan seluruh atribut data mata kuliah. Detail implementasi pembuatan tabel mataKuliah dapat dilihat pada Kode Sumber 4-13.

```
create table MATAKULIAH
(
  KODEMK          varchar(10) not null,
  MATAKULIAH     varchar(100),
  SKS             int,
  primary key (KODEMK)
);
```

Kode Sumber 4-13 Implementasi Tabel mataKuliah

4.4.8. Implementasi Tabel dosen

Tabel ini digunakan untuk menyimpan seluruh atribut data dosen. Detail implementasi pembuatan tabel dosen dapat dilihat pada Kode Sumber 4-14.

```
create table DOSEN
(
  KODEDOSEN          varchar(5) not null,
  NAMA                varchar(100),
  PASSWORD            varchar(100),
  primary key (KODEDOSEN)
);
```

Kode Sumber 4-14 Implementasi Tabel dosen

4.4.9. Implementasi Tabel login

Tabel ini digunakan untuk menyimpan seluruh atribut data administrator. Detail implementasi pembuatan tabel login dapat dilihat pada Kode Sumber 4-15.

```
create table LOGIN
(
  IDLOGIN            int not null,
  NAMA                varchar(100),
  USERNAME           varchar(50),
  PASSWORD            varchar(100),
  KATEGORI            varchar(50),
  primary key (IDLOGIN)
);
```

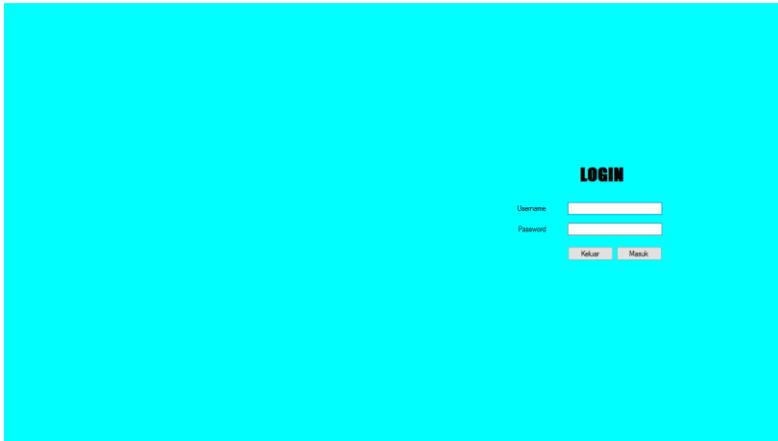
Kode Sumber 4-15 Implementasi Tabel login

4.5. Implementasi Antarmuka Pengguna

Pada subbab ini membahas implementasi antarmuka pengguna yang berdasarkan rancangan antarmuka yang telah dibahas pada bab sebelumnya. Implementasi antarmuka pengguna dibangun dengan aplikasi berbasis *desktop* dan menggunakan bahasa pemrograman C#.

4.5.1. Implementasi Antarmuka Halaman Login

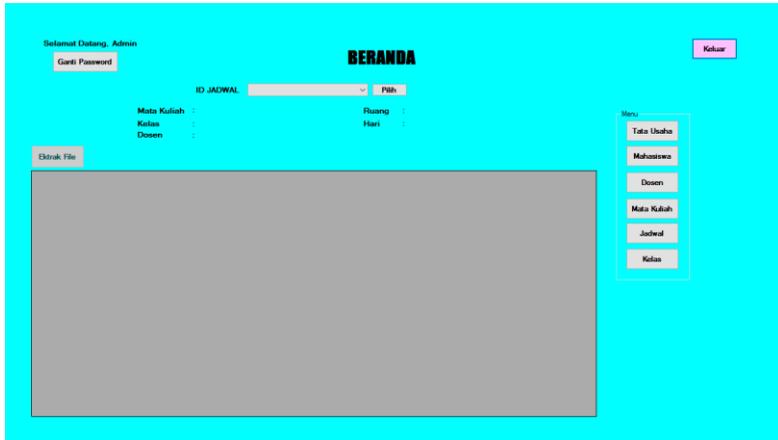
Halaman ini merupakan implementasi antarmuka halaman Login. Terdapat *TextBox* yang berisi Username dan Password untuk pengguna masuk ke dalam sistem. Implementasi antarmuka halaman Login dapat dilihat pada gambar 4-1.



Gambar 4-1 Implementasi Antarmuka Halaman Login

4.5.2. Implementasi Antarmuka Halaman Beranda untuk Administrator

Halaman ini merupakan implementasi antarmuka halaman Beranda untuk administrator. Halaman ini ditampilkan setelah administrator masuk dari halaman Login. Terdapat enam menu yang bisa dipilih, yaitu Tata Usaha untuk masuk ke halaman Data Tata Usaha, Mahasiswa untuk masuk ke halaman Data Mahasiswa, Dosen untuk masuk ke halaman Data Dosen, Mata Kuliah untuk masuk ke halaman Data Mata Kuliah, Jadwal untuk masuk ke halaman Data Jadwal Perkuliahan, dan Kelas untuk masuk ke halaman Data Mahasiswa Per Kelas. Implementasi halaman Beranda untuk administrator dapat dilihat pada Gambar 4-2.

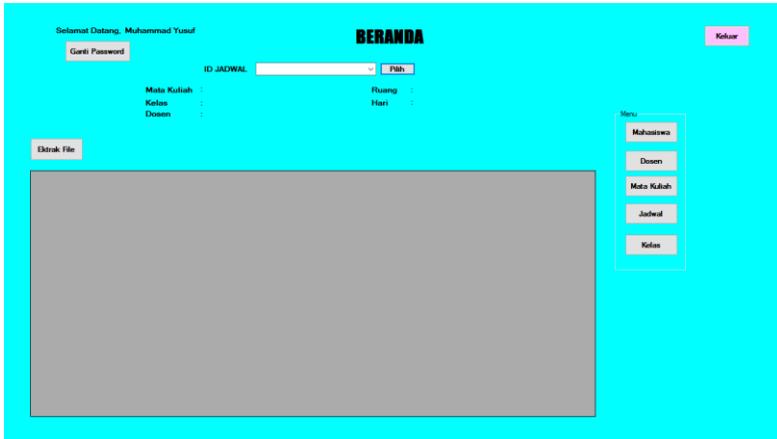


Gambar 4-2 Implementasi Antarmuka Halaman Beranda untuk Administrator

Pada halaman ini administrator bisa melihat absensi perkuliahan mahasiswa. Sebelumnya administrator harus memilih kelas yang mau dilihat data absensinya berdasarkan ID Jadwal pada *ComboBox* ID Jadwal. Kemudian administrator menekan tombol Pilih dan sistem akan menampilkan data absensi perkuliahan mahasiswa.

4.5.3. Implementasi Antarmuka Halaman Beranda untuk Tata Usaha

Halaman ini merupakan implementasi antarmuka halaman Beranda untuk tata usaha. Halaman ini ditampilkan setelah tata usaha masuk dari halaman Login. Terdapat lima menu yang dapat dipilih, yaitu Mahasiswa untuk masuk ke halaman Data Mahasiswa, Dosen untuk masuk ke halaman Data Dosen, Mata Kuliah untuk masuk ke halaman Data Mata Kuliah, Jadwal untuk masuk ke halaman Data Jadwal Perkuliahan, dan Kelas untuk masuk ke halaman Data Mahasiswa Per Kelas. Implementasi halaman Beranda untuk tata usaha dapat dilihat pada Gambar 4-3.

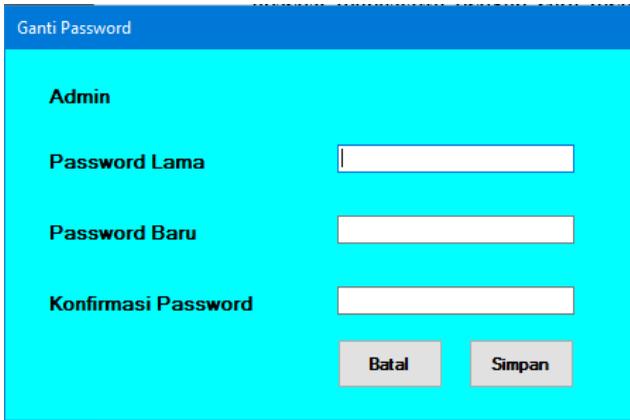


Gambar 4-3 Implementasi Antarmuka Halaman Beranda untuk Tata Usaha

Pada halaman ini tata usaha juga bisa melihat absensi perkuliahan mahasiswa per pertemuan. Sebelumnya tata usaha harus memilih kelas yang mau dilihat data absensinya berdasarkan ID Jadwal pada *ComboBox* ID Jadwal. Kemudian tata usaha menekan tombol Pilih dan sistem akan menampilkan data absensi perkuliahan mahasiswa.

4.5.4. Implementasi Antarmuka Ganti Password

Halaman ini merupakan implementasi antarmuka *Form* Ganti Password untuk administrator, tata usaha, dan dosen. Halaman ini ditampilkan setelah pengguna menekan tombol Ganti Password. Terdapat tiga kolom yang harus diisi oleh pengguna untuk mengubah *password*, kemudian pengguna dapat menekan tombol Simpan untuk menyimpan perubahan *password*. Implementasi halaman *Form* Ganti Password dapat dilihat pada Gambar 4-4.



Ganti Password

Admin

Password Lama

Password Baru

Konfirmasi Password

Batal **Simpan**

Gambar 4-4 Implementasi Antarmuka Ganti Password

4.5.5. Implementasi Antarmuka Halaman Data Tata Usaha

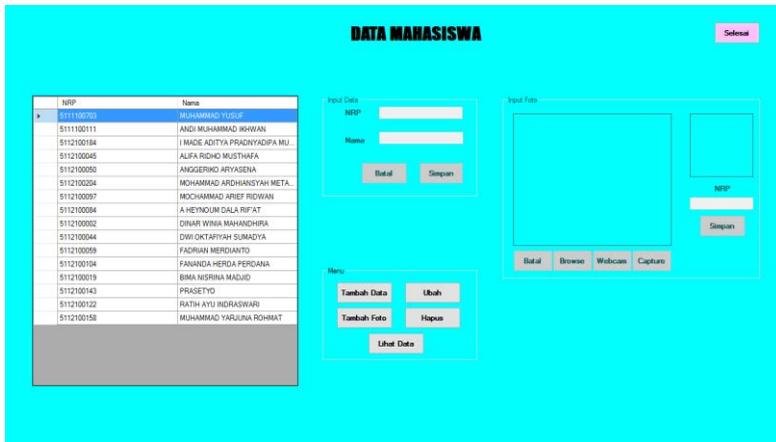
Halaman ini merupakan implementasi antarmuka halaman Data Tata Usaha. Halaman ini hanya bisa diakses oleh administrator. Halaman ini ditampilkan setelah administrator menekan tombol Tata Usaha pada halaman Beranda untuk administrator. Terdapat tiga menu yang dapat dipilih, yaitu Tambah, Ubah, dan Hapus. Pada halaman ini juga dapat menampilkan data tata usaha yang telah ditambahkan oleh administrator. Implementasi halaman Beranda untuk tata usaha dapat dilihat pada Gambar 4-5.



Gambar 4-5 Implementasi Antarmuka Halaman Data Tata Usaha

4.5.6. Implementasi Antarmuka Halaman Data Mahasiswa

Halaman ini merupakan implementasi antarmuka halaman Data Mahasiswa. Halaman ini bisa diakses oleh administrator dan tata usaha. Halaman ini ditampilkan setelah pengguna menekan tombol Mahasiswa pada halaman Beranda. Terdapat lima menu yang dapat dipilih, yaitu Tambah Data, Tambah Foto, Lihat Data, Ubah, dan Hapus. Pada halaman ini juga menampilkan data mahasiswa yang telah ditambahkan oleh pengguna. Implementasi halaman Data Mahasiswa dapat dilihat pada Gambar 4-6.



Gambar 4-6 Implementasi Antarmuka Halaman Data Mahasiswa

4.5.7. Implementasi Antarmuka Form Lihat Data Mahasiswa

Halaman ini merupakan implementasi antarmuka *form* Lihat Data Mahasiswa. Halaman ini ditampilkan setelah administrator atau tata usaha menekan tombol Lihat Data pada halaman Data Mahasiswa. Untuk melihat koleksi foto, pengguna dapat menekan tombol *next* dan *previous* pada *form*. Pengguna juga dapat menghapus foto yang sedang ditampilkan dengan menekan tombol Hapus. Implementasi halaman *Form* Lihat Data Mahasiswa dapat dilihat pada Gambar 4-7.



Gambar 4-7 Implementasi Antarmuka Form Lihat Data Mahasiswa

4.5.8. Implementasi Antarmuka Halaman Data Mata Kuliah

Halaman ini merupakan implementasi antarmuka halaman Data Mata Kuliah. Halaman ini bisa diakses oleh administrator dan tata usaha. Halaman ini ditampilkan setelah pengguna menekan tombol Mata Kuliah pada halaman Beranda. Terdapat tiga menu yang dapat dipilih, yaitu Tambah, Ubah, dan Hapus. Pada halaman ini juga dapat menampilkan data mata kuliah yang telah ditambahkan oleh pengguna. Implementasi halaman Data Mata Kuliah dapat dilihat pada Gambar 4-8.



Gambar 4-8 Implementasi Antarmuka Halaman Data Mata Kuliah

4.5.9. Implementasi Antarmuka Halaman Data Jadwal Perkuliahan

Halaman ini merupakan implementasi antarmuka halaman Data Jadwal Perkuliahan. Halaman ini bisa diakses oleh administrator dan tata usaha. Halaman ini ditampilkan setelah pengguna menekan tombol Jadwal pada halaman Beranda. Terdapat tiga menu yang dapat dipilih, yaitu Tambah, Ubah, dan Hapus. Pada halaman ini juga dapat menampilkan data jadwal perkuliahan yang telah ditambahkan oleh pengguna. Implementasi halaman Data Jadwal Perkuliahan dapat dilihat pada Gambar 4-9.

DATA JADWAL PERKULIAHAN

[Selesai](#)

ID Jadwal	Dosen	Mata Kuliah	Kelas	Ruang	Hari	Jam Mulai	Jam Selesai
KI141447/	Orison, P.	Tekn. Sist. A	Lah. Alpa	Riku	16:30	20:00	
KI141306/	Prof. Y. Sup.	Organisasi ... A	IF-101	Serim	07:30	10:00	
KI141329/	Fajar Basik.	Manajemen ... A	IF-101	Serim	10:30	13:00	
KI141314/	Ir. Mucham.	Jaringan Ko. A	IF-101	Serim	13:30	16:00	
KI141306/	Ir. Mucham.	Organisasi ... B	IF-102	Serim	07:30	10:00	
KI141415/	Baskoro Ad.	Teknologi B. A	IF-102	Serim	10:30	13:00	
KI141314/	Tahar Alim.	Jaringan Ko. B	IF-102	Serim	13:30	16:00	
KI141306/	Ir. FK. Anu.	Organisasi ... C	IF-103	Serim	07:30	10:00	
KI141329/	Sarwoto, S.	Manajemen ... C	IF-103	Serim	10:30	13:00	
KI141314/	Baskoro Ad.	Jaringan Ko. C	IF-103	Serim	13:30	16:00	
KI141306/	Hennig Te.	Organisasi ... D	IF-104	Serim	07:30	10:00	
KI141329/	Adhatus Sa.	Manajemen ... D	IF-104	Selasa	10:30	13:00	
KI141314/	Rasyana M.	Jaringan Ko. D	IF-104	Serim	13:30	16:00	
KI141329/	Umi Lak Yc.	Manajemen ... B	IF-105A	Serim	07:30	10:00	
KI141329/	Umi Lak Yc.	Manajemen ... E	IF-105A	Serim	10:30	13:00	
KI141314/	Rialtyo An.	Jaringan Ko. E	IF-105A	Serim	13:30	16:00	
KI141329/	Fajar Basik.	Manajemen ... F	IF-105B	Serim	07:30	10:00	
KI141314/	Rialtyo An.	Jaringan Ko. F	IF-105B	Serim	10:30	13:00	
KI141317/	Victor Hena.	Disusika A	IF-106	Serim	07:30	10:00	

Formulir Data:

Dosen:

Mata Kuliah:

Kelas:

Ruang:

Hari:

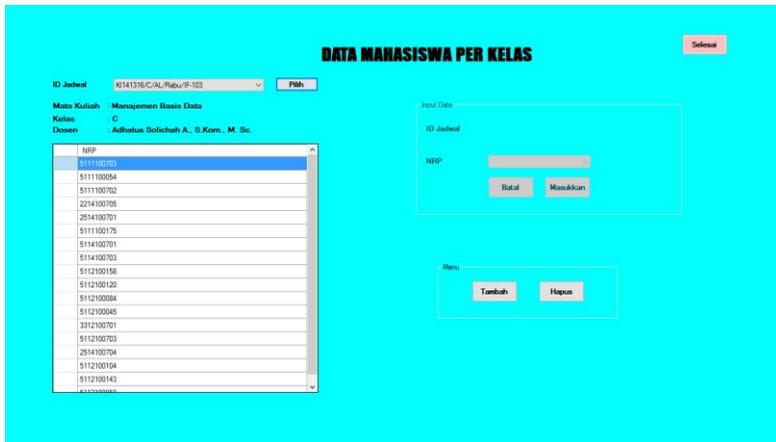
Jam: -

Menu:

Gambar 4-9 Implementasi Antarmuka Halaman Data Jadwal Perkuliahan

4.5.10. Implementasi Antarmuka Halaman Data Mahasiswa Per Kelas

Halaman ini merupakan implementasi antarmuka halaman Data Mahasiswa Per Kelas. Halaman ini bisa diakses oleh administrator dan tata usaha. Halaman ini ditampilkan setelah pengguna menekan tombol Kelas pada halaman Beranda. Terdapat tiga menu yang dapat dipilih, yaitu Tambah, Ubah, dan Hapus. Pada halaman ini juga dapat menampilkan data mahasiswa per kelas yang telah ditambahkan oleh pengguna. Implementasi halaman Data Mahasiswa Per Kelas dapat dilihat pada Gambar 4-10.



