



TUGAS AKHIR - KI091391

**RANCANG BANGUN ALAT PENGUKUR
TINGKAT KEPEKATAN ASAP BERDASARKAN
RINGELMANN SMOKE CHART PADA
PERANGKAT BERGERAK**

**M. CHOIRUL R. FAUZI
NRP 5110100182**

**Dosen Pembimbing
Daniel Oranova Siahaan, S.Kom., M.Sc., P.D.Eng.**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2014**



FINAL PROJECT - KI091391

**DESIGN AND IMPLEMENTATION OF SMOKE
OPACITY LEVEL MEASUREMENT INSTRUMENT
BASED ON RINGELMANN SMOKE CHART ON
MOBILE DEVICE**

**M. CHOIRUL R. FAUZI
NRP 5110100182**

**Advisor
Daniel Oranova Siahaan, S.Kom., M.Sc., P.D.Eng.**

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
SEPULUH NOPEMBER INSTITUTE OF TECHNOLOGY
SURABAYA
2014**

RANCANG BANGUN ALAT PENGUKUR TINGKAT KEPEKATAN ASAP BERDASARKAN RINGELMANN SMOKE CHART PADA PERANGKAT BERGERAK

Nama Mahasiswa : M. Choirul R. Fauzi
NRP : 5110 100 182
Jurusan : Teknik Informatika FTIf-ITS
Dosen Pembimbing I : Daniel Oranova Siahaan, S.Kom., M.Sc., P.D.Eng.

Abstrak

Pencemaran udara merupakan salah satu masalah yang ditimbulkan oleh pembakaran bahan bakar pada proses industri. Untuk mencegah pencemaran udara, dilakukan kegiatan pengamatan cerobong asap oleh Badan Lingkungan Hidup. Pengamat Badan Lingkungan Hidup akan mengukur tingkat kepekatan asap dari cerobong industri untuk mengetahui apakah asap yang dihasilkan oleh cerobong industri melebihi batas aman yang sudah ditetapkan atau tidak. Badan lingkungan hidup menggunakan skala Ringelmann Smoke Chart untuk menentukan tingkat kepekatan asap.

Selama ini pengukuran tingkat kepekatan asap dilakukan pengamat dengan melihat citra asap melalui teropong dan kemudian membandingkannya dengan Ringelmann Smoke Chart secara manual. Pengukuran dengan cara tersebut memiliki keterbatasan. Pengamatan tidak dapat dilakukan ketika penglihatan pengamat tidak dalam kondisi yang bagus. Selain itu data pengamatan masih dicatat dengan tulisan tangan pada kertas laporan pengamatan, sehingga tidak praktis dalam melihat perkembangan kepekatan asap dari cerobong. Untuk mengatasi permasalahan tersebut, dibuatlah aplikasi pengukuran tingkat

kepekatan asap dan pencatatan data pengamatan yang berjalan pada smartphone berbasis Windows Phone.

Aplikasi ini mengukur tingkat kepekatan asap dengan menghitung euclidean distance histogram RGB (Red, Green, Blue) dari citra asap dengan masing-masing tingkatan pada Ringelmann Smoke Chart. Tingkat kepekatan asap diambil berdasarkan tingkatan pada Ringelmann Smoke Chart yang memiliki euclidean distance terendah terhadap citra asap. Selain pengukuran tingkat kepekatan asap, aplikasi ini juga memiliki beberapa fitur lain, yaitu penyimpanan data pengamatan pada server, menampilkan riwayat pengukuran, dan benchmark nilai kepekatan asap.

Kata kunci: Pengukuran tingkat kepekatan asap, Ringelmann Smoke Chart, Histogram, Windows Phone.

DESIGN AND IMPLEMENTATION OF SMOKE OPACITY LEVEL MEASUREMENT INSTRUMENT BASED ON RINGELMANN SMOKE CHART ON MOBILE DEVICE

Student Name : M. Choirul R. Fauzi
NRP : 5110 100 182
Major : Teknik Informatika FTIf-ITS
Advisor I : Daniel Oranova Siahaan, S.Kom., M.Sc.,
P.D.Eng.

Abstract

Air pollution is one of the many problems caused by fuel combustion in industrial processes. To prevent air pollution, chimney observation activities conducted by the environmental agency. People from BLH (Badan Lingkungan Hidup) will measure the density of smoke from The industrial chimney to determine whether the smoke produced by The chimney exceeds safe limits that have been set or not. People from Badan Lingkungan Hidup is using the Ringelmann Smoke Chart scale to determine the opacity of the smoke produced by the chimney.

Until now, the smoke density measurements performed by the observers is to see the image of the smoke through binoculars and then compare it with the Ringelmann Smoke Chart manually. Measurements in this way has its limitations, the observations can not be performed when the visual capability of the observer is not in a good condition, and the observational data is still written by hand on a paper reporting the observations, so that observers has difficulty in seeing the development of the smoke from the chimney. To overcome these problems, an application of smoke opacity measurement and recording of observational data ,that runs on Windows Phone, is made.

This application measures the opacity of smoke by calculating the euclidean distance histogram of RGB (Red, Green, Blue) from the image of smoke with each level on the Ringelmann Smoke Chart. Smoke opacity levels chosen based on the Ringelmann Smoke Chart that has the lowest euclidean distance with the image of the smoke. In addition to the smoke opacity measurement, this application also has some other features such as, data storage of the observation data on the server, displays a history of measurement, and benchmarking the value of smoke opacity.

Keywords: Smoke opacity level measurement, Ringelmann Smoke Chart, Histogram, Windows Phone.

LEMBAR PENGESAHAN

RANCANG BANGUN ALAT PENGUKUR TINGKAT KEPEKATAN ASAP BERDASARKAN RINGELMANN SMOKE CHART PADA PERANGKAT BERGERAK

Tugas Akhir

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Bidang Studi Rekayasa Perangkat Lunak
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

M. CHOIRUL R. FAUZI

NRP. 5110 100 182

Disetujui oleh Dosen Pembimbing Tugas Akhir:

DANIEL ORANOVA SIAHAAN
M.Sc., P.D.Eng.
NIP: 197411232006041001



(pembimbing 1)

**SURABAYA
JULI, 2014**

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Puji syukur kepada Allah SWT atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan tugas akhir yang berjudul :

“Rancang Bangun Alat Pengukur Tingkat Kepekatan Asap Berdasarkan Ringelmann Smoke Chart pada Perangkat Bergerak”

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, tanpa mengurangi rasa hormat penulis ingin mengucapkan terima kasih sebesar-besarnya kepada:

1. Orang tua dan keluarga yang selalu memberikan dukungan penuh untuk menyelesaikan tugas akhir ini.
2. Bapak Daniel Oranova Siahaan selaku dosen pembimbing yang telah bersedia meluangkan waktu untuk memberikan petunjuk selama proses pengerjaan tugas akhir ini.
3. Bapak, Ibu dosen Jurusan Teknik Informatika ITS yang telah banyak memberikan ilmu dan bimbingan yang tak ternilai harganya bagi penulis.
4. Seluruh staf dan karyawan FTIf ITS yang banyak memberikan kelancaran administrasi akademik kepada penulis.
5. Rizki, Gegi, Ardian yang banyak membantu penulis dalam menyelesaikan tugas akhir ini.
6. Teman-teman dari penulis, Oi, Ichank, Zendra, Afid, Mis, Agil, Grezio, Sigit, Reza, Lely yang selalu memberikan semangat, dukungan, nasihat, mendengarkan keluh kesah penulis, dan menemani keseharian penulis

7. Teman-teman angkatan 2010 jurusan Teknik Informatika ITS yang telah ramah dan berbaik hati membantu penulis selama berada di Teknik Informatika.
8. Serta pihak-pihak lain yang tidak dapat disebutkan di sini 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 selanjutnya.