Gambar 4-10 Implementasi Antarmuka Halaman Data Mahasiswa Per Kelas

4.5.11. Implementasi Antarmuka Halaman Absensi

Halaman ini merupakan implementasi antarmuka halaman Absensi. Halaman ini ditampilkan setelah dosen masuk dari halaman Login. Halaman ini terdapat empat fungsi, yaitu mengubah *password*, mengisi berita acara, melakukan absensi, dan melihat data absensi kehadiran mahasiswa. Fungsi mengubah *password* adalah fungsi yang digunakan oleh dosen yang sedang mengakses sistem. Kemudian fungsi mengisi berita acara adalah kasus penggunaan yang juga dilakukan oleh dosen. Setelah mengisi berita acara dengan menekan tombol Mulai Absen, fungsi absensi mahasiswa baru bisa diaktifkan. Kemudian setelah proses absensi mahasiswa selesai dilakukan, fungsi melihat data absensi bisa dilakukan. Data absensi berupa pertemuan akan ditampilkan pada *DataGridView* pertemuan, data absensi berupa daftar mahasiswa yang hadir dalam suatu pertemuan akan ditampilkan pada *DataGridView* Daftar Kehadiran, dan materi perkuliahan pada suatu pertemuan akan ditampilkan dalam *RichBox* Detail Materi Perkuliahan. Implementasi halaman Absensi dapat dilihat pada Gambar 4-11.

Gambar 4-11 Implementasi Antarmuka Halaman Absensi

4.5.12. Impementasi Antarmuka Halaman Data Absensi

Halaman ini merupakan implementasi antarmuka Halaman Data Absensi. Halaman ini ditampilkan setelah dosen atau mahasiswa menekan tombol Lihat Absensi pada halaman Absensi. Implementasi halaman *Form* Lihat Data Mahasiswa dapat dilihat pada Gambar 4-12.

DATA ABSENSI

ID Jodoh: K141316/CAL/Isabu/IF-103 Dosen: Athatus Solichah A., S.Kom., M. Sc.
Mata Kuliah: Manajemen Basis Data Ruang: IF-103
Kelas: C Hari: Sabtu

NRP	NAMA	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
5111100703	MUHAMMAD YUSUF	hadir															
5111100094	Saina Katana L.	hadir	hadir	X	hadir												
5111100702	M. Rizal Ar	hadir	hadir	hadir	hadir	hadir	hadir	X	X	X	X	X	X	X	X	X	X
2514100705	Muhammad Fauz	hadir	hadir	hadir	hadir	hadir	hadir	X	X	X	X	X	X	X	X	X	X
2514100701	Muhammad Rifq Rusydians	hadir	hadir	hadir	hadir	X	X	X	X	X	X	X	X	X	hadir	X	X
5111100175	Byan Athath	hadir	X	X	X												
5114100701	Ar Raad Stone	hadir	hadir	X	X	X	X	X	X	X	X	X	hadir	X	X	X	X
5114100703	Ahmad Muazaka Ishaq	hadir	X	X	X	X	X	X	X	X	X	hadir	X	X	X	X	X
5112100195	MUHAMMAD YARLUNA RICHMAT	hadir	X														
5112100120	Muhammad Saifurrah	hadir	hadir	hadir	hadir	hadir	hadir	X	X	X	X	X	X	X	X	X	hadir
5112100044	A HEYINDUM DALA RIFAT	hadir	X	X													
5112100045	ALIFA RIDHO MUSTHFA	hadir	X	X	X	X	X										
5112100701	Moh. Khoirul Rohman	hadir	X	X	X	X	X	X									
5112100703	Muhammad Husen	hadir	X	X	X	X	X	X									
2514100704	Ahmad Athallah	hadir	X	X	X	X	X	X	X	X							
5112100104	FANANCA HERDA PERDANA	hadir	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100105	Adnan Nurrahman	hadir	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

OK

Gambar 4-12 Implementasi Antarmuka Halaman Data Absensi

BAB V

PENGUJIAN DAN EVALUASI

Bab ini membahas pengujian dan evaluasi pada aplikasi yang dikembangkan. Proses pengujian dilakukan terhadap kebutuhan fungsionalitas sistem. Hasil evaluasi menjabarkan tentang rangkuman hasil pengujian pada bagian akhir bab ini.

5.1. Lingkungan Pengujian

Lingkungan pengujian sistem pada pengerjaan tugas akhir ini dilakukan pada lingkungan dan alat kakas.

Prosesor	: Intel(R) Core(TM) i3-3217U CPU 1.80GHz
Memori	: 4.00 GB
Sistem Operasi	: Windows 10 Home Single Language 64-bit
Ukuran layar	: 1280 x 720 pixels.

5.2. Skenario Pengujian

Pada bagian ini akan membahas tentang data yang dilakukan pada saat pengujian dan skenario pengujian yang dilakukan. Pengujian yang dilakukan adalah pengujian kebutuhan fungsionalitas. Pengujian fungsionalitas menggunakan metode kotak hitam (*black box*). Metode ini menekankan pada kesesuaian hasil keluaran sistem.

5.2.1. Data Pengujian

Pada subbab ini akan membahas data yang digunakan pada saat pengujian. Data-data yang digunakan pada saat pengujian diantaranya yaitu:

1. Data administrator dan tata usaha merupakan data *dummy* yang dapat dilihat pada Tabel 5-1.

Tabel 5-1 Data Uji Administrator dan Tata Usaha

No.	Nama	Username	Kategori
1.	Admin	admin	Admin
2.	Muhammad Yusuf	yusuf	Tata usaha

2. Data dosen merupakan data dosen pengajar aktif di jurusan Teknik Informatika ITS yang dapat dilihat pada Tabel 5-2.

Tabel 5-2 Data Uji Dosen

No.	Nama	Kode Dosen
1.	Dr.tech. Ir. R. V. Hari Ginardi, M. Sc.	HG
2.	Adhatus Solichah A., S.Kom., M. Sc.	AL

3. Data mata kuliah merupakan data mata perkuliahan di jurusan Teknik Informatika ITS kurikulum 2014 yang dapat dilihat pada Tabel 5-3.

Tabel 5-3 Data Uji Mata Kuliah

No.	Kode MK	Mata Kuliah	SKS
1.	KI141447	Topik Khusus Manajemen Informasi	3
2.	KI141316	Manajemen Basis Data	3

4. Data jadwal merupakan data jadwal perkuliahan di jurusan Teknik Informatika pada semester genap 2015/2016. Data jadwal yang digunakan dalam pengujian dapat dilihat pada Tabel 5-4.
5. Data mahasiswa merupakan data mahasiswa aktif ITS dari berbagai jurusan.
6. Data foto merupakan foto yang diinput berdasarkan dari data mahasiswa yang telah diinput sebelumnya. Masing-masing mahasiswa diambil tiga foto sebagai data uji. Data mahasiswa dan *dataset* foto wajah yang digunakan dalam pengujian dapat dilihat pada Tabel 5-5.

Tabel 5-4 Data Uji Jadwal Perkuliahan

ID Jadwal	Kode MK	Kode Dosen	Kelas	Ruang	Hari	Pukul	
						Mulai	Selesai
KI141447/A/HG/Selasa/Lab. Alpro	KI141447	HG	A	Lab. Alpro	Rabu	16:00	18:00
KI141316/C/AL/Rabu/IF-103	KI141316	AL	C	IF-103	Minggu	07:30	10:00

Tabel 5-5 Data Uji Foto Wajah Mahasiswa

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
1.	5111100703	Muhammad Yusuf			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
2.	5111100054	Satria Katana L.			
3.	5111100702	M. Rizal Ar.			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
4.	2214100705	Muhammad Faiz			
5.	2514100701	Muhammad Rifqi Rusydani			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
6.	5111100175	Bryan Alfadhori			
7.	5114100701	Ar Raad Stone			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
8.	5114100703	Ahmad Muzakki Ishaqi			
9.	5112100158	Muhammad Yarjuna Rohmat			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
10.	5112100120	Mochammad Bakhtiar			
11.	5112100084	A Heynoum Dala Rif'at			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
12.	5112100045	Alifa Ridho Musthafa			
13.	3312100701	Moh. Khoiror Rohman			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
14.	5112100702	Mujahid Husein			
15.	2514100704	Ahmad Athoillah			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
16.	5112100104	Fananda Herda Perdana			
17.	5112100143	Prasetyo			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
18.	5112100050	Anggeriko Aryasena			
19.	1411100703	Jazirotul Rohmah			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
20.	1311100702	Ulul Azmi			
21.	1411100702	Anik Fathimatuz Zahro			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
22.	2511100703	Marsida			
23.	2211100701	Nani Lathifatun Nada			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
24.	1511100703	Din Dzakamala Fafi Rohmatillah			
25.	1511100701	Liziyatin Nasihah			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
26.	3111100702	Urifatul Azizah			
27.	5211100702	Endang Sulistiyani			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
28.	1511100702	Dian Fitriani			
29.	2411100703	Khusnul Khotimah			

No	NRP	Nama	Wajah 1	Wajah 2	Wajah 3
30.	1211100702	Mariatul Ilmiah			

5.2.2. Pengujian Fungsionalitas

Pada subbab ini akan membahas pengujian fungsionalitas pada sistem untuk menambah, mengubah, menghapus, dan melihat data yang dibutuhkan pada aplikasi yang dibangun. Pengguna pada aplikasi ini adalah administrator, tata usaha, dosen, dan mahasiswa.

5.2.2.1. Pengujian Menambah Data Mahasiswa

Berikut ini membahas tentang pengujian menambah data mahasiswa. Pada proses menambah data mahasiswa, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Mahasiswa. Skenario pengujian menambah data mahasiswa dapat dilihat pada Tabel 5-6.

Tabel 5-6 Skenario Pengujian Menambah Data Mahasiswa

Kode Pengujian	PUJ.UC-0001
Nama	Pengujian Menambah Data Mahasiswa
Referensi Kasus Penggunaan	UC-0007
Tujuan	Memeriksa fungsi untuk menambah data mahasiswa berfungsi atau tidak
Kondisi Awal	Data mahasiswa belum ada
Skenario	<ol style="list-style-type: none">1. Administrator atau tata usaha melakukan <i>login</i>2. Administrator atau tata usaha masuk ke halaman Data Mahasiswa3. Administrator atau tata usaha menekan tombol Tambah Data4. Administrator atau tata usaha mengisi data mahasiswa5. Administrator atau tata usaha menekan tombol Simpan
Masukan	Data mahasiswa baru

Keluaran yang Diharapkan	Data mahasiswa tersimpan ke dalam basis data
Hasil Pengujian	Data mahasiswa berhasil ditambahkan ke dalam basis data. Pengujian berhasil.

Data yang perlu diisi oleh administrator atau tata usaha seperti pada Gambar 5-1.

Input Data

NRP

Nama

Batal **Simpan**

Gambar 5-1 Pengisian Data Mahasiswa

Selanjutnya administrator atau tata usaha menekan tombol Simpan dan sistem akan menyimpan data mahasiswa yang baru kemudian menampilkannya pada *DataGridView*. Hasil data mahasiswa yang telah berhasil ditambahkan dapat dilihat pada Gambar 5-2.

	NRP	Nama
▶	5111100703	Muhammad Yusuf

Gambar 5-2 Hasil Penambahan Data Mahasiswa

5.2.2.2. Pengujian Menambah Foto Mahasiswa

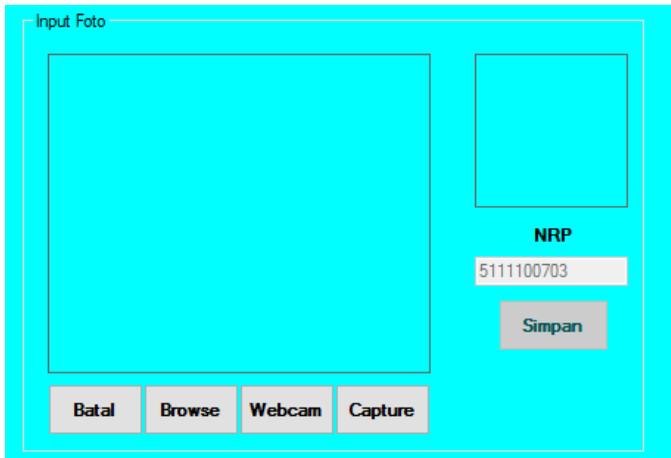
Berikut ini membahas tentang pengujian menambah foto mahasiswa. Pada proses menambah mahasiswa, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Mahasiswa. Skenario pengujian menambah foto mahasiswa dapat dilihat pada Tabel 5-7.

Tabel 5-7 Skenario Pengujian Menambah Foto Mahasiswa

Kode Pengujian	PUJ.UC-0002
Nama	Pengujian Menambah Foto Mahasiswa
Referensi Kasus Penggunaan	UC-0008
Tujuan	Memeriksa fungsi untuk menambah foto mahasiswa berfungsi atau tidak
Kondisi Awal	Data foto mahasiswa belum ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Mahasiswa 3. Administrator atau tata usaha memilih mahasiswa yang ingin ditambah foto 4. Administrator atau tata usaha menekan tombol Tambah Foto 5. Administrator atau tata usaha memilih cara untuk menambah foto, <i>browse</i> atau <i>webcam</i> 6. Administrator atau tata usaha menekan tombol Capture 7. Administrator atau tata usaha menekan tombol Simpan
Masukan	Foto baru mahasiswa
Keluaran yang Diharapkan	Foto mahasiswa tersimpan ke dalam basis data

Hasil Pengujian	Foto baru mahasiswa berhasil ditambahkan ke dalam basis data. Pengujian berhasil.
-----------------	--

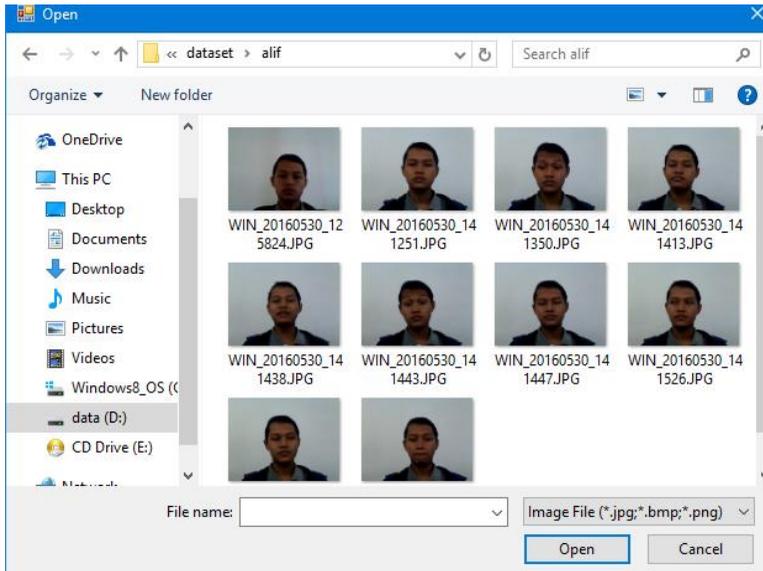
Data mahasiswa yang dipilih administrator atau tata usaha untuk ditambah foto baru berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Tambah Foto. Selanjutnya administrator atau tata usaha memilih metode tambah foto, *browse* atau *webcam*. *Form Input* Foto dapat dilihat pada Gambar 5-3.



The image shows a software interface titled "Input Foto". It features a large empty rectangular box on the left for a photo. To its right is a smaller empty rectangular box. Below the smaller box is a label "NRP" and a text input field containing the number "5111100703". Below the input field is a "Simpan" button. At the bottom of the form are four buttons: "Batal", "Browse", "Webcam", and "Capture".

Gambar 5-3 *Form Input* Foto Mahasiswa

Kemudian ketika memilih metode *browse*, maka sistem akan membuka *file explorer* seperti pada Gambar 5-4.



Gambar 5-4 Input Foto Metode Browse

Sedangkan ketika memilih metode *webcam*, maka sistem akan mengaktifkan *webcam* pada komputer seperti pada Gambar 5-5.