Surabaya, Juli 2014

M. Choirul R. Fauzi

DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
Abstrak	ix
Abstract	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR	xxi
DAFTAR TABEL	xxv
DAFTAR KODE SUMBER	xxvii
1. BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah.....	3
1.4. Tujuan Pembuatan Tugas Akhir.....	3
1.5. Metodologi Penulisan.....	3
a. Studi literatur.....	4
b. Analisis dan desain perangkat lunak	4
c. Implementasi perangkat lunak.....	4
d. Pengujian dan evaluasi	5
e. Penyusunan Buku Tugas Akhir.....	5
1.6. Sistematika Penulisan.....	6
2. BAB II TINJAUAN PUSTAKA.....	7
2.1. Ringelmann Smoke Chart	7
2.2. Windows Phone SDK 8.0.....	8

2.3. Windows Communication Foundation (WCF)	8
2.4. Internet Information Services (IIS)	8
2.5. <i>Image Retrieval Based on Color Histogram</i>	9
3. BAB III ANALISIS DAN PERANCANGAN PERANGKAT LUNAK.....	11
3.1. Analisis.....	11
3.1.1. Analisis Permasalahan.....	11
3.1.2. Deskripsi Umum Perangkat Lunak	12
3.1.3. Spesifikasi Kebutuhan Perangkat Lunak.....	12
3.1.3.1 Analisa Aktor	12
3.1.3.2 Kasus Penggunaan.....	13
3.1.3.3 Kebutuhan Fungsional.....	22
3.2. Perancangan	24
3.2.1. Lingkungan Perancangan Perangkat Lunak	24
3.2.2 Arsitektur Sistem.....	25
3.2.3 Perancangan Aplikasi <i>Web Service</i>	26
3.2.3.1 Perancangan Diagram Kelas.....	26
3.2.4 Perancangan Aplikasi Perangkat Bergerak.....	27
3.2.4.1 Perancangan Diagram Kelas.....	27
3.2.4.2 Perancangan Antarmuka Pengguna	29
3.2.4.2.2 Rancangan Antarmuka Halaman Register... 31	
3.2.4.3 Perancangan Proses Pengukuran Kecepatan Asap	58
4. BAB IV IMPLEMENTASI PERANGKAT LUNAK.....	61
4.1. Lingkungan Implementasi	61
4.1.1. Lingkungan Implementasi Perangkat Keras.....	61

4.1.2.	Lingkungan Implementasi Perangkat Lunak.....	61
4.2.	Implementasi Antarmuka Pengguna.....	61
4.3.1	Implementasi Antarmuka Halaman Login	62
4.3.2	Implementasi Antarmuka Halaman Register.....	62
4.3.3	Implementasi Antarmuka Halaman Pilihan Menu 63	
4.3.4	Implementasi Antarmuka Halaman Ambil Gambar 64	
4.3.5	Implementasi Antarmuka Halaman Hasil Pengukuran.....	65
4.3.6	Implementasi Antarmuka Halaman Pilih Perusahaan pada Fitur Pengukuran	66
4.3.7	Implementasi Antarmuka Halaman Tambah Perusahaan.....	67
4.3.8	Implementasi Antarmuka Halaman Peta Cerobong 68	
4.3.9	Implementasi Antarmuka <i>MessageBox</i> Pilih Cerobong	69
4.3.10	Implementasi Antarmuka <i>MessageBox</i> Tambah Cerobong	70
4.3.11	Implementasi Antarmuka Halaman Formulir Pengamatan	71
4.3.12	Implementasi Antarmuka Halaman Rekaman Data 72	
4.3.13	Implementasi Antarmuka Halaman Pilih Perusahaan pada Fitur Riwayat	73
4.3.14	Implementasi Antarmuka Halaman Riwayat.....	74
4.3.15	Implementasi Antarmuka Halaman Detail Pengamatan	75

4.3.16	Implementasi Antarmuka Halaman Grafik Riwayat	76
4.3.17	Implementasi Antarmuka Halaman Benchmark..	77
4.3.	Implementasi Kasus Penggunaan	78
4.4.1	Implementasi Kasus Penggunaan Mengukur Tingkat Kepekatan Asap	78
4.4.2	Implementasi Kasus Penggunaan Mengunggah Rekaman Data ke <i>Server</i>	81
4.4.3	Implementasi Kasus Penggunaan Menampilkan Riwayat Pengukuran.....	84
4.4.4	Implementasi Kasus Penggunaan Menampilkan <i>Benchmark</i> Nilai Kepekatan Asap.....	85
5.	BAB V PENGUJIAN DAN EVALUASI	89
5.1	Lingkungan Pengujian.....	89
5.2	Skenario Pengujian.....	90
5.2.1	Pengujian Kasus Penggunaan.....	90
5.2.1.1	Pengujian Fitur Mengukur Tingkat Kepekatan Asap	90
5.2.1.2	Pengujian Fitur Mengunggah Rekaman Data ke <i>Server</i>	96
5.2.1.3	Pengujian Fitur Menampilkan Riwayat Pengukuran.....	98
5.2.1.4	Pengujian Fitur Menampilkan <i>Benchmark</i> Nilai Kepekatan Asap.....	100
5.2.2	Pengujian Kegunaan & Kemudahan Aplikasi ...	102
5.3	Evaluasi Pengujian	104
5.3.1	Evaluasi Pengujian Fungsionalitas	104
5.3.2	Evaluasi Pengujian Kegunaan & Kemudahan.....	105

5.3.3	Evaluasi Hasil Pengukuran Tingkat Kepekatan Asap	106
6	BAB VI PENUTUP.....	109
6.1	Kesimpulan.....	109
6.2	Saran.....	109
	DAFTAR PUSTAKA	111
7	LAMPIRAN A. Diagram Aktivitas Kasus Penggunaan ...	113
8	LAMPIRAN B. Diagram Urutan Kasus Penggunaan	115
9	LAMPIRAN C. Diagram Kelas	119
10	LAMPIRAN D. Hasil Kuesioner	121
11	LAMPIRAN E. Hasil Pengukuran	123
	BIODATA PENULIS	127

DAFTAR GAMBAR

Gambar 1.1 Digital Ringelmann Smoke Opacity Meter	2
Gambar 2.1 Ringelmann Smoke Chart.....	7
Gambar 3.1 Diagram Kasus Penggunaan.....	13
Gambar 3.2 Diagram Aktivitas Menunggah Rekaman Data ke <i>Server</i>	18
Gambar 3.3 Diagram Aktivitas Menampilkan Riwayat Pengukuran	20
Gambar 3.4 Diagram Aktivitas Menampilkan <i>Benchmark</i> Nilai Kepekatan Asap.....	22
Gambar 3.5 Diagram Urutan Menampilkan <i>Benchmark</i> Nilai Kepekatan Asap.....	23
Gambar 3.6 Arsitektur Sistem.....	25
Gambar 3.7 Diagram Kelas Aplikasi <i>Web Service</i>	26
Gambar 3.8 Rancangan Antarmuka Halaman Login	30
Gambar 3.9 Rancangan Antarmuka Halaman Register.....	31
Gambar 3.10 Rancangan Antarmuka Halaman Pilihan Menu	33
Gambar 3.11 Rancangan Antarmuka Halaman Ambil Gambar ..	35
Gambar 3.12 Rancangan Antarmuka Halaman Hasil Pengukuran Kepekatan.....	37
Gambar 3.13 Rancangan Antarmuka Halaman Pilih Perusahaan pada Fitur Pengukuran.....	39
Gambar 3.14 Rancangan Antarmuka Halaman Tambah Perusahaan	41
Gambar 3.15 Rancangan Antarmuka Halaman Peta Cerobong ..	43
Gambar 3.16 Rancangan Antarmuka <i>MessageBox</i> Pilih Cerobong	45
Gambar 3.17 Rancangan Antarmuka <i>MessageBox</i> Tambah Cerobong	47
Gambar 3.18 Rancangan Antarmuka Halaman Formulir Pengamatan	49
Gambar 3.19 Rancangan Antarmuka Halaman Rekaman Data ..	50