Gambar 5-5 *Input Foto Metode Webcam*

Kemudian administrator atau tata usaha menekan tombol Capture dan sistem akan menampilkan foto hasil *capture* pada *ImageBox* seperti pada Gambar 5-6.



Gambar 5-6 *Hasil Capture Wajah*

Selanjutnya administrator atau tata usaha menekan tombol Simpan. Sistem akan menyimpan foto baru mahasiswa. Hasil foto baru mahasiswa yang telah berhasil ditambah dapat dilihat pada pengujian lihat data mahasiswa.

5.2.2.3. Pengujian Mengisi Berita Acara

Berikut ini membahas tentang pengujian mengisi berita acara. Pada proses mengisi berita acara, dosen diharuskan telah masuk ke dalam sistem dan berada pada halaman Absensi. Skenario pengujian mengisi berita acara dapat dilihat pada Tabel 5-8.

Tabel 5-8 Skenario Pengujian Mengisi Berita Acara

Kode Pengujian	PUJ.UC-0003
Nama	Pengujian Mengisi Berita Acara
Referensi Kasus Penggunaan	UC-0023
Tujuan	Memeriksa fungsi untuk mengisi berita acara berfungsi atau tidak
Kondisi Awal	Data berita acara belum ada
Skenario	<ol style="list-style-type: none"> 1. Dosen melakukan <i>login</i> 2. Dosen berada pada halaman Absensi 3. Dosen mengisi data berita acara 4. Dosen menekan tombol Mulai Absen
Masukan	Data berita acara baru
Keluaran yang Diharapkan	Data berita acara tersimpan ke dalam basis data
Hasil Pengujian	Data berita acara berhasil ditambahkan ke dalam basis data. Pengujian berhasil.

Data yang perlu diisi oleh dosen seperti pada Gambar 5-7.

Selamat Datang. Dr.tech. Ir. R. V. Hari Ginardi, M. Sc.

Ganti Password

Berita Acara

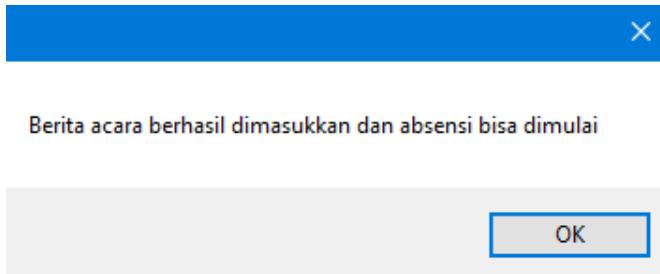
ID Jadwal : KI141447/A/HG/Selasa/Lab. Alpro
Mata Kuliah : Topik Khusus Manajemen Informasi
Kelas : A
Tanggal : 31/05/2016
Pertemuan : 16

Detail :

Mulai Absen

Gambar 5-7 Pengisian Berita Acara

Selanjutnya dosen menekan tombol Mulai Absen dan sistem akan menyimpan data berita acara yang baru kemudian fitur untuk melakukan absensi perkuliahan mahasiswa diaktifkan sehingga absensi bisa dimulai seperti pada Gambar 5-8.



Gambar 5-8 Keterangan Berita Acara Berhasil Disimpan dan Absensi Bisa Dimulai

5.2.2.4. Pengujian Melakukan Absensi

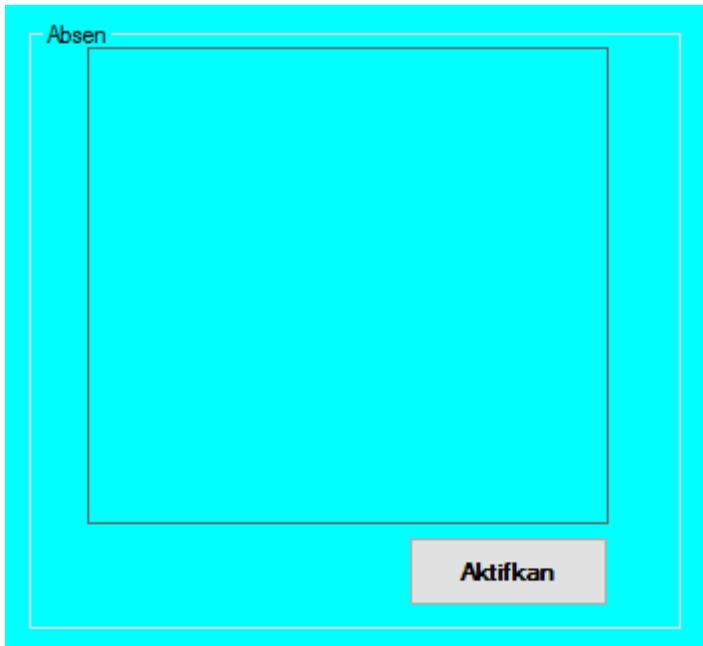
Berikut ini membahas tentang pengujian melakukan absensi. Pada proses melakukan absensi, mahasiswa diharuskan telah berada pada halaman Absensi. Skenario pengujian melakukan absensi dapat dilihat pada Tabel 5-9.

Tabel 5-9 Skenario Pengujian Melakukan Absensi

Kode Pengujian	PUJ.UC-0004
Nama	Pengujian Melakukan Absensi
Referensi Kasus Penggunaan	UC-0024
Tujuan	Memeriksa fungsi untuk melakukan absensi berfungsi atau tidak
Kondisi Awal	Data-data yang dibutuhkan sudah ada di sistem
Skenario	<ol style="list-style-type: none"> 1. Mahasiswa menekan tombol Aktifkan 2. Mahasiswa mengkonfirmasi apakah benar mahasiswa tersebut adalah mahasiswa yang dikenali oleh sistem
Masukan	Data absensi baru
Keluaran yang Diharapkan	Data absensi tersimpan ke dalam basis data

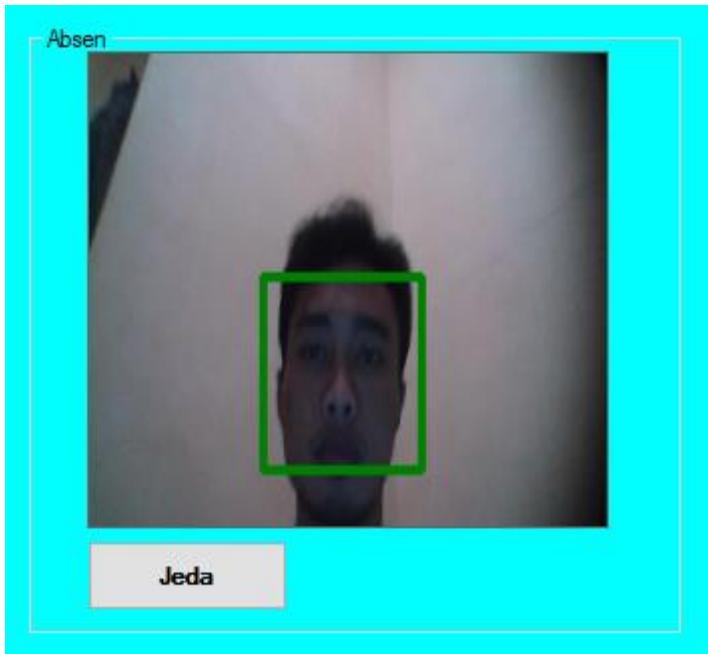
Hasil Pengujian	Data absensi mahasiswa berhasil ditambahkan ke dalam basis data untuk kategori wajah yang memiliki pencahayaan dan resolusi yang sama pada saat melakukan absensi seperti dijelaskan pada bab-bab sebelumnya. Pengujian berhasil dengan dengan persentase 93,33%.
-----------------	--

Fitur absensi mahasiswa dapat dilihat seperti pada Gambar 5-9.



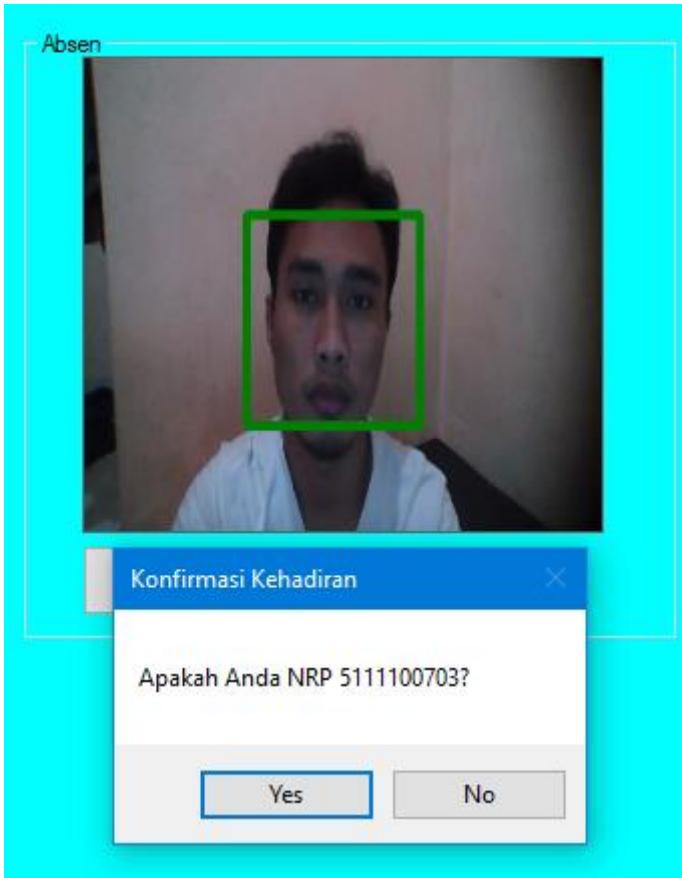
Gambar 5-9 Fitur Absensi Mahasiswa

Mahasiswa menekan tombol Aktifkan dan sistem akan mengaktifkan fitur *webcam* komputer seperti pada Gambar 5-10.



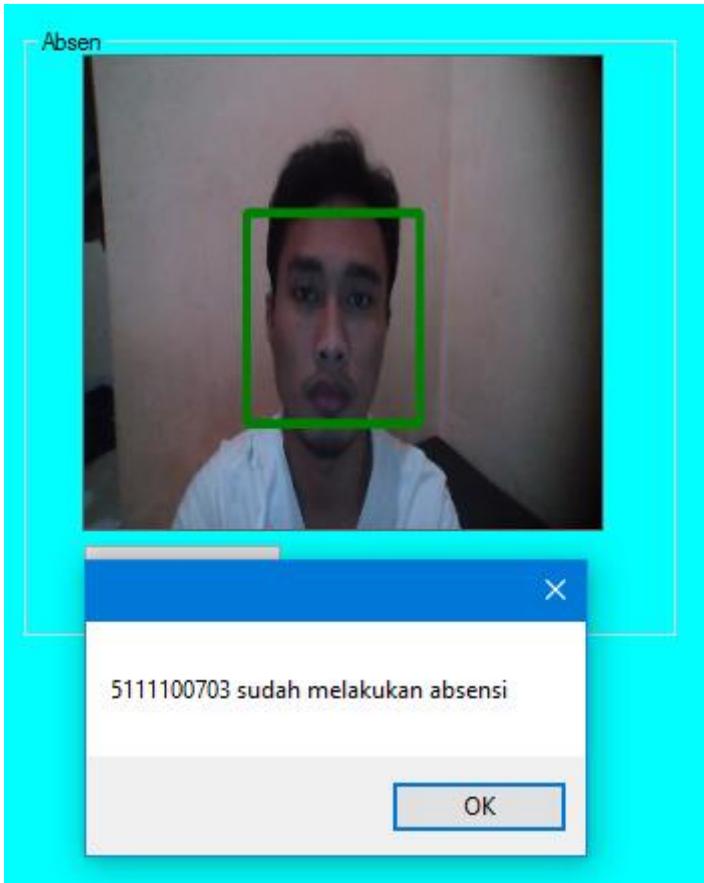
Gambar 5-10 Fitur Absensi Mahasiswa Diaktifkan

Kemudian sistem akan memberikan konfirmasi apakah benar mahasiswa yang sedang melakukan absensi adalah mahasiswa yang dikenali sistem seperti pada Gambar 5-11.



Gambar 5-11 Konfirmasi Kebenaran Identitas Mahasiswa

Setelah mahasiswa memberikan konfirmasi benar, maka sistem akan menambahkan data absensi pada kelas dan pertemuan tersebut dan menampilkan keterangan bahwa mahasiswa telah melakukan absensi seperti pada Gambar 5-12.



Gambar 5-12 Mahasiswa Telah Melakukan Absensi

Pengujian melakukan absensi ini dilakukan kepada mahasiswa dengan *dataset* wajahnya telah di-*input* sebelumnya. Adapun hasil yang diperoleh dapat dilihat pada Tabel 5-10 berikut.

Tabel 5-10 Hasil Pengujian Melakukan Absensi

No	NRP	Nama	Hasil
1.	5111100703	Muhammad Yusuf	Dikenali

No	NRP	Nama	Hasil
2.	5111100054	Satria Katana L.	Dikenali
3.	5111100702	M. Rizal Ar.	Dikenali
4.	2214100705	Muhammad Faiz	Dikenali
5.	2514100701	Muhammad Rifqi Rusydani	Dikenali
6.	5111100175	Bryan Alfadhori	Dikenali
7.	5114100701	Ar Raad Stone	Dikenali
8.	5114100703	Ahmad Muzakki Ishaqi	Dikenali
9.	5112100158	Muhammad Yarjuna Rohmat	Dikenali
10.	5112100120	Mochammad Bakhtiar	Dikenali
11.	5112100084	A Heynoum Dala Rif'at	Dikenali
12.	5112100045	Alifa Ridho Musthafa	Dikenali
13.	3312100701	Moh. Khoiror Rohman	Dikenali
14.	5112100702	Mujahid Husein	Dikenali
15.	2514100704	Ahmad Athoillah	Dikenali
16.	5112100104	Fananda Herda Perdana	Dikenali
17.	5112100143	Prasetyo	Tidak Dikenali
18.	5112100050	Anggeriko Aryasena	Tidak Dikenali
19.	1411100703	Jaziroatul Rohmah	Dikenali
20.	1311100702	Ulul Azmi	Dikenali
21.	1411100702	Anik Fathimatuz Zahro	Dikenali
22.	2511100703	Marsida	Dikenali
23.	2211100701	Nani Lathifatun Nada	Dikenali
24.	1511100703	Din Dzakamala Fafi Rohmatillah	Dikenali
25.	1511100701	Liziyatin Nasihah	Dikenali
26.	3111100702	Urifatul Azizah	Dikenali
27.	5211100702	Endang Sulistiyani	Dikenali
28.	1511100702	Dian Fitriani	Dikenali
29.	2411100703	Khusnul Khotimah	Dikenali
30.	1211100702	Mariatul Ilmiah	Dikenali

Foto yang dikenali sistem merupakan foto yang diambil dari webcam laptop dan foto yang tidak dikenali sistem merupakan foto

yang diambil dari kamera ponsel mahasiswa dengan resolusi dan pencahayaan yang tinggi.

5.2.2.5. Pengujian Melihat Absensi

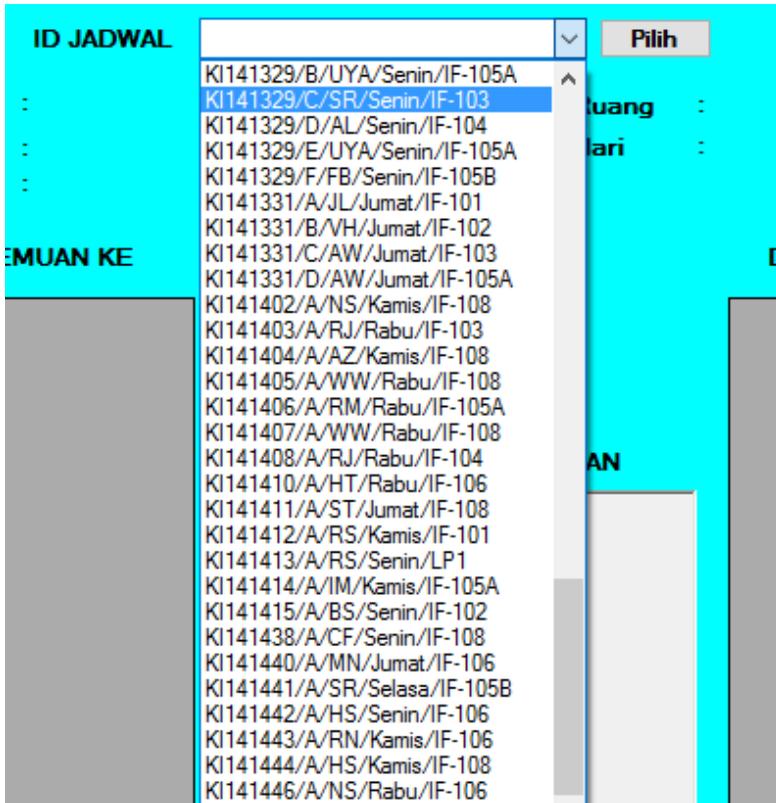
Berikut ini membahas tentang pengujian melihat absensi. Pada proses melihat absensi, pengguna diharuskan telah berada pada halaman Beranda masing-masing. Skenario pengujian melihat absensi dapat dilihat pada Tabel 5-11.

Tabel 5-11 Skenario Pengujian Melihat Absensi

Kode Pengujian	PUJ.UC-0005
Nama	Pengujian Melihat Absensi
Referensi Kasus Penggunaan	UC-0021
Tujuan	Memeriksa fungsi untuk melihat absensi berfungsi atau tidak
Kondisi Awal	Data absensi sudah ada
Skenario	<ol style="list-style-type: none"> 1. Pengguna telah berada pada halaman Beranda masing-masing 2. Untuk administrator atau tata usaha memilih kelas yang ingin dilihat data absensinya berdasarkan ID Jadwal terlebih dahulu 3. Administrator atau tata usaha menekan tombol Pilih 4. Pengguna memilih pertemuan yang ingin dilihat daftar hadir mahasiswa 5. Pengguna menekan tombol Lihat Kehadiran
Masukan	-
Keluaran yang Diharapkan	Data pertemuan dan daftar hadir mahasiswa ditampilkan oleh sistem
Hasil Pengujian	Data pertemuan dan daftar hadir mahasiswa ditampilkan oleh sistem.

	Pengujian berhasil.
--	---------------------

Data kelas berdasarkan ID Jadwal yang bisa dipilih oleh administrator atau tata usaha seperti pada Gambar 5-13.



Gambar 5-13 Daftar Data Kelas Berdasarkan ID Jadwal

Kemudian administrator atau tata usaha menekan tombol Pilih dan sistem akan menampilkan data absensi perkuliahan mahasiswa pada suatu kelas berdasarkan ID Jadwal yang dipilih seperti pada Gambar 5-14.

Selamat Datang, Admin

Ganti Password

BERANDA

Keluar

ID JADWAL:

Mata Kuliah : **Topik Khusus Manajemen Informasi** Ruang : **Lab. Alpro**
Kelas : **A** Hari : **Sabtu**
Dosen : **Dr.tech. Ir. R. V. Hari Ginardi, M. Sc.**

Ekstrak File

NRP	NAMA	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
5112100204	MOHAMMAD ARDHIANSYAH METANA PUTRA	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100184	I MADE ADITYA PRADNYADIPA MUSTIKA	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100158	MUHAMMAD YARJUNA ROHMAT	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100143	PRASETYO	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100122	RATIH AYU INDRASWARI	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100104	FANANDA HERDA PERDANA	hadir	h													
5112100097	MOHAMMAD ARIEF RIDWAN	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100094	A HEYNOUM DALA RIF'AT	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100059	FADRIAN MERDIANTO	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100050	ANGGERIKO ARYASENA	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100045	ALIFA RIDHO MUSTHAFA	hadir	h													
5112100044	DWI OKTAFIYAH SUMADYA	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100019	BIMA NISRINA MADJID	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5112100002	DINAR WINIA MAHANDHIRA	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5111100703	MUHAMMAD YUSUF	hadir	h													
5111100111	ANDI MUHAMMAD IKHWAN	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Menu

Tata Usaha

Mahasiswa

Dosen

Mata Kuliah

Jadwal

Kelas

Gambar 5-14 Data Absensi Mahasiswa

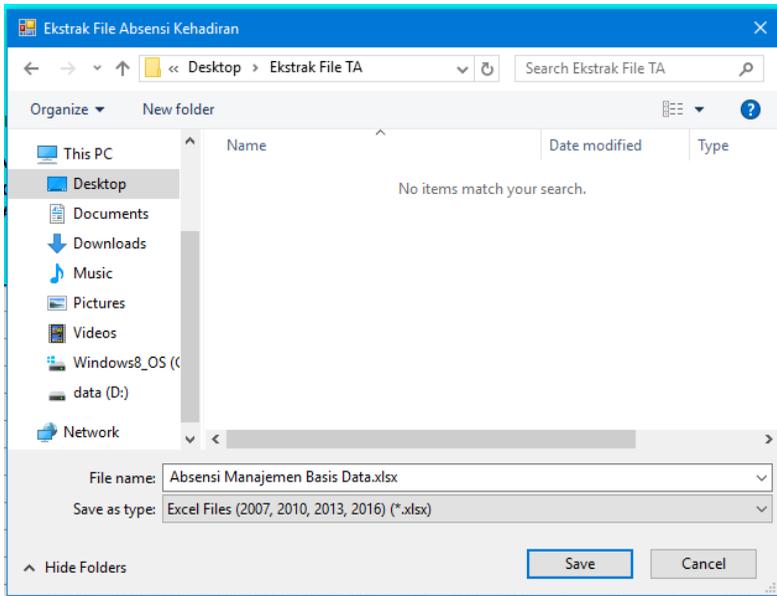
5.2.2.6. Pengujian Mengekstrak Data Absensi

Berikut ini membahas tentang pengujian mengekstrak data absensi. Pada proses mengekstrak data absensi, administrator atau tata usaha diharuskan telah masuk ke dalam sistem, berada pada halaman Beranda, dan telah melihat absensi. Skenario pengujian mengekstrak data absensi dapat dilihat pada Tabel 5-12.

Tabel 5-12 Skenario Pengujian Mengekstrak Data Absensi

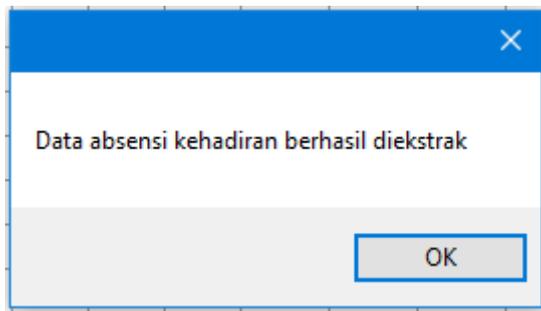
Kode Pengujian	PUJ.UC-0006
Nama	Pengujian Mengekstrak Data Absensi
Referensi Kasus Penggunaan	UC-0022
Tujuan	Memeriksa fungsi untuk mengekstrak data absensi berfungsi atau tidak
Kondisi Awal	Data sudah memenuhi 16 kali pertemuan
Skenario	<ol style="list-style-type: none">1. Administrator atau tata usaha menekan tombol Ekstrak File2. Administrator atau tata usaha memilih direktori File Explorer3. Administrator atau tata usaha menekan tombol Save
Masukan	Data berita acara baru
Keluaran yang Diharapkan	Data berita acara tersimpan ke dalam basis data
Hasil Pengujian	Data berita acara berhasil ditambahkan ke dalam basis data. Pengujian berhasil.

Administrator atau tata usaha menekan tombol Ekstrak File. Kemudian memilih direktori File Explorer untuk menyimpan *file* seperti pada Gambar 5-15.



Gambar 5-15 Memilih Direktori Penyimpanan File

Kemudian akan muncul pesan bahwa data absensi berhasil diekstrak seperti pada Gambar 5-16.



Gambar 5-16 Pesan Data Absensi Berhasil Diekstrak

Hasil ekstraksi file dapat dilihat pada Gambar 5-17.

Absensi Manajemen Basis Data.xlsx - Excel

File Home Insert Page Layout Formulas Data Review View Tell me what you want to do... Yusuf Alkathoni Share

A1 : Mata Kuliah

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Mata Kuliah	Manajemen Basis Data														
2	Kelas	C														
3	Dosen	Adhatus Solichah A., S.Kom., M. Sc.														
4	Ruang	IF-103														
5	Hari	Sabtu														
6																
7	NRP	NAMA	1	2	3	4	5	6	7	8	9	10	11	12	13	14
8	5111100703	MUHAMMAD YUSUF	hadir													
9	5111100054	Satria Katana L.	hadir	hadir	X	hadir										
10	5111100702	M. Rizal Ar.	hadir	hadir	hadir	hadir	hadir	hadir	X	X	X	X	X	X	X	X
11	2214100705	Muhammad Faiz	hadir	hadir	hadir	hadir	hadir	X	X	X	X	X	X	X	X	X
12	2514100701	Muhammad Rifqi Rusydani	hadir	hadir	hadir	X	X	X	X	X	X	X	X	X	hadir	X
13	5111100175	Bryan Alfadhori	hadir	X	X											
14	5114100701	Ar Raad Stone	hadir	hadir	X	X	X	X	X	X	X	X	hadir	X	X	X
15	5114100703	Ahmad Muzakka Ishaqi	hadir	X	X	X	X	X	X	X	X	hadir	X	X	X	X
16	5112100158	MUHAMMAD YARJUNA ROHMAT	hadir													
17	5112100120	Mochammad Bakhtiar	hadir	hadir	hadir	hadir	hadir	hadir	X	X	X	X	X	X	X	hadir
18	5112100084	A HEYNOUM DALA RIF'AT	hadir	X	X											
19	5112100045	ALIFA RIDHO MUSTHafa	hadir	X	X	X										
20	3312100701	Moh. Khoirur Rohman	hadir	X	X	X	X	X								
21	5112100703	Mujahid Husein	hadir	X	X	X	X	X								
22	2514100704	Ahmad Athoillah	hadir	X	X	X	X	X	X							
23	5112100104	FANANDA HERDA PERDANA	hadir	X	X	X	X	X	X	X	X	X	X	X	X	X
24	5112100143	PRASETYO	X	X	X	X	X	X	X	X	X	X	X	X	X	X
25	5112100050	ANGGERIKO ARYASENA	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Sheet1

Ready

Gambar 5-17 File Data Hasil Ekstraksi

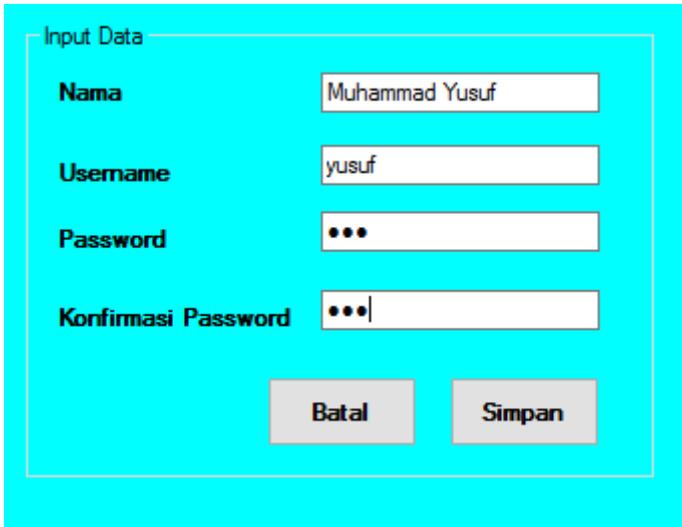
5.2.2.7. Pengujian Menambah Data *User* TU

Berikut ini membahas tentang pengujian menambah data *user* tata usaha. Pada proses menambah data *user* tata usaha, administrator diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Tata Usaha. Skenario pengujian menambah data *user* tata usaha dapat dilihat pada Tabel 5-13.

Tabel 5-13 Skenario Pengujian Menambah Data *User* TU

Kode Pengujian	PUJ.UC-0007
Nama	Pengujian Menambah Data <i>User</i> TU
Referensi Kasus Penggunaan	UC-0001
Tujuan	Memeriksa fungsi untuk menambah data <i>user</i> tata usaha berfungsi atau tidak
Kondisi Awal	Data <i>user</i> tata usaha belum ada
Skenario	<ol style="list-style-type: none">1. Administrator melakukan <i>login</i>2. Administrator masuk ke halaman Data Tata Usaha3. Administrator menekan tombol Tambah4. Administrator mengisi data <i>user</i> tata usaha5. Administrator menekan tombol Simpan
Masukan	Data <i>user</i> tata usaha baru
Keluaran yang Diharapkan	Data <i>user</i> tata usaha tersimpan ke dalam basis data
Hasil Pengujian	Data <i>user</i> tata usaha berhasil ditambahkan ke dalam basis data. Pengujian berhasil.

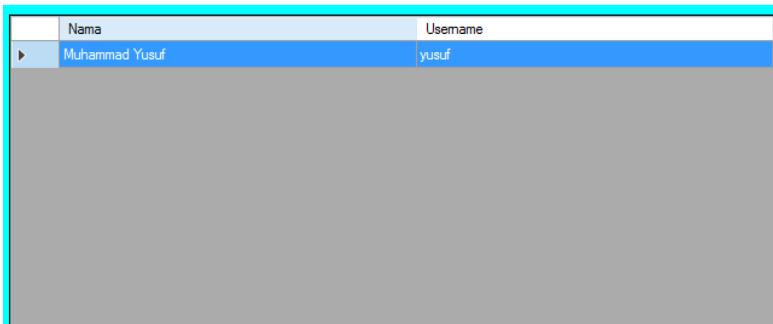
Data yang perlu diisi oleh administrator seperti pada Gambar 5-18.



The image shows a web form titled "Input Data" with a light blue background. It contains four input fields: "Nama" with the value "Muhammad Yusuf", "Username" with the value "yusuf", "Password" with three dots, and "Konfirmasi Password" with three dots. Below the fields are two buttons: "Batal" and "Simpan".

Gambar 5-18 Pengisian Data Tata Usaha

Selanjutnya administrator menekan tombol Simpan dan sistem akan menyimpan data *user* tata usaha yang baru kemudian menampilkannya pada *DataGridView*. Hasil data *user* tata usaha yang telah berhasil ditambahkan dapat dilihat pada Gambar 5-19.



	Nama	Username
▶	Muhammad Yusuf	yusuf

Gambar 5-19 Hasil Penambahan Data User Tata Usaha

5.2.2.8. Pengujian Mengubah Data *User* TU

Berikut ini membahas tentang pengujian mengubah data *user* tata usaha. Pada proses mengubah data *user* tata usaha, administrator diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Tata Usaha. Skenario pengujian mengubah data *user* tata usaha dapat dilihat pada Tabel 5-14.

Tabel 5-14 Skenario Pengujian Mengubah Data *User* TU

Kode Pengujian	PUJ.UC-0008
Nama	Pengujian Mengubah Data <i>User</i> TU
Referensi Kasus Penggunaan	UC-0002
Tujuan	Memeriksa fungsi untuk mengubah data <i>user</i> tata usaha berfungsi atau tidak
Kondisi Awal	Data <i>user</i> tata usaha sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator melakukan <i>login</i> 2. Administrator masuk ke halaman Data Tata Usaha 3. Administrator memilih <i>user</i> tata usaha yang ingin diubah 4. Administrator menekan tombol Ubah 5. Administrator mengisi data <i>user</i> tata usaha yang ingin diubah 6. Administrator menekan tombol Simpan
Masukan	Data <i>user</i> tata usaha baru
Keluaran yang Diharapkan	Data <i>user</i> tata usaha tersimpan ke dalam basis data
Hasil Pengujian	Data <i>user</i> tata usaha berhasil diubah dan diperbarui ke dalam basis data. Pengujian berhasil.

Data *user* tata usaha yang dipilih administrator berada pada *DataGridView*. Kemudian administrator menekan tombol Ubah.

Data yang perlu diisi untuk diubah oleh administrator seperti pada Gambar 5-20.



Input Data

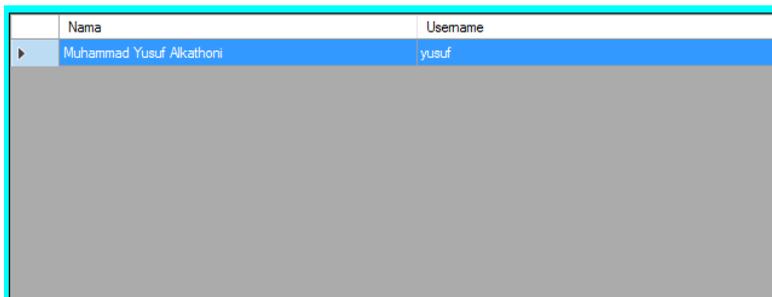
Nama Muhammad Yusuf Alkathoni

Username yusuf

Batal Simpan

Gambar 5-20 Perubahan Data *User* Tata Usaha

Selanjutnya administrator menekan tombol Simpan dan sistem akan menyimpan data *user* tata usaha yang baru kemudian menampilkannya pada *DataGridView*. Hasil data *user* tata usaha yang telah berhasil diubah dan diperbarui dapat dilihat pada Gambar 5-21.



	Nama	Username
▶	Muhammad Yusuf Alkathoni	yusuf

Gambar 5-21 Hasil Perubahan Data *User* Tata Usaha

5.2.2.9. Pengujian Menghapus Data *User* TU

Berikut ini membahas tentang pengujian menghapus data *user* tata usaha. Pada proses menghapus data *user* tata usaha, administrator diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Tata Usaha. Skenario pengujian menghapus data *user* tata usaha dapat dilihat pada Tabel 5-15.