Gambar 3.20 Rancangan Antarmuka Halaman Pilih Perusahaan pada Fitur Riwayat	52
Gambar 3.21 Rancangan Antarmuka Halaman Riwayat.....	53
Gambar 3.22 Rancangan Antarmuka Halaman Detail Pengamatan	55
Gambar 3.23 Rancangan Antarmuka Halaman Grafik Riwayat .	56
Gambar 3.24 Rancangan Antarmuka Halaman Benchmark.....	57
Gambar 3.25 Antarmuka Kamera	59
Gambar 4.1 Implementasi Halaman <i>Login</i>	62
Gambar 4.2 Implementasi Halaman Register.....	63
Gambar 4.3 Implementasi Halaman Pilihan Menu	64
Gambar 4.4 Implementasi Halaman Ambil Gambar	65
Gambar 4.5 Implementasi Halaman Hasil Pengukuran	66
Gambar 4.6 Implementasi Halaman Pilih Perusahaan pada Fitur Pengukuran.....	67
Gambar 4.7 Implementasi Halaman Tambah Perusahaan	68
Gambar 4.8 Implementasi Halaman Peta Cerobong	69
Gambar 4.9 Implementasi <i>MessageBox</i> Pilih Cerobong	70
Gambar 4.10 Implementasi <i>MessageBox</i> Tambah Cerobong	71
Gambar 4.11 Implementasi Halaman Formulir Pengamatan	72
Gambar 4.12 Implementasi Halaman Rekaman Data	73
Gambar 4.13 Implementasi Halaman Pilih Perusahaan pada Fitur Riwayat.....	74
Gambar 4.14 Implementasi Halaman Riwayat.....	75
Gambar 4.15 Implementasi Halaman Detail Pengamatan	76
Gambar 4.16 Implementasi Halaman Grafik Riwayat	77
Gambar 4.17 Implementasi Halaman <i>Benchmark</i>	78
Gambar 5.1 Hasil Pengujian Fitur Mengukur Tingkat Kepekatan Asap Untuk Skenario Pertama	93
Gambar 5.2 Hasil Pengujian Fitur Mengukur Tingkat Kepekatan Asap Untuk Skenario Kedua	94
Gambar 5.3 Hasil Pengujian Fitur Mengukur Tingkat Kepekatan Asap Untuk Skenario Ketiga.....	96
Gambar 5.4 Hasil Pengujian Fitur Mengunggah Rekaman Data ke <i>Server</i>	98

Gambar 5.5 Hasil Pengujian Fitur Menampilkan Riwayat Pengukuran.....	100
Gambar 5.6 Hasil Pengujian Fitur Menampilkan <i>Benchmark</i> Nilai Kepekatan Asap.....	102
Gambar 7.1 Diagram Aktivitas Mengukur Tingkat Kepekatan Asap	113
Gambar 8.1 Diagram Urutan Mengukur Tingkat Kepekatan Asap	115
Gambar 8.2 Diagram Urutan Mengunggah Rekaman Data ke <i>Server</i>	116
Gambar 8.3 Diagram Urutan Menampilkan Riwayat Pengukuran	117

DAFTAR KODE SUMBER

Kode Sumber 4.1 Sintaks untuk Mengukur Tingkat Kepekatan Asap.....	81
Kode Sumber 4.2 Sintaks untuk Mengunggah Rekaman Data ke <i>Server</i>	84
Kode Sumber 4.3 Sintaks untuk Menampilkan Riwayat Pengukuran.....	85
Kode Sumber 4.4 Sintaks untuk Menampilkan Benchmark Nilai Kepekatan Asap.....	87

DAFTAR TABEL

Tabel 3.1 Daftar Kode Diagram Kasus Penggunaan.....	13
Tabel 3.2 Spesifikasi Kasus Penggunaan Mengukur Tingkat Kepekatan Asap.....	14
Tabel 3.3 Spesifikasi Kasus Penggunaan Mengunggah Rekaman Data ke <i>Server</i>	17
Tabel 3.4 Spesifikasi Kasus Penggunaan Menampilkan Riwayat Pengukuran.....	19
Tabel 3.5 Spesifikasi Kasus Penggunaan Menampilkan <i>Benchmark</i> Nilai Kepekatan Asap.....	21
Tabel 3.6 Daftar Kebutuhan Fungsional Perangkat Lunak	23
Tabel 3.7 Lingkungan Perancangan Perangkat Lunak.....	24
Tabel 3.8 Spesifikasi Atribut Antarmuka Halaman Login	30
Tabel 3.9 Spesifikasi Atribut Antarmuka Halaman Register	31
Tabel 3.10 Spesifikasi Atribut Antarmuka Halaman Pilihan Menu	33
Tabel 3.11 Spesifikasi Atribut Antarmuka Halaman Ambil Gambar	35
Tabel 3.12 Spesifikasi Atribut Antarmuka Halaman Hasil Pengukuran Kepekatan.....	37
Tabel 3.13 Spesifikasi Atribut Antarmuka Halaman Pilih Perusahaan pada Fitur Pengukuran	39
Tabel 3.14 Spesifikasi Atribut Antarmuka Halaman Tambah Perusahaan.....	41
Tabel 3.15 Spesifikasi Atribut Antarmuka Halaman Peta Cerobong	43
Tabel 3.16 Spesifikasi Atribut Antarmuka <i>MessageBox</i> Pilih Cerobong	45
Tabel 3.17 Spesifikasi Atribut Antarmuka <i>MessageBox</i> Tambah Cerobong	47
Tabel 3.18 Spesifikasi Atribut Antarmuka Halaman Formulir Pengamatan	49

Tabel 3.19 Spesifikasi Atribut Antarmuka Halaman Rekaman Data	50
Tabel 3.20 Spesifikasi Atribut Antarmuka Halaman Pilih Perusahaan pada Fitur Riwayat	52
Tabel 3.21 Spesifikasi Atribut Antarmuka Halaman Riwayat	53
Tabel 3.22 Spesifikasi Atribut Antarmuka Halaman Detail Pengamatan	55
Tabel 3.23 Spesifikasi Atribut Antarmuka Halaman Grafik Riwayat.....	56
Tabel 3.24 Spesifikasi Atribut Antarmuka Halaman Benchmark	57
Tabel 5.1 Pengujian Mengukur Tingkat Kepekatan Asap Skenario Pertama.....	91
Tabel 5.2 Pengujian Mengukur Tingkat Kepekatan Asap Skenario Kedua	92
Tabel 5.3 Pengujian Mengukur Tingkat Kepekatan Asap Skenario Ketiga	94
Tabel 5.4 Skenario Pengujian Mengunggah Rekaman Data ke <i>Server</i>	97
Tabel 5.5 Skenario Pengujian Menampilkan Riwayat Pengukuran	99
Tabel 5.6 Skenario Pengujian Menampilkan <i>Benchmark</i> Nilai Kepekatan Asap.....	100
Tabel 5.7 Kuesioner Pendapat Pengguna	103
Tabel 5.8 Rangkuman Hasil Pengujian	104
Tabel 5.9 Hasil Pengujian Kegunaan & Kemudahan	105
Tabel 5.10 Akurasi Pengukuran Oleh Aplikasi	107
Tabel 11.1 Perbandingan Hasil Pengukuran Oleh Aplikasi dengan Hasil Pengukuran Oleh Pengamat Secara Manual	123

BAB I

PENDAHULUAN

Pada bagian ini akan dijelaskan mengenai latar belakang pembuatan tugas akhir ini, permasalahan yang dihadapi, batasan masalah, tujuan dan manfaat yang ingin dicapai dalam pengerjaan tugas akhir ini, metodologi pembuatan tugas akhir, dan sistematika penulisan yang digunakan dalam pembuatan tugas akhir ini.

1.1. Latar Belakang

Seiring dengan meningkatnya kegiatan industri, khususnya industri yang melakukan pembakaran bahan bakar, maka pencemaran udara pun semakin meningkat. Hal tersebut mendorong pemerintah untuk membuat aturan mengenai pengendalian pencemaran udara yang tertuang pada Keputusan Kepala Bapedal No. 205 Tahun 1996 [1]. Dari peraturan tersebut maka dilakukanlah pengukuran tingkat kepekatan asap oleh Badan Lingkungan Hidup berdasarkan Ringelmann Smoke Chart.

Untuk mengetahui tingkat kepekatan asap sulit dilakukan dengan tanpa alat bantu. Salah satu alat bantu yang memudahkan untuk mengetahui tingkat kepekatan dari asap adalah Digital Ringelmann Smoke Opacity Meter. Digital Ringelmann Smoke Opacity Meter yang ditunjukkan oleh Gambar 1.1 merupakan sebuah alat yang terdiri dari teropong, indikator tingkat kepekatan asap dan kamera. Indikator tingkat kepekatan asap terdiri dari 5 tingkatan sesuai dengan Ringelmann Smoke Chart. Sedangkan kamera digunakan untuk mendokumentasikan citra asap yang sedang diukur tingkat kepekataannya. Cara menentukan tingkat kepekatan asap adalah pengamat melihat asap melalui teropong dan membandingkan asap tersebut sesuai dengan skala Ringlemann.

Selama ini pengukuran tingkat kepekatan asap masih dilakukan secara manual. Yaitu pengamat melihat asap melalui teropong dan membandingkannya dengan Ringelmann Smoke Chart. Kamera

hanya digunakan untuk mendokumentasikan kegiatan pengamatan asap. Pengukuran dengan cara tersebut memiliki keterbatasan. Pengamatan tidak dapat dilakukan ketika penglihatan pengamat tidak dalam kondisi yang bagus. Selain itu data pengamatan masih dicatat dengan tulisan tangan pada kertas laporan pengamatan, sehingga tidak praktis dalam melihat perkembangan kepekatan asap dari cerobong.