Tabel 5-15 Skenario Pengujian Menghapus Data *User* Tata Usaha

Kode Pengujian	PUJ.UC-0009
Nama	Pengujian Menghapus Data <i>User</i> TU
Referensi Kasus Penggunaan	UC-0003
Tujuan	Memeriksa fungsi untuk menghapus data <i>user</i> tata usaha berfungsi atau tidak
Kondisi Awal	Data <i>user</i> tata usaha sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator melakukan <i>login</i> 2. Administrator masuk ke halaman Data Tata Usaha 3. Administrator memilih <i>user</i> tata usaha yang ingin dihapus 4. Administrator menekan tombol Hapus
Masukan	-
Keluaran yang Diharapkan	Data <i>user</i> tata usaha terhapus dari basis data
Hasil Pengujian	Data <i>user</i> tata usaha berhasil dihapus dari basis data. Pengujian berhasil.

Data *user* tata usaha yang dipilih administrator berada pada *DataGridView*. Kemudian administrator menekan tombol Hapus. Kemudian sistem akan memberikan konfirmasi apakah administrator yakin ingin menghapus data. Setelah administrator memberikan konfirmasi yakin maka sistem akan menghapus data

user tata usaha kemudian menghapusnya dari basis data. Hasil data *user* tata usaha yang telah berhasil dihapus dapat dilihat pada Gambar 5-22.



Nama	Username

Gambar 5-22 Hasil Penghapusan Data *User* Tata Usaha

5.2.2.10. Pengujian Menambah Data Mata Kuliah

Berikut ini membahas tentang pengujian menambah data mata kuliah. Pada proses menambah data mata kuliah, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Mata Kuliah. Skenario pengujian menambah data mata kuliah dapat dilihat pada Tabel 5-16.

Tabel 5-16 Skenario Pengujian Menambah Data Mata Kuliah

Kode Pengujian	PUJ.UC-00010
Nama	Pengujian Menambah Data Mata Kuliah
Referensi Kasus Penggunaan	UC-0004
Tujuan	Memeriksa fungsi untuk menambah data mata kuliah berfungsi atau tidak
Kondisi Awal	Data mata kuliah belum ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Mata Kuliah 3. Administrator atau tata usaha menekan tombol Tambah 4. Administrator atau tata usaha mengisi data mata kuliah 5. Administrator atau tata usaha menekan tombol Simpan
Masukan	Data mata kuliah baru
Keluaran yang Diharapkan	Data mata kuliah tersimpan ke dalam basis data
Hasil Pengujian	Data mata kuliah berhasil ditambahkan ke dalam basis data. Pengujian berhasil.

Data yang perlu diisi oleh administrator atau tata usaha seperti pada Gambar 5-23.



Input Data

Kode MK: KI141447

Mata Kuliah: TKMI

SKS: 3

Batal Simpan

Gambar 5-23 Pengisian Data Mata Kuliah

Selanjutnya administrator atau tata usaha menekan tombol Simpan dan sistem akan menyimpan data mata kuliah yang baru kemudian menampilkannya pada *DataGridView*. Hasil data mata kuliah yang telah berhasil ditambahkan dapat dilihat pada Gambar 5-24.



	Kode MK	Mata Kuliah	SKS
▶	KI141447	TKMI	3

Gambar 5-24 Hasil Penambahan Data Mata Kuliah

5.2.2.11. Pengujian Mengubah Data Mata Kuliah

Berikut ini membahas tentang pengujian mengubah data mata kuliah. Pada proses mengubah data mata kuliah, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Mata Kuliah. Skenario pengujian menambah data mata kuliah dapat dilihat pada Tabel 5-17.

Tabel 5-17 Skenario Pengujian Mengubah Data Mata Kuliah

Kode Pengujian	PUJ.UC-0011
Nama	Pengujian Mengubah Data Mata Kuliah
Referensi Kasus Penggunaan	UC-0005
Tujuan	Memeriksa fungsi untuk mengubah data mata kuliah berfungsi atau tidak
Kondisi Awal	Data mata kuliah sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Mata Kuliah 3. Administrator atau tata usaha memilih mata kuliah yang ingin diubah 4. Administrator atau tata usaha menekan tombol Ubah 5. Administrator atau tata usaha mengisi data mata kuliah yang ingin diubah 6. Administrator atau tata usaha menekan tombol Simpan
Masukan	Data mata kuliah baru
Keluaran yang Diharapkan	Data mata kuliah tersimpan ke dalam basis data
Hasil Pengujian	Data mata kuliah berhasil diubah dan diperbarui ke dalam basis data. Pengujian berhasil.

Data mata kuliah yang dipilih administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Ubah. Data perlu diisi untuk diubah oleh administrator atau tata usaha seperti pada Gambar 5-25.



Input Data

Kode MK: KI141447

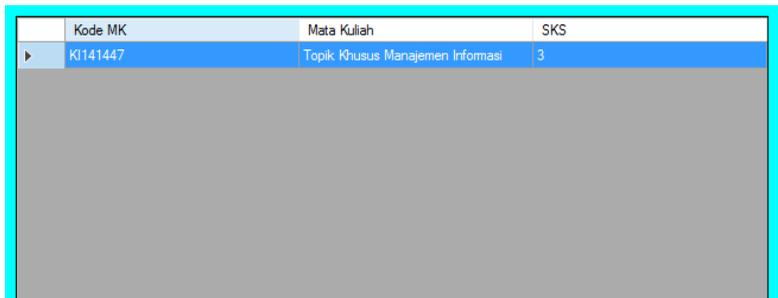
Mata Kuliah: Topik Khusus Manajemen Informasi

SKS: 3

Batal Simpan

Gambar 5-25 Pengubahan Data Mata Kuliah

Selanjutnya administrator atau tata usaha menekan tombol Simpan dan sistem akan menyimpan data mata kuliah yang baru kemudian menampilkannya pada *DataGridView*. Hasil data mata kuliah yang telah berhasil diubah dan diperbarui dapat dilihat pada Gambar 5-26.



Kode MK	Mata Kuliah	SKS
KI141447	Topik Khusus Manajemen Informasi	3

Gambar 5-26 Hasil Pengubahan Data Mata Kuliah

5.2.2.12. Pengujian Menghapus Data Mata Kuliah

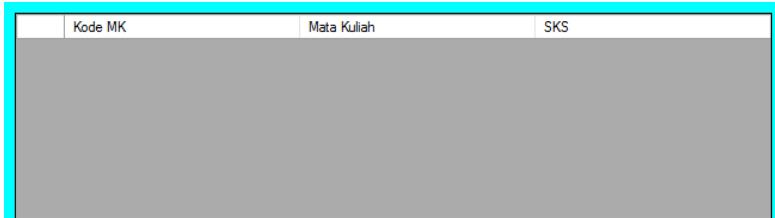
Berikut ini membahas tentang pengujian menghapus data mata kuliah. Pada proses menghapus data mata kuliah, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Mata Kuliah. Skenario pengujian menghapus data mata kuliah dapat dilihat pada Tabel 5-18.

Tabel 5-18 Skenario Pengujian Menghapus Data Mata Kuliah

Kode Pengujian	PUJ.UC-0012
Nama	Pengujian Menghapus Data Mata Kuliah
Referensi Kasus Penggunaan	UC-0006
Tujuan	Memeriksa fungsi untuk menghapus data mata kuliah berfungsi atau tidak
Kondisi Awal	Data mata kuliah sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Mata Kuliah 3. Administrator atau tata usaha memilih mata kuliah yang ingin dihapus 4. Administrator atau tata usaha menekan tombol Hapus
Masukan	-
Keluaran yang Diharapkan	Data mata kuliah terhapus dari basis data
Hasil Pengujian	Data mata kuliah berhasil dihapus dari basis data. Pengujian berhasil.

Data mata kuliah yang dipilih administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Hapus. Kemudian sistem akan memberikan

konfirmasi apakah administrator atau tata usaha yakin ingin menghapus data. Setelah administrator atau tata usaha memberikan konfirmasi yakin maka sistem akan menghapus data mata kuliah dari basis data. Hasil data mata kuliah yang telah berhasil dihapus dapat dilihat pada Gambar 5-27.



Gambar 5-27 Hasil Penghapusan Data Mata Kuliah

5.2.2.13. Pengujian Mengubah Data Mahasiswa

Berikut ini membahas tentang pengujian mengubah data mahasiswa. Pada proses mengubah data mahasiswa, administrator atau tata usaha diharuskan sudah masuk ke dalam sistem dan berada pada halaman Data Mahasiswa. Skenario pengujian mengubah data mahasiswa dapat dilihat pada Tabel 5-19.

Tabel 5-19 Skenario Pengujian Mengubah Data Mahasiswa

Kode Pengujian	PUJ.UC-0013
Nama	Pengujian Mengubah Data Mahasiswa
Referensi Kasus Penggunaan	UC-0009
Tujuan	Memeriksa fungsi untuk mengubah data mahasiswa berfungsi atau tidak
Kondisi Awal	Data mahasiswa sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Mahasiswa

	<ol style="list-style-type: none"> 3. Administrator atau tata usaha memilih mahasiswa yang ingin diubah 4. Administrator atau tata usaha menekan tombol Ubah 5. Administrator atau tata usaha mengisi data mahasiswa yang ingin diubah 6. Administrator atau tata usaha menekan tombol Simpan
Masukan	Data baru mahasiswa
Keluaran yang Diharapkan	Data baru mahasiswa tersimpan ke dalam basis data
Hasil Pengujian	Data mahasiswa berhasil diubah dan diperbarui ke dalam basis data. Pengujian berhasil.

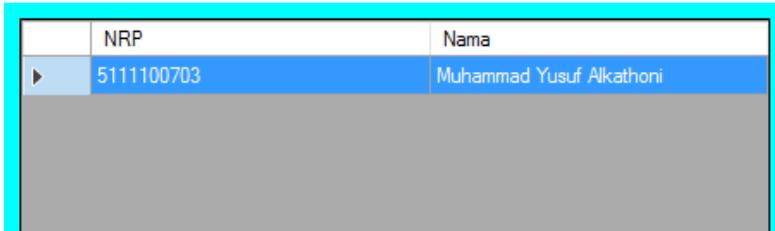
Data mahasiswa yang dipilih administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Ubah. Data perlu diisi untuk diubah oleh administrator atau tata usaha seperti pada Gambar 5-28.

The image shows a web form titled "Input Data" with a light blue background. It contains two input fields: "NRP" with the value "5111100703" and "Nama" with the value "Muhammad Yusuf Alkathoni". Below the fields are two buttons: "Batal" and "Simpan".

Gambar 5-28 Perubahan Data Mahasiswa

Selanjutnya administrator atau tata usaha menekan tombol Simpan dan sistem akan menyimpan data mahasiswa yang baru ke basis

data kemudian menampilkannya pada *DataGridView*. Hasil data mahasiswa yang telah berhasil diubah dan diperbarui dapat dilihat pada Gambar 5-29.



	NRP	Nama
▶	5111100703	Muhammad Yusuf Alkathoni

Gambar 5-29 Hasil Pengubahan Data Mahasiswa

5.2.2.14. Pengujian Menghapus Data Mahasiswa

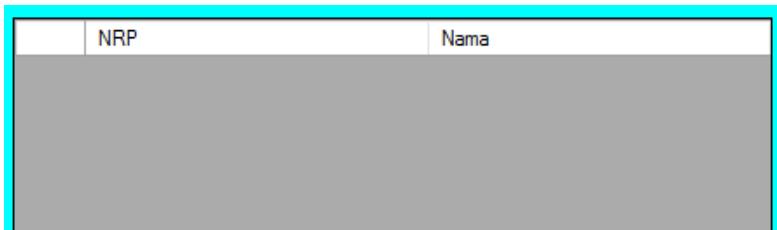
Berikut ini membahas tentang pengujian menghapus data mahasiswa. Pada proses menghapus data mahasiswa, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Mahasiswa. Skenario pengujian menghapus data mahasiswa dapat dilihat pada Tabel 5-20.

Tabel 5-20 Skenario Pengujian Menghapus Data Mahasiswa

Kode Pengujian	PUJ.UC-0014
Nama	Pengujian Menghapus Data Mahasiswa
Referensi Kasus Penggunaan	UC-0010
Tujuan	Memeriksa fungsi untuk menghapus data mahasiswa berfungsi atau tidak
Kondisi Awal	Data mahasiswa sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Mahasiswa 3. Administrator atau tata usaha memilih mahasiswa yang ingin dihapus

	4. Administrator atau tata usaha menekan tombol Hapus
Masukan	-
Keluaran yang Diharapkan	Data mahasiswa terhapus dari basis data
Hasil Pengujian	Data mahasiswa berhasil dihapus dari basis data. Pengujian berhasil.

Data mahasiswa yang dipilih oleh administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Hapus. Kemudian sistem akan memberikan konfirmasi apakah administrator atau tata usaha yakin ingin menghapus data. Setelah administrator atau tata usaha memberikan konfirmasi yakin, maka sistem akan menghapus data mahasiswa dari basis data. Hasil data mahasiswa yang telah berhasil dihapus dapat dilihat pada Gambar 5-30.



NRP	Nama

Gambar 5-30 Hasil Penghapusan Data Mahasiswa

5.2.2.15. Pengujian Melihat Data Mahasiswa

Berikut ini membahas tentang pengujian melihat data mahasiswa. Pada proses melihat data mahasiswa, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Mahasiswa. Skenario pengujian melihat data mahasiswa dapat dilihat pada Tabel 5-21.

Tabel 5-21 Skenario Pengujian Melihat Data Mahasiswa

Kode Pengujian	PUJ.UC-0015
Nama	Pengujian Melihat Data Mahasiswa
Referensi Kasus Penggunaan	UC-0011
Tujuan	Memeriksa fungsi untuk melihat data mahasiswa berfungsi atau tidak
Kondisi Awal	Data mahasiswa sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Mahasiswa 3. Administrator atau tata usaha memilih mahasiswa yang ingin dilihat 4. Administrator atau tata usaha menekan tombol Lihat Data
Masukan	-
Keluaran yang Diharapkan	Data mahasiswa ditampilkan oleh sistem
Hasil Pengujian	Data mahasiswa berhasil ditampilkan oleh sistem. Pengujian berhasil.

Data mahasiswa yang dipilih oleh administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Lihat Data dan sistem akan menampilkan data mahasiswa yang dipilih. Hasil data mahasiswa yang telah berhasil ditampilkan dapat dilihat pada Gambar 5-31.



Gambar 5-31 Hasil Lihat Data Mahasiswa

5.2.2.16. Pengujian Menambah Data Dosen

Berikut ini membahas tentang pengujian menambah data dosen. Pada proses menambah data dosen, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Dosen. Skenario pengujian menambah data dosen dapat dilihat pada Tabel 5-22.

Tabel 5-22 Skenario Pengujian Menambah Data Dosen

Kode Pengujian	PUJ.UC-0016
Nama	Pengujian Menambah Data Dosen
Referensi Kasus Penggunaan	UC-0012

Tujuan	Memeriksa fungsi untuk menambah data dosen berfungsi atau tidak
Kondisi Awal	Data dosen belum ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Dosen 3. Administrator atau tata usaha menekan tombol Tambah 4. Administrator atau tata usaha mengisi data dosen 5. Administrator atau tata usaha menekan tombol Simpan
Masukan	Data dosen baru
Keluaran yang Diharapkan	Data dosen tersimpan ke dalam basis data
Hasil Pengujian	Data dosen berhasil ditambahkan ke dalam basis data. Pengujian berhasil.

Data yang perlu diisi oleh administrator atau tata usaha seperti pada Gambar 5-32.

The image shows a web form titled "Input Data" with a light blue background. It contains four input fields, each with a label to its left:

- Kode Dosen:** A text input field containing the value "HG".
- Nama:** A text input field containing the value "Dr.tech. Ir. Raden Venantius Hari Ginardi, M. Sc.".
- Password:** A text input field with three black dots, indicating a password field.
- Konfirmasi Password:** A text input field with three black dots, indicating a confirmation password field.

At the bottom of the form, there are two buttons: "Batal" (Cancel) on the left and "Simpan" (Save) on the right.

Gambar 5-32 Pengisian Data Dosen

Selanjutnya administrator atau tata usaha menekan tombol Simpan dan sistem akan menyimpan data dosen yang baru kemudian menampilkannya pada *DataGridView*. Hasil data dosen yang telah berhasil ditambahkan dapat dilihat pada Gambar 5-33.

	Kode Dosen	Nama
▶	HG	Dr.tech. Ir. Raden Venantius Hari Ginardi, M. Sc.

Gambar 5-33 Hasil Penambahan Data Dosen

5.2.2.17. Pengujian Mengubah Data Dosen

Berikut ini membahas tentang pengujian mengubah data dosen. Pada proses mengubah data dosen, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Dosen. Skenario pengujian mengubah data dosen dapat dilihat pada Tabel 5-23.

Tabel 5-23 Skenario Pengujian Mengubah Data Dosen

Kode Pengujian	PUJ.UC-0017
Nama	Pengujian Mengubah Data Dosen
Referensi Kasus Penggunaan	UC-0013
Tujuan	Memeriksa fungsi untuk mengubah data dosen berfungsi atau tidak
Kondisi Awal	Data dosen sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Dosen 3. Administrator atau tata usaha memilih dosen yang ingin diubah

	<ol style="list-style-type: none"> 4. Administrator atau tata usaha menekan tombol Ubah 5. Administrator atau tata usaha mengisi data dosen yang ingin diubah 6. Administrator atau tata usaha menekan tombol Simpan
Masukan	Data dosen baru
Keluaran yang Diharapkan	Data dosen tersimpan ke dalam basis data
Hasil Pengujian	Data dosen berhasil diubah dan diperbarui ke dalam basis data. Pengujian berhasil.