Gambar 1.1 Digital Ringelmann Smoke Opacity Meter

Untuk mengatasi permasalahan di atas, maka akan dibuat sistem untuk melakukan pengukuran tingkat kepekatan asap secara otomatis dan juga pencatatan data pengamatan. Pengukuran dengan sistem yang akan dibangun ini dilakukan dengan cara memotret asap menggunakan perangkat bergerak (*smartphone*) yang ditempelkan pada teropong. Kemudian citra asap tersebut diproses pada perangkat bergerak. Perangkat bergerak dipilih karena pengukuran tingkat kepekatan asap dilakukan di lapangan, sehingga pengukuran dan penyimpanan data pengukurannya bisa dilakukan dengan cepat.

1.2. Rumusan Masalah

Beberapa rumusan masalah dalam perancangan dan pembuatan perangkat lunak ini adalah :

1. Bagaimana cara mengukur tingkat kepekatan asap berdasarkan Ringelmann Smoke Chart.
2. Bagaimana cara membangun aplikasi *client* yang dapat mengukur tingkat kepekatan asap dan aplikasi *server* yang dapat melakukan penyimpanan data pengukuran.

1.3. Batasan Masalah

Batasan masalah dalam perancangan dan pembuatan perangkat lunak ini adalah :

1. Aplikasi *server* dibangun menggunakan kerangka kerja Windows Communication Foundation (WCF) yang berjalan pada *web server* berbasis pada Internet Information Services (IIS).
2. Aplikasi *client* dibangun menggunakan Windows Phone SDK 8.0 dengan menggunakan bahasa pemrograman C# dengan kakas bantu Visual Studio 2012.
3. Aplikasi *client* yang dibangun adalah aplikasi yang berjalan pada perangkat dengan sistem operasi Windows Phone dengan versi 8.0.

1.4. Tujuan Pembuatan Tugas Akhir

Tujuan pembuatan Tugas Akhir adalah:

1. Membangun aplikasi untuk mengukur tingkat kepekatan asap.
2. Membangun aplikasi yang bisa melakukan penyimpanan data pengukuran tingkat kepekatan asap.

1.5. Metodologi Penulisan

Metodologi penulisan TA ini memiliki beberapa tahapan sebagai berikut.

a. Studi literatur

Pada tahap ini dilakukan pencarian informasi dan studi mengenai referensi yang ada, antara lain :

1. Metode pengukuran tingkat kepekatan asap berdasarkan Ringelmann Smoke Chart.
2. Pembuatan aplikasi pada perangkat bergerak menggunakan kaskas Visual Studio 2012 dan pustaka Windows Phone SDK 8.0.
3. Pembuatan *web service* menggunakan kerangka kerja Windows Communication Foundation (WCF).
4. Penyimpanan data pada basis data MySQL dan basis data lokal pada perangkat bergerak.

b. Analisis dan desain perangkat lunak

Pada tahap ini dilakukan analisis dan pendefinisian kebutuhan sistem untuk masalah yang sedang dihadapi. Selanjutnya, dilakukan perancangan sistem dengan beberapa tahap sebagai berikut.

1. Analisis aktor yang terlibat di dalam sistem.
2. Perancangan proses aplikasi.
3. Perancangan antar muka sistem.
4. Perancangan diagram kelas sistem.

c. Implementasi perangkat lunak

Pada tahap ini dilakukan pembuatan elemen perangkat lunak. Sistem yang dibuat berpedoman pada rancangan yang telah dibuat pada proses perancangan dan analisis sistem.

Perincian tahap ini adalah sebagai berikut.

1. Implementasi pengukuran tingkat kepekatan asap berdasarkan foto yang diambil menggunakan kamera perangkat bergerak.

2. Implementasi penyimpanan data pengukuran tingkat kepekatan asap pada *web service* atau pada basis data lokal.

d. Pengujian dan evaluasi

Pengujian perangkat lunak akan dilakukan secara *black-box*, yaitu pengujian yang menguji fungsionalitas perangkat lunak apakah sudah sesuai dengan yang diharapkan [2]. Sedangkan evaluasi hasil pengukuran tingkat kepekatan asap akan dilakukan dengan membandingkan hasil pengukuran oleh aplikasi dengan hasil pengukuran secara manual.

e. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
 - 1.1. Latar Belakang
 - 1.2. Rumusan Masalah
 - 1.3. Batasan Tugas Akhir
 - 1.4. Tujuan
 - 1.5. Metodologi
 - 1.6. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

1.6. Sistematika Penulisan

Buku TA ini disusun dengan sistematika penulisan sebagai berikut.

BAB I. PENDAHULUAN

Bab ini berisi latar belakang, permasalahan, tujuan, batasan permasalahan, metodologi, dan sistematika penulisan.

BAB II. TINJAUAN PUSTAKA

Bab ini berisi dasar ilmu yang mendukung pembahasan tugas akhir ini.

BAB III. PERANCANGAN PERANGKAT LUNAK

Bab ini berisi tentang desain sistem, *flowchart*, dan desain antarmuka yang akan dibuat.

BAB IV. IMPLEMENTASI PERANGKAT LUNAK

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa *pseudocode* dari fungsi utama dan *screenshot* aplikasi.

BAB V. PENGUJIAN DAN EVALUASI

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian fungsionalitas dan pengujian performa dalam beberapa skenario.

BAB VI. PENUTUP

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

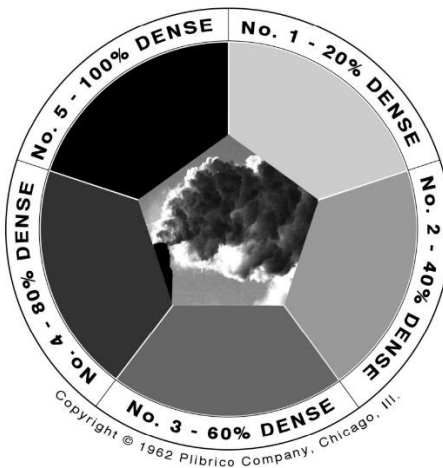
BAB II

TINJAUAN PUSTAKA

Pada bab ini akan dibahas mengenai teori-teori yang menjadi dasar dari pembuatan tugas akhir. Teori-teori tersebut meliputi Ringelmann Smoke Chart, Windows Phone SDK 8.0, Windows Communication Foundation (WCF), Internet Information Service (IIS), *Image Retrieval Based on Color Histogram*.

2.1. Ringelmann Smoke Chart

Ringelmann Smoke Chart merupakan sebuah kartu yang digunakan untuk menentukan tingkat kepekatan asap. Kartu Ringelmann terdiri dari 5 buah tingkatan yang diberi nomor mulai 1-5 sesuai pada Gambar 2.1. Kartu-kartu tersebut berwarna putih sampai hitam pekat. Selanjutnya kartu-kartu tersebut juga diberi skala dengan % kepekatan.



Gambar 2.1 Ringelmann Smoke Chart

Cara menggunakan kartu ini sederhana saja yaitu dengan melihat asap yang keluar dari cerobong serta menyamakan warna asap dengan salah satu warna dari kartu Ringlemann sehingga tingkat kepekatan dapat diketahui. Asap yang masih dianggap aman terhadap lingkungan adalah asap dengan kepekatan dibawah 40 % [3].

2.2. Windows Phone SDK 8.0

Windows Phone SDK 8.0 merupakan lingkungan pengembangan yang digunakan untuk membangun aplikasi dan permainan untuk Windows Phone 8.0 dan Windows Phone 7.5. Windows Phone SDK menyediakan Visual Studio Express edisi 2012 untuk Windows Phone yang berdiri sendiri atau sebagai sebuah *add-in* untuk Visual Studio 2012 edisi Professional, Premium atau Ultimate. Dengan SDK, pengembang dapat menggunakan keterampilan pemrograman dan kode yang telah ada untuk membangun aplikasi. Selain itu SDK mencakup beberapa *emulator* dan kakas-kakas tambahan untuk pengujian aplikasi Windows Phone sesuai kondisi dunia nyata [4].

2.3. Windows Communication Foundation (WCF)

Windows Communication Foundation (WCF) adalah suatu kerangka kerja untuk membangun aplikasi berorientasi layanan. Menggunakan WCF, pengembang dapat mengirim data sebagai pesan *asynchronous* dari satu *endpoint* layanan lain.

Sebuah *endpoint* layanan dapat menjadi bagian dari layanan tersedia yang diselenggarakan oleh IIS atau dapat menjadi layanan *host* dalam sebuah aplikasi. Sebuah *endpoint* dapat menjadi klien dari layanan yang meminta data dari *endpoint* layanan. Pesan dapat sesederhana satu karakter atau kata dikirim sebagai XML, atau sekomples aliran data biner [5].

2.4. Internet Information Services (IIS)

Internet Information Services (IIS, sebelumnya Internet Information Server) adalah sebuah *web server* yang dibuat oleh

Microsoft untuk digunakan oleh keluarga Windows NT. IIS mendukung HTTP, HTTPS, FTP, FTPS, SMTP dan NNTP. IIS telah menjadi bagian integral dari keluarga Windows NT sejak Windows NT 4.0, meskipun mungkin absen dari beberapa edisi, misalnya Windows XP Home Edition. IIS tidak diaktifkan secara *default* ketika Windows di-*install*. IIS Manager diakses melalui Microsoft Management Console atau Administrative Tools di Control Panel [6].

2.5. Image Retrieval Based on Color Histogram

Adalah sebuah metode yang digunakan untuk mencari kemiripan antar gambar berdasarkan histogram warna berdasarkan *paper* yang ditulis oleh Jain and Vailaya [7]. Berikut ini alur dari metode yang digunakan.

1. Membaca gambar pada *database* dan mengekstrak informasi RGB dari masing-masing *pixel* pada gambar.
2. Membuat histogram pada *array* dengan panjang 48 yang dinormalisasi untuk masing-masing komponen pada RGB dari setiap gambar dari *database*. Dengan demikian, masing-masing gambar akan memiliki 3 histogram yang terkait dengan itu.
3. Ekstrak informasi RGB pada masing-masing *pixel* pada gambar.
4. Menghitung *euclidean distance* antara gambar-gambar yang dibandingkan.
5. Urutkan gambar dalam *database* dalam urutan menaik berdasarkan *euclidean distance*. Semakin kecil *euclidean distance*, maka semakin besar kemiripan dari dua gambar yang dibandingkan.

Pada *paper* tersebut dilakukan serangkaian pengujian dan didapatkan akurasi sebesar 66%.

(Halaman ini sengaja dikosongkan)

BAB III

ANALISIS DAN PERANCANGAN PERANGKAT LUNAK

Pada bab ini dibahas mengenai analisis dan perancangan perangkat lunak dari sistem yang akan dibangun pada tugas akhir ini. Bagian pertama dari bab ini membahas analisis dan kebutuhan sistem dan bagian kedua membahas perancangan sistem bagian kedua membahas perancangan sistem berdasarkan hasil analisis dari bagian pertama.

3.1. Analisis

Tahap ini dibagi menjadi beberapa bagian, antara lain analisa permasalahan, deskripsi umum sistem, dan spesifikasi kebutuhan perangkat lunak.

3.1.1. Analisis Permasalahan

Permasalahan utama yang diangkat dalam pembuatan tugas akhir ini adalah bagaimana mengukur tingkat kepekatan asap secara akurat. Untuk mengukur tingkat kepekatan asap dilakukan dengan menggunakan pendekatan *euclidean distance* histogram RGB antara citra asap dengan masing-masing tingkatan pada Ringelmann Smoke Chart. Tingkat kepekatan didapatkan dari tingkatan Ringelmann Smoke Chart yang memiliki *euclidean distance* terendah. Permasalahan kedua yaitu bagaimana menyimpan data pengamatan di *server*. Apabila koneksi ke *server* terputus, data pengamatan akan disimpan di basis data lokal. Penyimpanan data ini dimaksudkan agar pengguna bisa melihat riwayat pengamatan asap dan mengetahui *benchmark* nilai kepekatan asap. Permasalahan ketiga adalah bagaimana menampilkan riwayat pengamatan asap yang didapatkan dari *server*. Permasalahan keempat adalah bagaimana melakukan *benchmarking* terhadap nilai kepekatan yang tersimpan di *server*.

3.1.2. Deskripsi Umum Perangkat Lunak

Aplikasi yang dibuat merupakan aplikasi berbasis Windows Phone untuk melakukan pengukuran tingkat kepekatan asap industri. Aplikasi ini mengambil citra asap melalui kamera *smartphone* dan langsung memprosesnya setelah citra asap didapatkan. Aplikasi ini memanfaatkan *web service* untuk melakukan penyimpanan data.

Aplikasi ini terdiri dari beberapa fitur, yaitu:

1. Pengguna dapat melakukan pengukuran tingkat kepekatan asap dan menyimpan data pengamatannya pada *server*.
2. Data pengamatan akan tersimpan pada basis data lokal apabila koneksi internet terputus ketika proses unggah data pengamatan ke *server*.
3. Pengguna dapat melihat riwayat pengamatan yang tersimpan pada *server*.
4. Pengguna dapat melihat *benchmarking* nilai kepekatan asap dari keseluruhan data yang tersimpan pada *server*.

3.1.3. Spesifikasi Kebutuhan Perangkat Lunak

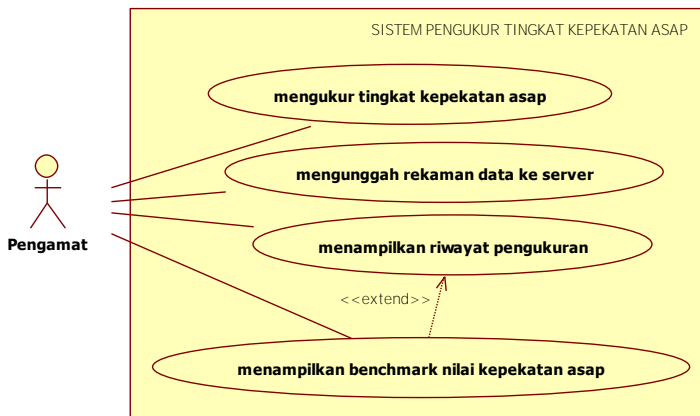
Bagian ini berisi kebutuhan perangkat lunak yang terdiri dari aktor dan kasus penggunaan yang diuraikan dalam bentuk diagram kasus penggunaan, kelas analisis, diagram urutan dan diagram aktivitas. Setiap diagram menjelaskan perilaku dari perangkat lunak. Selain itu juga terdapat kebutuhan perangkat lunak dalam sistem yang meliputi kebutuhan fungsional dan kebutuhan non fungsional. Pada bab ini akan dijelaskan spesifikasi pada masing-masing kebutuhan fungsional dan kebutuhan non fungsional.

3.1.3.1 Analisa Aktor

Aktor mendefinisikan entitas-entitas yang terlibat dan berinteraksi langsung dengan sistem. Entitas ini bisa berupa manusia maupun sistem atau perangkat lunak yang lain. Aktor yang terdapat pada sistem ini hanya memiliki sebuah peran yaitu sebagai pengguna. Pengguna perangkat ini adalah pengamat tingkat kepekatan asap dari badan lingkungan hidup atau instansi terkait.

3.1.3.2 Kasus Penggunaan

Berdasarkan analisis spesifikasi kebutuhan fungsional dan analisis aktor dari sistem dibuat kasus penggunaan sistem. Kasus-kasus penggunaan dalam sistem ini akan dijelaskan secara rinci pada subbab ini. Kasus penggunaan digambarkan dalam sebuah diagram kasus penggunaan. Diagram kasus penggunaan dapat dilihat pada Gambar 3.1. Tabel 3.1 berisi penjelasan dari setiap kasus penggunaan.



Gambar 3.1 Diagram Kasus Penggunaan

Tabel 3.1 Daftar Kode Diagram Kasus Penggunaan

Kode Kasus Penggunaan	Nama
UC-0001	Mengukur tingkat kepekatan asap
UC-0002	Mengunggah rekaman data ke <i>server</i>
UC-0003	Menampilkan riwayat pengukuran

UC-0004	Menampilkan <i>benchmark</i> nilai kepekatan asap
---------	---

3.1.3.2.1 Mengukur Tingkat Kepekatan Asap

Pada kasus penggunaan ini, sistem menerima *input* berupa perintah untuk mengukur tingkat kepekatan asap. Kemudian sistem akan menampilkan antarmuka kamera. Setelah pengguna menekan tombol untuk mengambil gambar, sistem akan menampilkan antarmuka yang berisi hasil pengukuran. Setelah itu pengguna dapat mengisi data pengamatan yang akan diunggah ke *server*. Apabila koneksi *internet* tidak tersambung, data akan disimpan di basis data lokal. Spesifikasi kasus penggunaan ini dapat dilihat pada Tabel 3.2. Diagram aktivitas dan diagram urutan dari kasus penggunaan ini bisa dilihat pada Lampiran A Gambar 7.1 dan Lampiran B Gambar 8.1.