Data dosen yang dipilih oleh administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Ubah. Data yang perlu diisi untuk diubah oleh administrator atau tata usaha seperti pada Gambar 5-34.

The image shows a web form titled "Input Data" with a light blue border. It contains two input fields: "Kode Dosen" with the value "HG" and "Nama" with the value "Dr.tech. Ir. R. V. Hari Ginardi, M. Sc.". At the bottom, there are two buttons: "Batal" and "Simpan".

Gambar 5-34 Pengubahan Data Dosen

Selanjutnya administrator atau tata usaha menekan tombol Simpan dan sistem akan menyimpan data dosen yang baru kemudian menampilkannya pada *DataGridView*. Hasil data dosen yang telah berhasil diubah dan diperbarui dapat dilihat pada Gambar 5-35.



	Kode Dosen	Nama
▶	HG	Drtech. Ir. R. V. Hari Ginardi, M. Sc.

Gambar 5-35 Hasil Pengubahan Data Dosen

5.2.2.18. Pengujian Menghapus Data Dosen

Berikut ini membahas tentang pengujian menghapus data dosen. Pada proses menghapus data dosen, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Dosen. Skenario pengujian menghapus data dosen dapat dilihat pada Tabel 5-24.

Tabel 5-24 Skenario Pengujian Menghapus Data Dosen

Kode Pengujian	PUJ.UC-0018
Nama	Pengujian Menghapus Data Dosen
Referensi Kasus Penggunaan	UC-0014
Tujuan	Memeriksa fungsi untuk menghapus data dosen berfungsi atau tidak
Kondisi Awal	Data dosen sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Dosen 3. Administrator atau tata usaha memilih dosen yang ingin dihapus

	4. Administrator menekan tombol Hapus
Masukan	-
Keluaran yang Diharapkan	Data dosen terhapus dari basis data
Hasil Pengujian	Data dosen berhasil dihapus dari basis data. Pengujian berhasil.

Data dosen yang dipilih oleh administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Hapus. Kemudian sistem akan memberikan konfirmasi apakah administrator atau tata usaha yakin ingin menghapus data. Setelah administrator atau tata usaha memberikan konfirmasi yakin, maka sistem akan menghapus data dosen dari basis data. Hasil data dosen yang telah berhasil dihapus dapat dilihat pada Gambar 5-36.



Kode Dosen	Nama

Gambar 5-36 Hasil Penghapusan Data Dosen

5.2.2.19. Pengujian Menambah Data Jadwal

Berikut ini membahas tentang pengujian menambah data jadwal. Pada proses menambah data jadwal, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Jadwal Perkuliahan. Skenario pengujian menambah data jadwal dapat dilihat pada Tabel 5-25.

Tabel 5-25 Skenario Pengujian Menambah Data Jadwal

Kode Pengujian	PUJ.UC-0019
Nama	Pengujian Menambah Data Jadwal
Referensi Kasus Penggunaan	UC-0015
Tujuan	Memeriksa fungsi untuk menambah data jadwal berfungsi atau tidak
Kondisi Awal	Data jadwal belum ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Jadwal Perkuliahan 3. Administrator atau tata usaha menekan tombol Tambah 4. Administrator atau tata usaha mengisi data jadwal 5. Administrator atau tata usaha menekan tombol Simpan
Masukan	Data jadwal baru
Keluaran yang Diharapkan	Data jadwal tersimpan ke dalam basis data
Hasil Pengujian	Data jadwal berhasil ditambahkan ke dalam basis data. Pengujian berhasil.

Data yang perlu diisi oleh administrator atau tata usaha seperti pada Gambar 5-37.

Input Data

Dosen Dr.tech. Ir. R. V. Hari Ginardi, M. S

Mata Kuliah Topik Khusus Manajemen Informa:

Kelas A

Ruang IF-105A

Hari Jumat

Jam 13:30:00 - 16:00:00

Batal **Simpan**

Gambar 5-37 Pengisian Data Jadwal

Selanjutnya administrator atau tata usaha menekan tombol Simpan dan sistem akan menyimpan data jadwal yang baru kemudian menampilkannya pada *DataGridView*. Hasil data jadwal yang telah berhasil ditambahkan dapat dilihat pada Gambar 5-38.

ID Jadwal	Dosen	Mata Kuliah	Kelas	Ruang	Hari	Jam Mulai	Jam Selesai
K1141447/A...	Dr.tech. Ir. ...	Topik Khus...	A	IF-105A	Jumat	133000	160000

Gambar 5-38 Hasil Penambahan Data Jadwal

5.2.2.20. Pengujian Mengubah Data Jadwal

Berikut ini membahas tentang pengujian mengubah data jadwal. Pada proses mengubah data jadwal, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Jadwal Perkuliahan. Skenario pengujian menambah data jadwal dapat dilihat pada Tabel 5-26.

Tabel 5-26 Skenario Pengujian Mengubah Data Jadwal

Kode Pengujian	PUJ.UC-0020
Nama	Pengujian Mengubah Data Jadwal
Referensi Kasus Penggunaan	UC-0016
Tujuan	Memeriksa fungsi untuk mengubah data jadwal berfungsi atau tidak
Kondisi Awal	Data jadwal sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Jadwal Perkuliahan 3. Administrator atau tata usaha memilih jadwal yang ingin diubah 4. Administrator atau tata usaha menekan tombol Ubah 5. Administrator atau tata usaha mengisi data jadwal yang ingin diubah 6. Administrator atau tata usaha menekan tombol Simpan
Masukan	Data jadwal baru
Keluaran yang Diharapkan	Data jadwal tersimpan ke dalam basis data
Hasil Pengujian	Data jadwal berhasil diubah dan diperbarui ke dalam basis data. Pengujian berhasil.

Data jadwal yang dipilih oleh administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Ubah. Data yang perlu diisi untuk diubah oleh administrator atau tata usaha seperti pada Gambar 5-39.

The screenshot shows a form titled "Input Data" with the following fields:

- Dosen:** Dr.tech. Ir. R. V. Hari Ginardi, M. S
- Mata Kuliah:** Topik Khusus Manajemen Infoma:
- Kelas:** A
- Ruang:** Lab. Alpro
- Hari:** Selasa
- Jam:** 16:00:00 - 18:00:00

At the bottom of the form are two buttons: "Batal" and "Simpan".

Gambar 5-39 Pengubahan Data Jadwal

Selanjutnya administrator atau tata usaha menekan tombol Simpan dan sistem akan menyimpan data jadwal yang baru kemudian menampilkannya pada *DataGridView*. Hasil data jadwal yang telah berhasil diubah dan diperbarui dapat dilihat pada Gambar 5-40.

	ID Jadwal	Dosen	Mata Kuliah	Kelas	Ruang	Hari	Jam Mulai	Jam Selesai
▶	KI141447/A...	Dr.tech. Ir. ...	Topik Khus...	A	Lab. Alpro	Selasa	160000	180000

Gambar 5-40 Hasil Pengubahan Data Jadwal

5.2.2.21. Pengujian Menghapus Data Jadwal

Berikut ini membahas tentang pengujian menghapus data jadwal. Pada proses menghapus data jadwal, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Jadwal Perkuliahan. Skenario pengujian menambah data jadwal dapat dilihat pada Tabel 5-27.

Tabel 5-27 Skenario Pengujian Menghapus Data Jadwal

Kode Pengujian	PUJ.UC-0021
Nama	Pengujian Menghapus Data Jadwal
Referensi Kasus Penggunaan	UC-0017
Tujuan	Memeriksa fungsi untuk menghapus data jadwal berfungsi atau tidak
Kondisi Awal	Data jadwal sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Jadwal Perkuliahan 3. Administrator atau tata usaha memilih jadwal yang ingin dihapus 4. Administrator atau tata usaha menekan tombol Hapus
Masukan	-
Keluaran yang Diharapkan	Data jadwal terhapus dari basis data
Hasil Pengujian	Data jadwal berhasil dihapus dari basis data. Pengujian berhasil.

Data jadwal yang dipilih oleh administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Hapus. Kemudian sistem akan memberikan konfirmasi apakah administrator atau tata usaha yakin ingin

menghapus data. Setelah administrator atau tata usaha memberikan konfirmasi yakin, maka sistem akan menghapus data jadwal dari basis data. Hasil data jadwal yang telah berhasil dihapus dapat dilihat pada Gambar 5-41.

ID Jadwal	Dosen	Mata Kuliah	Kelas	Ruang	Hari	Jam Mulai	Jam Selesai
[Empty table body]							

Gambar 5-41 Hasil Penghapusan Data Jadwal

5.2.2.22. Pengujian Menambah Data Mahasiswa Per Kelas

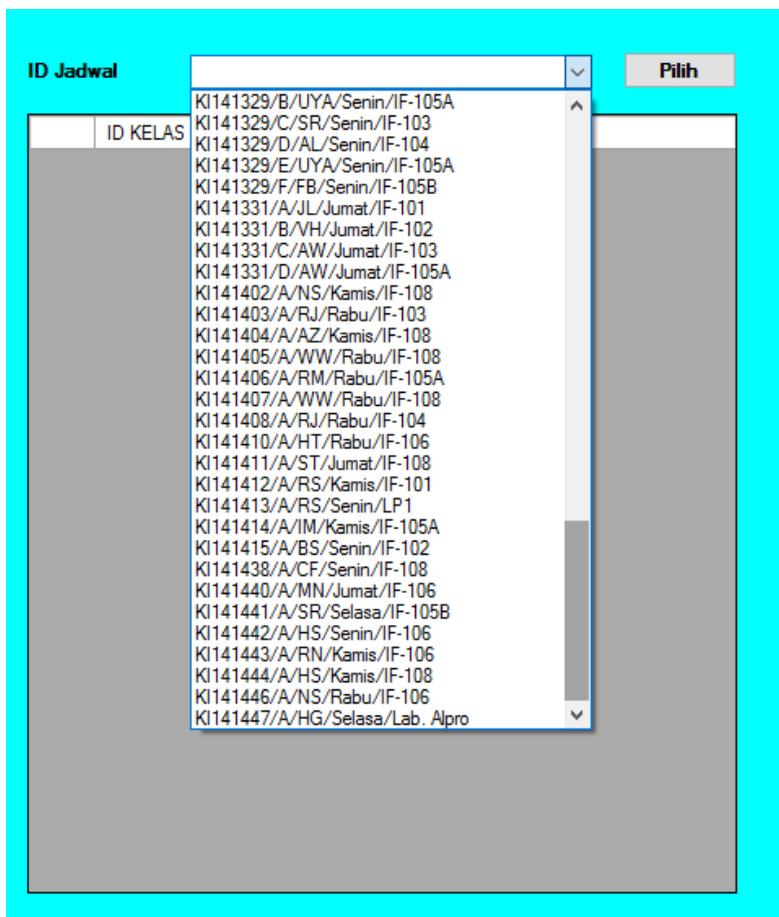
Berikut ini membahas tentang pengujian menambah data mahasiswa per kelas. Pada proses menambah data mahasiswa per kelas, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Mahasiswa Per Kelas. Skenario pengujian menambah data mahasiswa per kelas dapat dilihat pada Tabel 5-28.

Tabel 5-28 Skenario Pengujian Menambah Data Mahasiswa Per Kelas

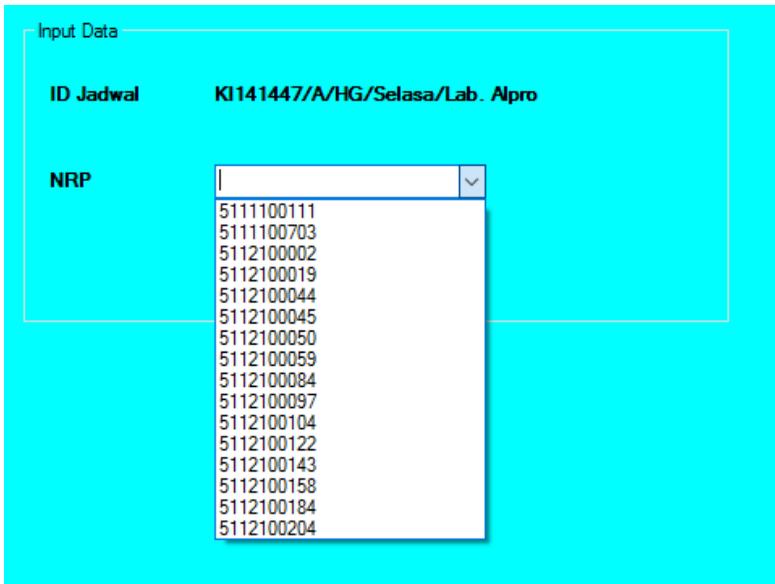
Kode Pengujian	PUJ.UC-0022
Nama	Pengujian Menambah Data Mahasiswa Per Kelas
Referensi Kasus Penggunaan	UC-0018
Tujuan	Memeriksa fungsi untuk menambah data mahasiswa per kelas berfungsi atau tidak
Kondisi Awal	Data mahasiswa per kelas belum ada
Skenario	1. Administrator atau tata usaha melakukan <i>login</i>

	<ol style="list-style-type: none"> 2. Administrator atau tata usaha masuk ke halaman Data Mahasiswa Per Kelas 3. Administrator atau tata usaha memilih kelas yang ingin ditambah mahasiswanya berdasarkan ID Jadwal 4. Administrator atau tata usaha menekan tombol Pilih 5. Administrator atau tata usaha menekan tombol Tambah 6. Administrator atau tata usaha memilih mahasiswa yang ingin ditambahkan pada kelas yang telah dipilih berdasarkan NRP 7. Administrator atau tata usaha menekan tombol Masukkan
Masukan	Data mahasiswa per kelas baru
Keluaran yang Diharapkan	Data mahasiswa per kelas tersimpan ke dalam basis data
Hasil Pengujian	Data mahasiswa per kelas berhasil ditambahkan ke dalam basis data. Pengujian berhasil.

Data kelas berdasarkan ID Jadwal dan data NRP mahasiswa yang bisa dipilih oleh administrator atau tata usaha seperti pada Gambar 5-42 dan Gambar 5-43.



Gambar 5-42 Daftar Data Kelas Berdasarkan ID Jadwal



Input Data

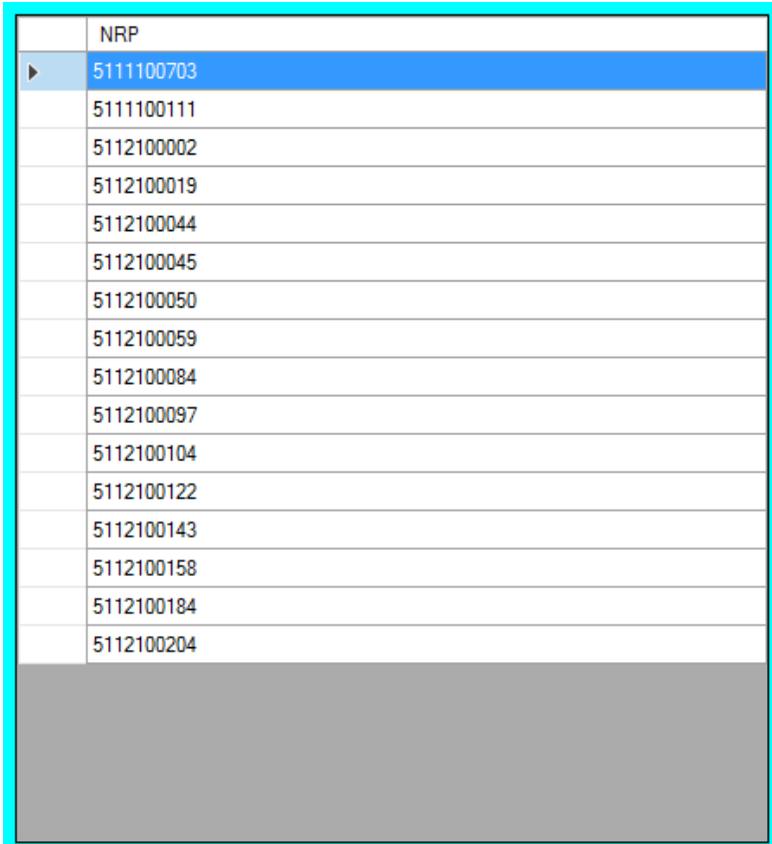
ID Jadwal KI141447/A/HG/Selasa/Lab. Alpro

NRP

5111100111
5111100703
5112100002
5112100019
5112100044
5112100045
5112100050
5112100059
5112100084
5112100097
5112100104
5112100122
5112100143
5112100158
5112100184
5112100204

Gambar 5-43 Daftar NRP Mahasiswa

Selanjutnya administrator atau tata usaha menekan tombol Masukkan dan sistem akan menyimpan data mahasiswa per kelas yang baru kemudian menampilkannya pada *DataGridView*. Hasil data mahasiswa per kelas yang telah berhasil ditambahkan dapat dilihat pada Gambar 5-44.