Tabel 3.2 Spesifikasi Kasus Penggunaan Mengukur Tingkat Kepekatan Asap

Nama	Mengukur tingkat kepekatan asap
Kode	UC-0001
Deskripsi	Mengukur tingkat kepekatan asap berdasarkan foto yang diambil pengguna.
Tipe	Fungsional
Pemicu	Pengguna memilih menu untuk mengukur tingkat kepekatan asap.
Aktor	Pengguna
Kondisi Awal	-

<p>Aliran: - Kejadian Normal</p>	<ol style="list-style-type: none"> 1. Pengguna memilih menu untuk mengukur tingkat kepekatan asap. 2. Sistem menampilkan antarmuka kamera. 3. Pengguna menekan tombol untuk mengambil gambar. 4. Sistem menampilkan antarmuka yang berisi hasil pengukuran tingkat kepekatan asap. 5. Pengguna menekan tombol untuk mengisi data pengamatan. 6. Sistem menampilkan daftar perusahaan. 7. Pengguna memilih perusahaan yang cerobongnya diamati. 8. Sistem menampilkan peta lokasi cerobong. 9. Pengguna memilih cerobong yang diamati. 10. Sistem menampilkan antarmuka yang berisi borang pengisian data pengamatan. 11. Pengguna mengisi borang pengisian data laporan pengamatan. 12. Pengguna menekan tombol untuk menyimpan data di <i>server</i>. 13. Sistem menyimpan data di <i>server</i>. 14. Sistem menampilkan pesan bahwa data berhasil disimpan di <i>server</i>.
<p>- Kejadian Alternatif</p>	<ol style="list-style-type: none"> 7.A. Perusahaan yang cerobongnya diamati tidak terdapat dalam daftar. 7.A.1 Sistem menampilkan antarmuka yang berisi borang pengisian data perusahaan baru. 7.A.2 Pengguna menambahkan data perusahaan baru. 7.A.3 Aliran berlanjut ke alur normal langkah nomor 8.

	<p>9.A. Cerobong yang diamati tidak terdapat dalam daftar.</p> <p>9.A.1 Pengguna menandai lokasi cerobong baru pada peta.</p> <p>9.A.2 Pengguna memilih cerobong baru yang ditambahkan.</p> <p>9.A.3 Sistem menampilkan borang pengisian data cerobong baru.</p> <p>9.A.4 Pengguna mengisi borang pengisian data cerobong baru.</p> <p>9.A.5 Aliran berlanjut ke alur normal langkah nomor 10.</p> <p>13.A. Koneksi <i>internet</i> tidak tersambung.</p> <p>13.A.1 Sistem menyimpan data pengukuran pada basis data lokal.</p> <p>13.A.2 Sistem menampilkan pesan bahwa data disimpan pada basis data lokal.</p> <p>13.A.3 Aliran selesai.</p>
Kondisi Akhir	Sistem menampilkan pesan bahwa data telah berhasil disimpan di <i>server</i> .

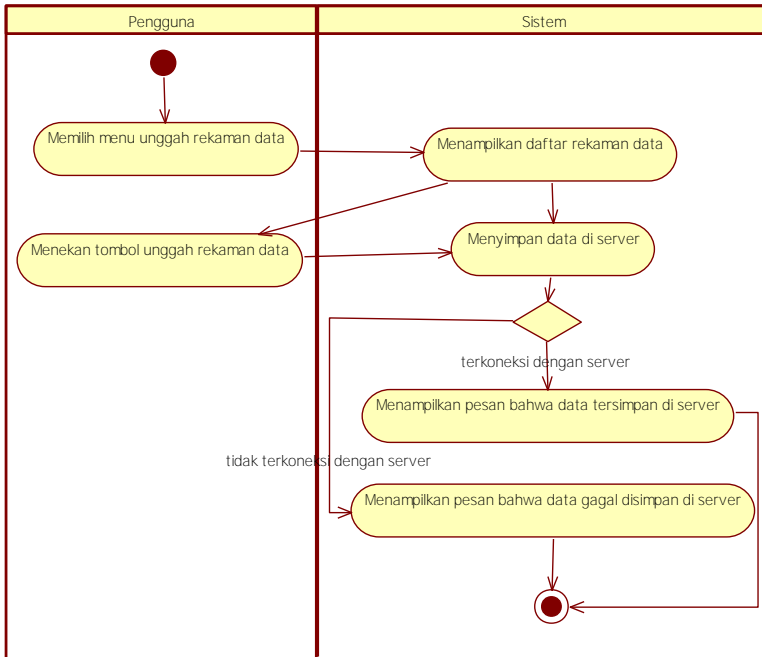
3.1.3.2.2 Mengunggah Rekaman Data ke *Server*

Rekaman data merupakan data laporan yang tersimpan pada basis data lokal apabila pengguna menyimpan data tersebut dalam kondisi koneksi *internet* tidak tersambung. Pengguna dapat mengunggah data pengamatan tersebut ke *server*. Spesifikasi kasus penggunaan ini dapat dilihat pada Tabel 3.3. Diagram aktivitas dan diagram urutan dari kasus penggunaan ini bisa dilihat pada Gambar 3.2 dan Lampiran B Gambar 8.2.

Tabel 3.3 Spesifikasi Kasus Penggunaan Mengunggah Rekaman Data ke *Server*

Nama	Mengunggah Rekaman Data ke <i>server</i>
Kode	UC-0002
Deskripsi	Mengunggah data laporan pengamatan yang tersimpan pada basis data lokal ke <i>server</i> .
Tipe	Fungsional
Pemicu	Pengguna memilih menu untuk mengunggah rekaman data ke <i>server</i> .
Aktor	Pengguna
Kondisi Awal	-
Aliran: - Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih menu untuk mengunggah rekaman data ke <i>server</i>. 2. Sistem menampilkan antamuka yang berisi rekaman data. 3. Pengguna menekan tombol untuk mengunggah data pengukuran ke <i>server</i>. 4. Sistem menyimpan data di <i>server</i>. 5. Sistem menampilkan pesan bahwa data berhasil disimpan di <i>server</i>.
- Kejadian Alternatif	<ol style="list-style-type: none"> 4.A. Koneksi <i>internet</i> terputus <ol style="list-style-type: none"> 4.A.1 Sistem menampilkan pesan bahwa data tidak berhasil disimpan di <i>server</i>. 4.A.2 Aliran selesai.

Kondisi Akhir	Sistem menampilkan pesan bahwa data berhasil disimpan di server.
----------------------	--



Gambar 3.2 Diagram Aktivitas Menunggah Rekaman Data ke Server

3.1.3.2.3 Menampilkan Riwayat Pengukuran

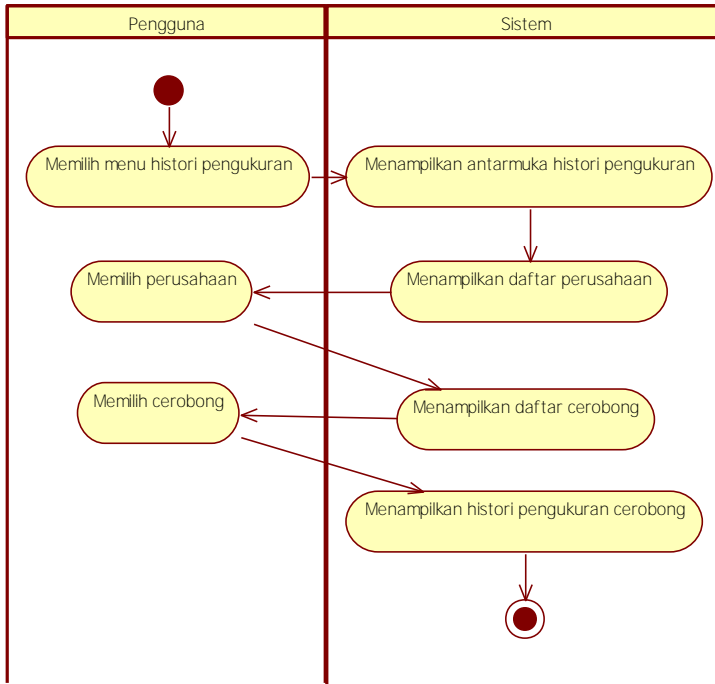
Pada kasus penggunaan ini, sistem menerima *input* berupa perintah untuk menampilkan riwayat pengukuran. Kemudian sistem akan menampilkan data riwayat pengukuran yang disajikan dalam bentuk *list* maupun grafik. Spesifikasi kasus penggunaan ini dapat dilihat pada Tabel 3.4. Diagram aktivitas dan diagram urutan dari

kasus penggunaan ini bisa dilihat pada Gambar 3.3 dan Lampiran B Gambar 8.3.