The image shows a screenshot of a web application interface displaying a table of student data. The table has a header row with the label 'NRP'. The first row is highlighted in blue and contains the value '5111100703'. The subsequent rows contain a list of NRP values: 5111100111, 5112100002, 5112100019, 5112100044, 5112100045, 5112100050, 5112100059, 5112100084, 5112100097, 5112100104, 5112100122, 5112100143, 5112100158, 5112100184, and 5112100204. The bottom portion of the table is obscured by a grey rectangular area.

NRP
5111100703
5111100111
5112100002
5112100019
5112100044
5112100045
5112100050
5112100059
5112100084
5112100097
5112100104
5112100122
5112100143
5112100158
5112100184
5112100204

Gambar 5-44 Hasil Penambahan Data Mahasiswa Per Kelas

5.2.2.23. Pengujian Menghapus Data Mahasiswa Per Kelas

Berikut ini membahas tentang pengujian menghapus data mahasiswa per kelas. Pada proses menghapus data mahasiswa per kelas, administrator atau tata usaha diharuskan telah masuk ke dalam sistem dan berada pada halaman Data Mahasiswa Per Kelas.

Skenario pengujian menghapus data mahasiswa per kelas dapat dilihat pada Tabel 5-29.

Tabel 5-29 Skenario Pengujian Menghapus Data Mahasiswa Per Kelas

Kode Pengujian	PUJ.UC-0023
Nama	Pengujian Menghapus Data Mahasiswa Per Kelas
Referensi Kasus Penggunaan	UC-0019
Tujuan	Memeriksa fungsi untuk menghapus data mahasiswa per kelas berfungsi atau tidak
Kondisi Awal	Data mahasiswa per kelas sudah ada
Skenario	<ol style="list-style-type: none"> 1. Administrator atau tata usaha melakukan <i>login</i> 2. Administrator atau tata usaha masuk ke halaman Data Mahasiswa Per Kelas 3. Administrator atau tata usaha memilih kelas berdasarkan ID Jadwal 4. Administrator atau tata usaha menekan tombol Pilih 5. Administrator atau tata usaha memilih mahasiswa yang ingin dihapus 6. Administrator atau tata usaha menekan tombol Hapus
Masukan	-
Keluaran yang Diharapkan	Data mahasiswa per kelas terhapus dari basis data
Hasil Pengujian	Data mahasiswa per kelas berhasil dihapus dari basis data. Pengujian berhasil.

Data kelas berdasarkan ID Jadwal yang bisa dipilih oleh administrator atau tata usaha seperti pada Gambar 5-42. Data

mahasiswa per kelas yang dipilih administrator atau tata usaha berada pada *DataGridView*. Kemudian administrator atau tata usaha menekan tombol Hapus. Kemudian sistem akan memberikan konfirmasi apakah administrator atau tata usaha yakin ingin menghapus data. Setelah administrator atau tata usaha memberikan konfirmasi yakin, maka sistem akan menghapus data mahasiswa per kelas dari basis data. Hasil data mahasiswa per kelas yang telah berhasil dihapus dapat dilihat pada Gambar 5-45.



Gambar 5-45 Hasil Penghapusan Data Mahasiswa Per Kelas

5.2.2.24. Pengujian Mengubah *Password*

Berikut ini membahas tentang pengujian mengubah *password*. Pada proses mengubah *password*, administrator, tata usaha, atau dosen diharuskan telah masuk ke dalam sistem dan berada pada halaman Beranda masing-masing. Skenario pengujian mengubah *password* dapat dilihat pada Tabel 5-30.

Tabel 5-30 Skenario Pengujian Mengubah *Password*

Kode Pengujian	PUJ.UC-0024
Nama	Pengujian Mengubah <i>Password</i>
Referensi Kasus Penggunaan	UC-0020
Tujuan	Memeriksa fungsi untuk mengubah <i>password</i> berfungsi atau tidak
Kondisi Awal	Data administrator, tata usaha, dan dosen sudah ada
Skenario	1. Pengguna melakukan <i>login</i>

	<ol style="list-style-type: none"> 2. Pengguna menekan tombol Ganti Password 3. Pengguna mengisi data pergantian <i>password</i> 4. Pengguna menekan tombol Simpan
Masukan	Data <i>password</i> baru
Keluaran yang Diharapkan	Data <i>password</i> baru tersimpan ke dalam basis data
Hasil Pengujian	Data <i>password</i> baru berhasil ditambahkan ke dalam basis data. Pengujian berhasil.

Data yang perlu diisi oleh pengguna seperti pada Gambar 5-46.

Ganti Password

Muhammad Yusuf Alkathoni

Password Lama

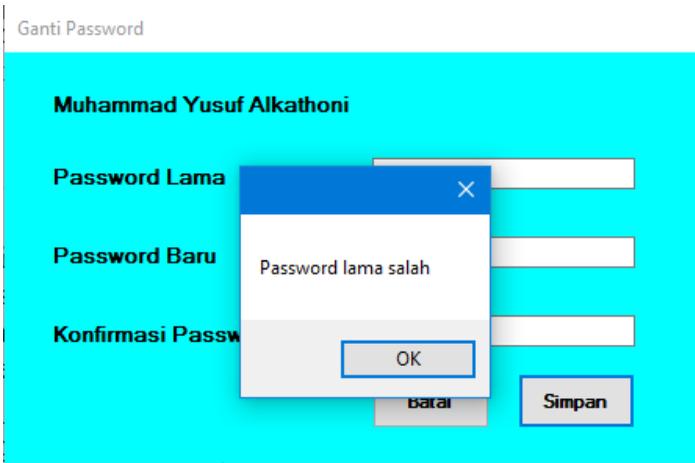
Password Baru

Konfirmasi Password

Batal **Simpan**

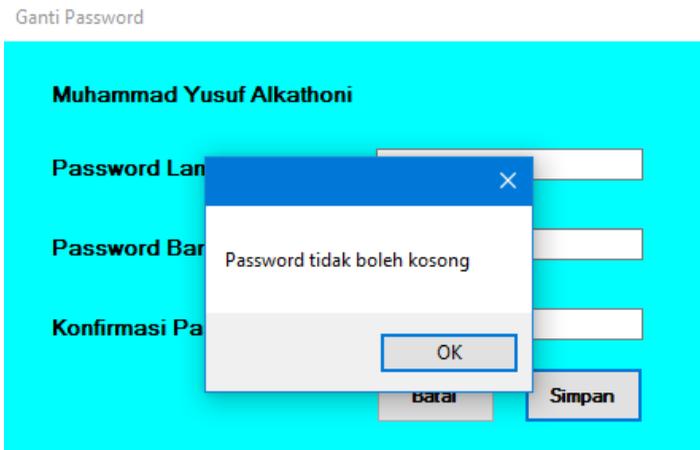
Gambar 5-46 Pengisian Ganti Password

Selanjutnya pengguna menekan tombol Simpan. Jika *password* lama yang diisi oleh pengguna salah, maka sistem akan menampilkan peringatan bahwa *password* lama yang diisi salah seperti pada Gambar 5-47.



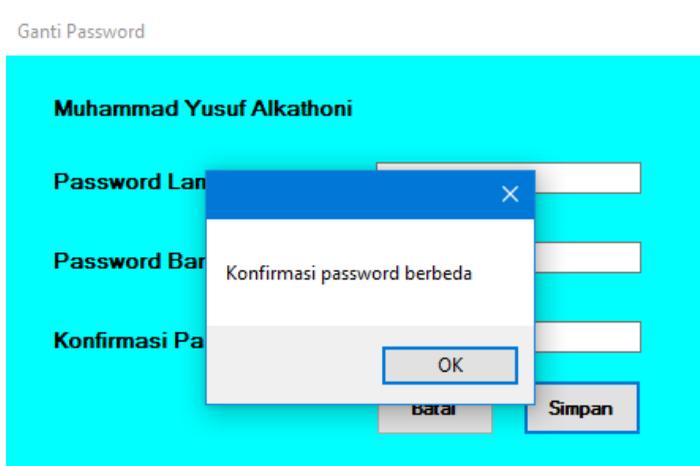
Gambar 5-47 Peringatan Password Lama Salah

Jika kolom *password* ada yang belum terisi, maka sistem akan menampilkan peringatan bahwa kolom *password* tidak boleh kosong seperti pada Gambar 5-48.



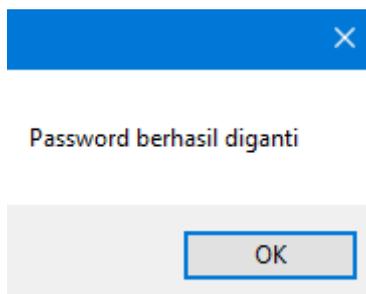
Gambar 5-48 Peringatan Password Tidak Boleh Kosong

Jika *password* baru dan konfirmasi *password* yang diisi oleh pengguna berbeda, maka sistem akan menampilkan peringatan bahwa konfirmasi *password* berbeda seperti pada Gambar 5-49.



Gambar 5-49 Peringatan Konfirmasi Password Berbeda

Jika kolom *password* terisi dengan benar, maka sistem akan menyimpan data *password* yang baru kemudian menampilkan konfirmasi bahwa *password* berhasil diubah seperti pada Gambar 5-50.



Gambar 5-50 Pesan Password Berhasil Diubah

5.3. Evaluasi Pengujian

Pada subbab ini akan membahas hasil evaluasi dari pengujian-pengujian yang telah dilakukan. Evaluasi yang diberikan meliputi evaluasi pengujian kebutuhan fungsional.

5.3.1. Evaluasi Pengujian

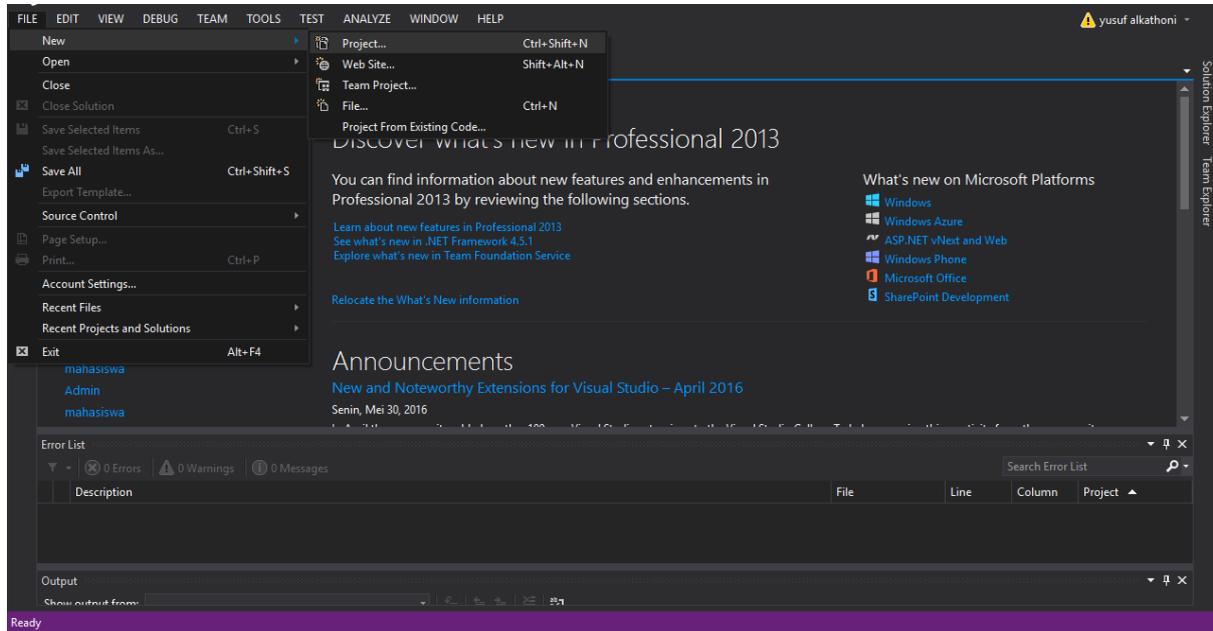
Pada subbab ini membahas tentang rangkuman hasil pengujian fungsionalitas yang dapat dilihat pada Tabel 5-31. Berdasarkan data pada tabel tersebut, semua skenario pengujian berhasil dan program berjalan dengan baik sehingga bisa disimpulkan bahwa fungsionalitas dari program telah bisa bekerja sesuai dengan yang diharapkan.

Tabel 5-31 Rangkuman Hasil Pengujian Fungsionalitas

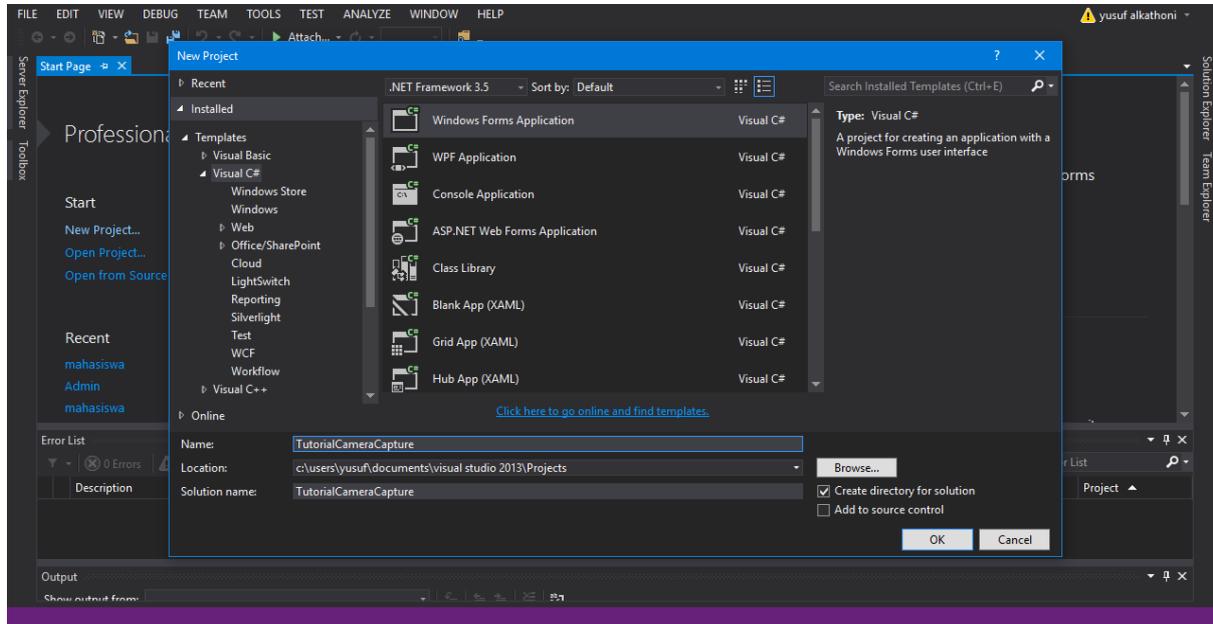
Kode Pengujian	Nama	Hasil
PUJ.UC-0001	Pengujian Menambah Data Mahasiswa	Berhasil
PUJ.UC-0002	Pengujian Menambah Foto Mahasiswa	Berhasil
PUJ.UC-0003	Pengujian Mengisi Berita Acara	Berhasil
PUJ.UC-0004	Pengujian Melakukan Absensi	Berhasil dengan persentase 93,33%
PUJ.UC-0005	Pengujian Melihat Absensi	Berhasil
PUJ.UC-0006	Pengujian Mengekstrak Data Absensi	Berhasil
PUJ.UC-0007	Pengujian Menambah Data User TU	Berhasil
PUJ.UC-0008	Pengujian Mengubah Data User TU	Berhasil

Kode Pengujian	Nama	Hasil
PUJ.UC-009	Pengujian Menghapus Data User TU	Berhasil
PUJ.UC-0010	Pengujian Menambah Data Mata Kuliah	Berhasil
PUJ.UC-0011	Pengujian Mengubah Data Mata Kuliah	Berhasil
PUJ.UC-0012	Pengujian Menghapus Data Mata Kuliah	Berhasil
PUJ.UC-0013	Pengujian Mengubah Data Mahasiswa	Berhasil
PUJ.UC-0014	Pengujian Menghapus Data Mahasiswa	Berhasil
PUJ.UC-0015	Pengujian Melihat Data Mahasiswa	Berhasil
PUJ.UC-0016	Pengujian Menambah Data Dosen	Berhasil
PUJ.UC-0017	Pengujian Mengubah Data Dosen	Berhasil
PUJ.UC-0018	Pengujian Menghapus Data Dosen	Berhasil
PUJ.UC-0019	Pengujian Menambah Data Jadwal	Berhasil
PUJ.UC-0020	Pengujian Mengubah Data Jadwal	Berhasil
PUJ.UC-0021	Pengujian Menghapus Data Jadwal	Berhasil
PUJ.UC-0022	Pengujian Menambah Data Mahasiswa Per Kelas	Berhasil
PUJ.UC-0023	Pengujian Menghapus Data Mahasiswa Per Kelas	Berhasil
PUJ.UC-0024	Pengujian Mengubah <i>password</i>	Berhasil