Tabel 3.4 Spesifikasi Kasus Penggunaan Menampilkan Riwayat Pengukuran

Nama	Menampilkan riwayat pengukuran
Kode	UC-0003
Deskripsi	Menampilkan riwayat pengukuran tingkat kepekatan asap.
Tipe	Fungsional
Pemicu	Pengguna memilih menu untuk menampilkan riwayat pengukuran tingkat kepekatan asap.
Aktor	Pengguna
Kondisi Awal	-
Aliran: - Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih menu untuk menampilkan riwayat pengukuran. 2. Pengguna memilih perusahaan. 3. Sistem menampilkan peta lokasi cerobong. 4. Pengguna memilih cerobong yang akan dilihat riwayatnya. 5. Sistem menampilkan antarmuka yang berisi riwayat pengukuran dalam bentuk <i>list</i> dan grafik.
- Kejadian Alternatif	-

Kondisi Akhir	Sistem menampilkan antarmuka yang berisi riwayat pengukuran dalam bentuk <i>list</i> dan grafik.
----------------------	--



Gambar 3.3 Diagram Aktivitas Menampilkan Riwayat Pengukuran

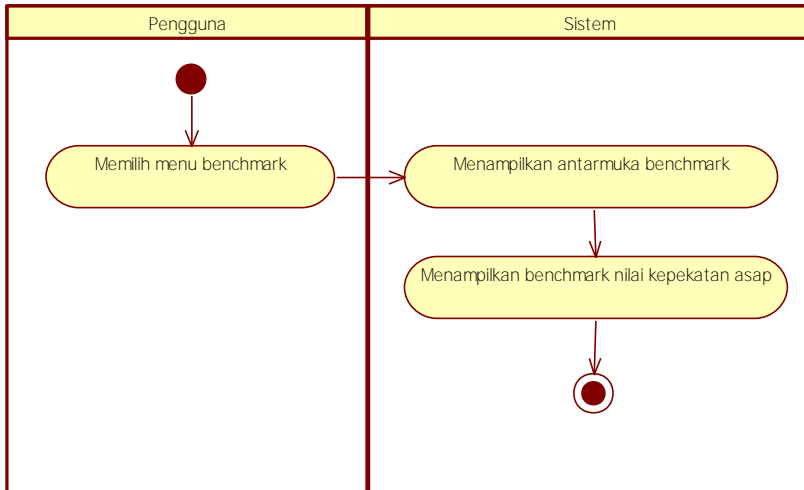
3.1.3.2.4 Menampilkan Benchmark Nilai Kepekatan Asap

Pada kasus penggunaan ini, sistem menerima *input* berupa perintah untuk menampilkan *benchmark*. Kemudian sistem akan menampilkan antarmuka yang berisi diagram yang menunjukkan *benchmark* pengamatan asap. *Benchmark* berisi nilai kepekatan tertinggi, nilai kepekatan rata-rata, dan nilai kepekatan terendah.

Spesifikasi kasus penggunaan ini dapat dilihat pada Tabel 3.5. Diagram aktivitas dan diagram urutan dari kasus penggunaan ini bisa dilihat pada Gambar 3.4 dan Gambar 3.5.

Tabel 3.5 Spesifikasi Kasus Penggunaan Menampilkan *Benchmark* Nilai Kepekatan Asap

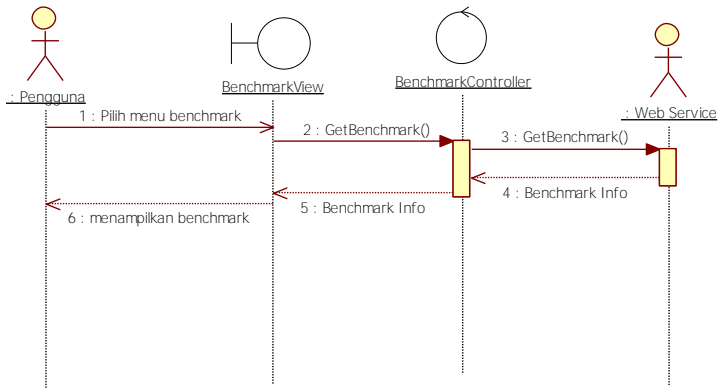
Nama	Menampilkan <i>benchmark</i> nilai kepekatan asap
Kode	UC-0004
Deskripsi	Menampilkan <i>benchmark</i> nilai kepekatan asap yang berisi nilai kepekatan tertinggi, nilai kepekatan rata-rata, dan nilai kepekatan terendah.
Tipe	Fungsional
Pemicu	Pengguna memilih menu untuk menampilkan <i>benchmark</i> nilai kepekatan asap
Aktor	Pengguna
Kondisi Awal	-
Aliran: - Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih menu untuk menampilkan <i>benchmark</i> nilai kepekatan asap. 2. Sistem menampilkan <i>benchmark</i> nilai kepekatan asap.
- Kejadian Alternatif	-
Kondisi Akhir	Sistem menampilkan <i>benchmark</i> nilai kepekatan asap.



Gambar 3.4 Diagram Aktivitas Menampilkan *Benchmark* Nilai Kepekatan Asap

3.1.3.3 Kebutuhan Fungsional

Kebutuhan fungsional berisi proses-proses yang harus dimiliki sistem. Kebutuhan fungsional mendefinisikan layanan yang harus disediakan dan reaksi sistem terhadap masukan atau pada situasi tertentu. Daftar kebutuhan fungsional dapat dilihat pada Tabel 3.6.



Gambar 3.5 Diagram Urutan Menampilkan *Benchmark* Nilai Kepekatan Asap

Tabel 3.6 Daftar Kebutuhan Fungsional Perangkat Lunak

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
F-0001	Mengukur tingkat kepekatan asap	Pengguna dapat mengetahui tingkat kepekatan asap berdasarkan foto yang diambil pengguna.
F-0002	Mengunggah rekaman data	Pengguna dapat mengunggah data pengamatan yang disimpan sementara pada basis data lokal ke <i>server</i>
F-0003	Menampilkan riwayat pengukuran	Pengguna dapat menampilkan riwayat pengukuran tingkat kepekatan asap yang dilakukan oleh pengguna.
F-0004	Menampilkan <i>benchmark</i> nilai kepekatan asap	Pengguna dapat menampilkan <i>benchmark</i> nilai kepekatan asap yang berisi nilai kepekatan tertinggi, nilai

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
		kepekatan rata-rata, dan nilai kepekatan terendah.

3.2. Perancangan

Perancangan perangkat lunak dibagi menjadi beberapa bagian yaitu perancangan diagram kelas dan perancangan antarmuka.

3.2.1. Lingkungan Perancangan Perangkat Lunak

Spesifikasi perangkat keras serta perangkat lunak yang digunakan dalam tahap perancangan perangkat lunak yang dikerjakan pada tugas akhir ini mempunyai rincian yang ditampilkan pada Tabel 3.7.

Tabel 3.7 Lingkungan Perancangan Perangkat Lunak

Perangkat Keras	Prosesor	Intel® Core™ i7-2670QM CPU (2.20GHz)
	Memori Primer	4 GB
	Memori Sekunder	1 TB
Perangkat Lunak	Sistem Operasi	Windows 8.1
	Perangkat Lunak	Visual Studio 2012 Ultimate Windows Phone SDK 8.0

3.2.2 Arsitektur Sistem

Rancangan arsitektur sistem pengukuran dan penyimpanan tingkat kepekatan asap yang akan dibuat secara lengkap ditunjukkan oleh Gambar 3.6. Sistem pengukuran tingkat kepekatan asap ini dibuat pada perangkat berbasis *desktop* dan perangkat bergerak. Namun pada tugas akhir ini pembuatan aplikasi difokuskan pada perangkat bergerak. Sehingga yang diimplementasikan pada tugas akhir ini adalah pengukuran tingkat kepekatan asap pada perangkat bergerak dan penyimpanan datanya pada *server*, ditunjukkan oleh bagian yang dibatasi garis hijau pada Gambar 3.6. Alur kerja sistem pertama kali adalah mengambil citra asap melalui aplikasi pada perangkat bergerak. Kemudian citra asap akan diproses untuk mendapatkan hasil pengukuran tingkat kepekatan asap. Pengguna kemudian bisa menambahkan data pendukung pengamatan seperti tanggal pengamatan, tempat pengamatan dan nama pengamat. Pengguna bisa mengunggah datanya ke *server*. Selain itu pengguna juga bisa menampilkan histori hasil pengamatan yang tersimpan pada *server*.