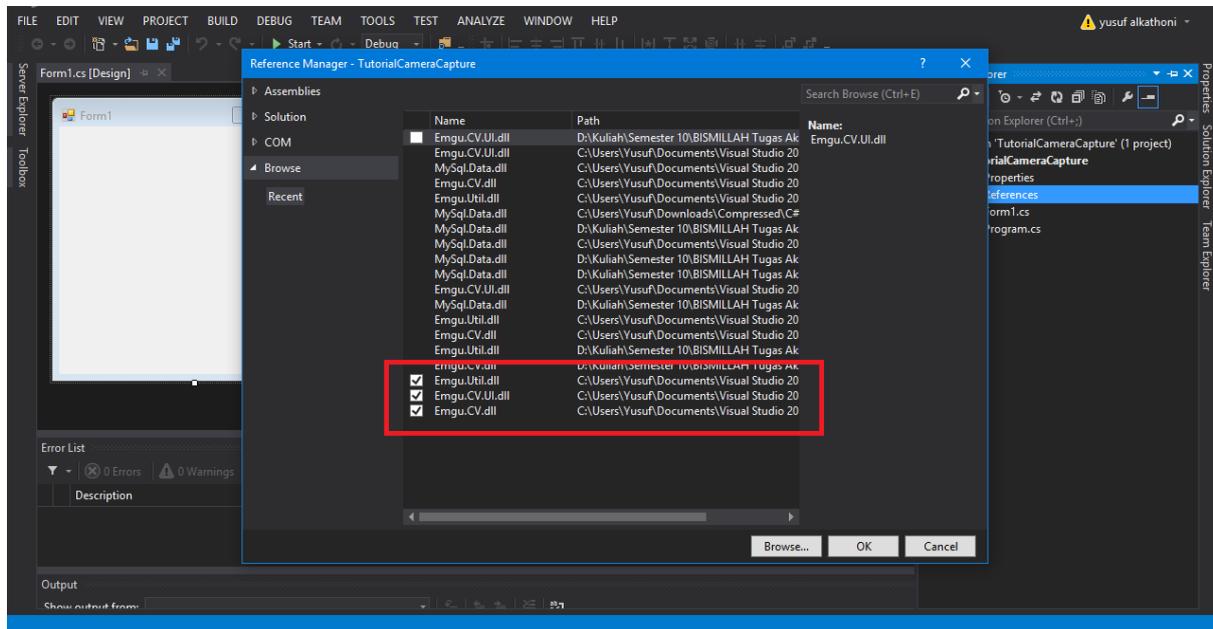
LAMPIRAN A – GAMBAR TUTORIAL CAMERA CAPTURE



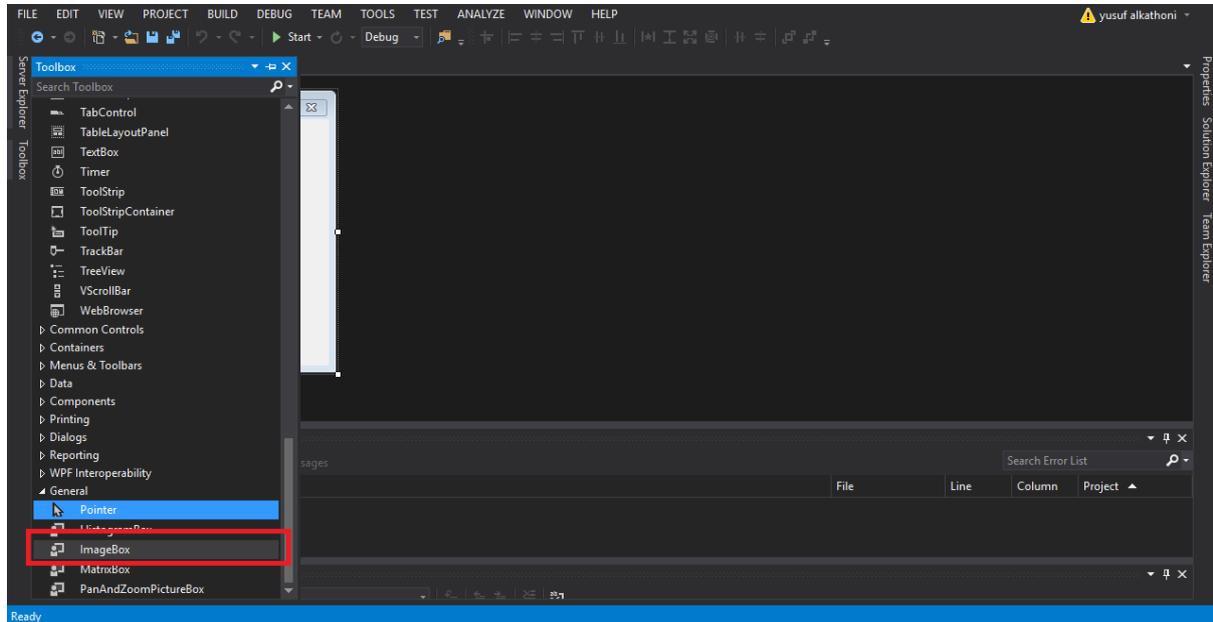
Gambar A-1 Buat Projek Baru pada Visual Studio



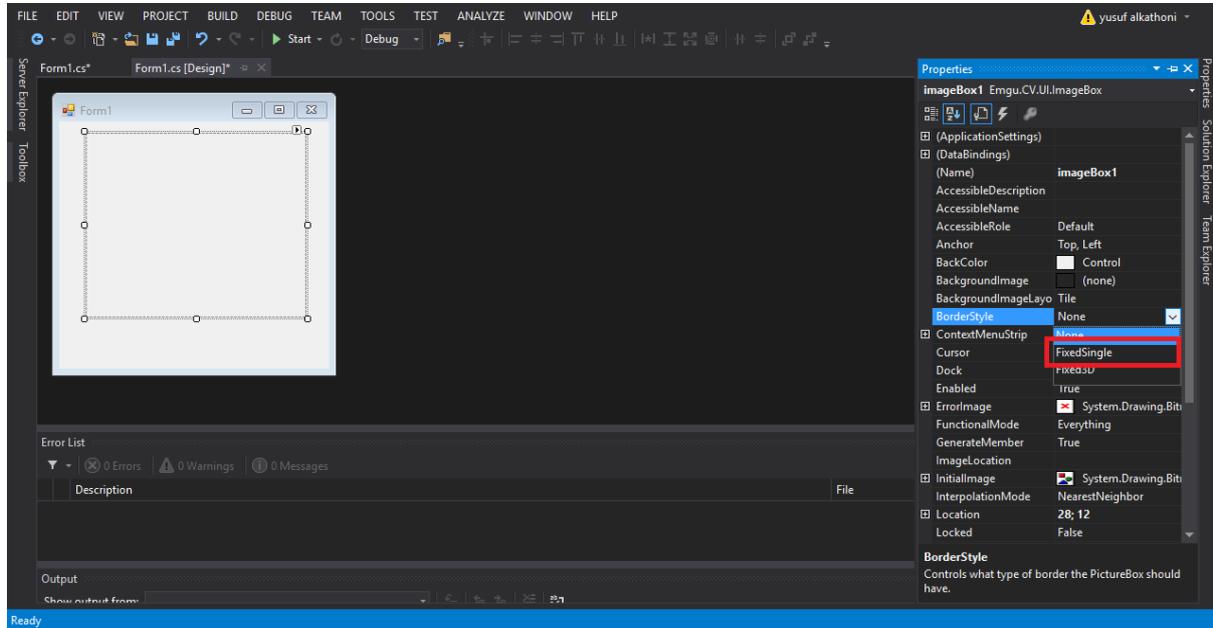
Gambar A-2 *Pilih Windows Form Application*



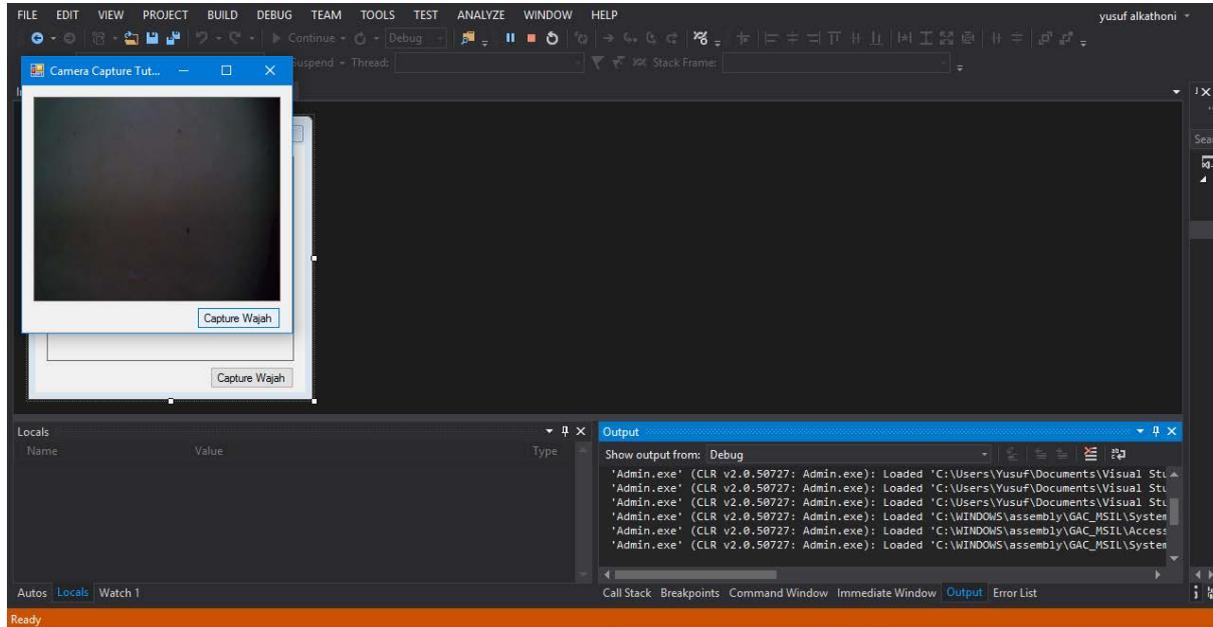
Gambar A-3 Tambahkan References Emgu CV



Gambar A-4 Tambahkan Image Box Emgu CV



Gambar A-5 Atur Border Style ImageBox Emgu CV menjadi Fixed Single



Gambar A-6 *Camera Capture* ketika Dijalankan

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini akan membahas tentang kesimpulan yang diambil selama pengerjaan tugas akhir serta saran-saran tentang pengembangan yang dapat dilakukan terhadap tugas akhir ini di masa yang akan datang.

6.1. Kesimpulan

Dari hasil selama proses perancangan, implementasi, serta pengujian aplikasi absensi perkuliahan mahasiswa dengan pengenalan wajah, dapat diambil kesimpulan sebagai berikut:

1. Aplikasi berhasil dibangun menggunakan bahasa pemrograman C# dengan pengenalan wajah menggunakan *library* Emgu CV.
2. Metode *Eigenface* dipanggil dari *library* Emgu CV yang digunakan pada proses pengenalan wajah mahasiswa untuk mencatat status kehadiran.
3. Rancangan basis data yang digunakan untuk aplikasi ini berhasil dibangun dengan relasional manajemen basis data MySQL yang berisi tabel login, mahasiswa, foto, dosen, matakuliah, jadwal, kelas, beritaacara, dan kehadiran.
4. Aplikasi yang dibangun berhasil mencatat kehadiran mahasiswa pada suatu perkuliahan dengan pengenalan wajah pada kondisi pencahayaan dan resolusi yang sama dengan kondisi foto wajah yang disimpan dalam basis data.
5. Aplikasi yang dibangun berhasil menampilkan data absensi dan mengekstraknya ke dalam *file* Excel.
6. Aplikasi yang dibangun berhasil melakukan pencatatan data absensi dan perekapan perkuliahan mahasiswa secara otomatis sehingga proses pencatatan dan perekapan perkuliahan mahasiswa lebih efisien.

6.2. Saran

Berikut saran-saran untuk pengembangan dan perbaikan sistem di masa yang akan datang. Diantaranya adalah sebagai berikut:

1. Aplikasi dibangun dengan basis *mobile*, karena lebih bersifat *portable* dan selalu dalam genggaman mahasiswa.
2. Aplikasi diintegrasikan dengan *Geographic Information System* (GIS) untuk *indoor positioning system* dalam peta Teknik Informatika ITS, sehingga proses absensi bisa diketahui posisi mahasiswa pada saat melakukan absensi.

DAFTAR PUSTAKA

- [1] A. M. S. Laurent, *Understanding Open Source and Free Software Licensing*, California: O'Reilly Media, 2004.
- [2] M. B. H. a. P. A.-M. Cynthia Franklin, *The School Practitioner's Concise Companion to Preventing Dropout and Attendance Problems*, 2008.
- [3] A. Raj, *REAL TIME MULTIPLE FACE RECOGNITION SECURITY SYSTEM (RTM-FS)*, ResearchGate, 2013.
- [4] E. CV, "Emgu CV," 1 May 2016. [Online]. Available: http://www.emgu.com/wiki/index.php/Main_Page. [Accessed 20 May 2016].
- [5] H. a. Fatta, in *Rekayasa Sistem Pengenalan Wajah*, Yogyakarta, ANDI, 2009, p. 74.
- [6] J. R. d. S. a. P. Navarrete, "Eigenspace-based face recognition: a comparative study of different approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 35, no. 3, pp. 315 - 325, 2005.
- [7] M. A. T. a. A. P. Pentland, "Face recognition using eigenfaces," in *IEEE Computer Society Conference*, Maui, HI, 1991.
- [8] B. I. Y. B. Muhammad Rizki Muliawan, "Implementasi Pengenalan Wajah dengan Metode Eigenface pada Sistem Absensi," *Jurnal Coding*, vol. 03, no. 1, pp. 52-62, 2015.
- [9] A. Kadir, in *Konsep & Tuntunan Praktis Basis Data*, Yogyakarta, ANDI, 2002, p. 217.
- [10] Kusrini, in *Perancangan dan Pengelolaan Basis Data*, Yogyakarta, ANDI, 2007, p. 102.
- [11] R. Yanto, in *Manajemen Basis Data Menggunakan MySQL*, Yogyakarta, Deepublish, 2016, p. 243.

- [12] MySQL, "MySQL," Oracle Corporation, [Online]. Available: <http://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>. [Accessed 20 May 2016].
- [13] P. Wagh, R. Thakare, J. Chaudhari and S. Patil, "Attendance System based on Face Recogniiton using using Eigen face and PCA Algorithm," *IEEE*, pp. 303-308, 2015.
- [14] M. T. a. A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, p. 71–86, 1991.
- [15] M. A. Turk and A. P. Pentland, "Face Recognition Using Eigenfaces," *IEEE*, pp. 586-591, 1991.
- [16] Indra, "Sistem Pengenalan Wajah dengan Metode Eigenface untuk Absensi pada PT Florindo Lestari," in *Seminar Nasional Teknologi Informasi & Komunikasi Terapan 2012*, Semarang, 2012.
- [17] R. S. Putra, "Perancangan Aplikasi Absensi dengan Deteksi Wajah Menggunakan Metode EIGenface," *Pelita Informatika Budi Darma*, vol. IV, pp. 130-137, 2013.
- [18] S. Wardoyo, R. Wiryadinata and R. Sagita, "Sistem Presensi Berbasis Algoritma Eigenface dengan Metode Principal Component Analysis," *SETRUM*, vol. III, pp. 61-68, 2014.

BIODATA PENULIS



Penulis dilahirkan di Pati, 26 November 1991, merupakan anak ketiga dari tiga bersaudara. Penulis telah menempuh pendidikan formal di SD Negeri 231 Palembang, SMP Terpadu AKN Marzuqi Pati, SMK Telkom Terpadu AKN Marzuqi Pati, dan MA Abadiyah Pati. Selanjutnya penulis melanjutkan pendidikan sarjana di Jurusan Teknik Informatika - FTIf, Institut Teknologi Sepuluh Nopember (ITS) Surabaya dengan jalur kerjasama kemitraan ITS dengan

Kementerian Agama Republik Indonesia melalui Program Beasiswa Santri Berprestasi (PBSB) pada tahun 2011 dan terdaftar dengan NRP 5111100703.

Selama kuliah penulis aktif dalam organisasi ekstra kampus yaitu *Communtiy of Santri Scholars of Ministry of Religious Affairs* Institut Teknologi Sepuluh Nopember Surabaya (CSS MoRA ITS) dan Pergerakan Mahasiswa Islam Indonesia Komisariat Sepuluh Nopember Surabaya (PMII Sepuluh Nopember).

Penulis dalam menyelesaikan pendidikan S1 mengambil Rumpun Mata Kuliah Manajemen Informasi (RMK MI) dan memiliki ketertarikan di bidang Manajemen dan Proses Bisnis. Penulis dapat dihubungi melalui email : muha.yusuf11@gmail.com