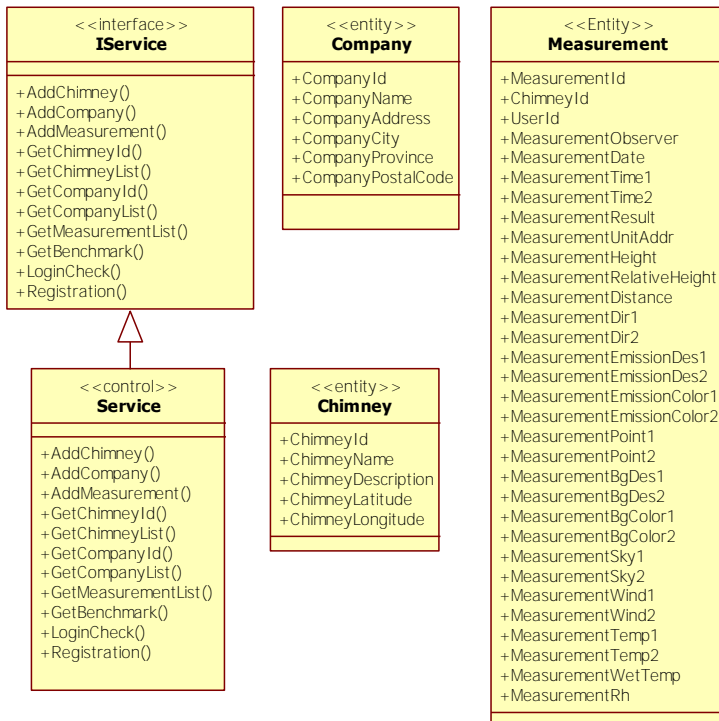
Gambar 3.6 Arsitektur Sistem

3.2.3 Perancangan Aplikasi *Web Service*

Tahap perancangan aplikasi *web service* hanya terdiri dari perancangan diagram kelas.

3.2.3.1 Perancangan Diagram Kelas

Diagram kelas untuk *web service* dapat dilihat pada Gambar 3.7. Terdapat lima kelas penyusun arsitektur *web service* yang terdiri dari dua kelas *control* dan tiga kelas *entity*.



Gambar 3.7 Diagram Kelas Aplikasi *Web Service*

Kelas *control* terdiri dari dua kelas yaitu kelas *IService* dan *Service*. Kelas *control* yang akan diimplementasikan sesuai dengan *pattern* dimiliki oleh WCF, di mana terdapat sebuah kelas utama yang berisi *method* yang dipanggil oleh aplikasi client, diturunkan dari kelas *interface*. Kelas *IService* merupakan kelas *interface* yang berisi *method* yang harus diimplementasikan pada kelas *Service*. *Method* yang diimplementasikan adalah *AddChimney* untuk menambahkan cerobong baru, *GetChimneyList* untuk mendapatkan *list* cerobong, *AddMeasurement* untuk menambahkan data pengamatan baru, *GetChimneyId* untuk mendapatkan *id* dari cerobong, *AddCompany* untuk menambahkan perusahaan baru, *GetCompanyList* untuk mendapatkan *list* perusahaan, *GetMeasurementList* untuk mendapatkan *list* data pengukuran, *GetBenchmark* untuk mendapatkan *benchmark* nilai kepekatan, *LoginCheck* untuk pengecekan proses *login*, dan *Registration* untuk menambahkan data registrasi pengguna baru.

Kelas *entity* terdiri dari tiga kelas yaitu kelas *Measurement*, *Chimney*, dan *Company*. Kelas *Measurement* merupakan kelas yang berisi atribut sesuai dengan data pengukuran. Kelas *Chimney* berisi atribut sesuai dengan data cerobong. Sedangkan kelas *Company* berisi atribut sesuai dengan data perusahaan.

3.2.4 Perancangan Aplikasi Perangkat Bergerak

Tahap perancangan aplikasi perangkat bergerak terdiri dari perancangan diagram kelas, perancangan antarmuka, dan perancangan proses pengukuran kepekatan asap.

3.2.4.1 Perancangan Diagram Kelas

Diagram Kelas untuk aplikasi perangkat bergerak dilampirkan pada Lampiran C. Diagram Kelas. Terdapat 9 kelas *boundary*, 9 kelas *control*, dan 3 kelas *entity*. Kelas *boundary* terdiri dari kelas *CameraView*, *MeasurementView*, *CompanyView*, *AddCompanyView*, *ChimneyMapView*, *MeasurementFormView*, *DraftView*, *HistoryView*, *BenchmarkView*. Kelas *CameraView* merupakan kelas yang digunakan untuk membuat antarmuka untuk

mengambil foto asap. Kelas *MeasurementView* merupakan kelas yang digunakan untuk membuat antarmuka untuk menampilkan hasil pengukuran kepekatan asap. Kelas *CompanyView* merupakan kelas yang digunakan untuk membuat antarmuka untuk memilih perusahaan. Kelas *AddCompanyView* merupakan kelas yang digunakan untuk membuat antarmuka untuk menambahkan perusahaan baru. Kelas *ChimneyMapView* merupakan kelas yang digunakan untuk membuat antarmuka peta yang berisi *pushpin* cerobong. Kelas *MeasurementFormView* merupakan kelas yang digunakan untuk membuat antarmuka untuk mengisi data pengamatan yang akan diunggah ke *server*. Kelas *DraftView* merupakan kelas yang digunakan untuk membuat antarmuka untuk menampilkan *list* rekaman data yang akan diunggah ke *server*. Kelas *HistoryView* merupakan kelas yang digunakan untuk membuat antarmuka untuk menampilkan riwayat pengukuran kepekatan asap. Kelas *BenchmarkView* merupakan kelas yang digunakan untuk membuat antarmuka untuk menampilkan *benchmark* nilai kepekatan asap.

Kelas *control* terdiri dari kelas *CameraController*, *MeasurementController*, *CompanyController*, *AddCompanyController*, *ChimneyMapController*, *MeasurementFormController*, *DraftController*, *HistoryController*, *BenchmarkController*. Kelas *CameraController* merupakan kelas yang digunakan untuk mengambil foto asap. Kelas *MeasurementController* merupakan kelas yang digunakan untuk mengukur tingkat kepekatan asap dari foto yang telah diambil. Kelas *CompanyController* merupakan kelas yang digunakan untuk mendapatkan *list* perusahaan dan memproses pencarian perusahaan. Kelas *AddCompanyController* merupakan kelas yang digunakan untuk menambahkan perusahaan baru. Kelas *ChimneyMapController* merupakan kelas yang digunakan untuk mendapatkan *list* cerobong. Kelas *MeasurementFormController* merupakan kelas yang digunakan untuk menambahkan data pengamatan yang akan diunggah ke *server*. Kelas *DraftController* merupakan kelas yang digunakan

untuk mendapatkan *list* rekaman data yang akan diunggah ke *server*. Kelas *HistoryController* merupakan kelas yang digunakan untuk mendapatkan data riwayat pengukuran kepekatan asap. Kelas *BenchmarkController* merupakan kelas yang digunakan untuk mendapatkan *benchmark* nilai kepekatan asap.

Kelas *entity* terdiri dari kelas *Measurement*, *Company*, dan *Chimney*. Kelas *Measurement* berisi atribut data pengukuran kepekatan asap. Kelas *Company* berisi atribut data perusahaan. Kelas *Chimney* berisi atribut data cerobong.

3.2.4.2 Perancangan Antarmuka Pengguna

Subbab ini akan menjelaskan bagaimana rancangan antarmuka grafis pengguna sistem yang dibangun pada aplikasi ini. Rancangan antarmuka grafis pengguna yang dibahas meliputi kontrol dan ketentuan masukan.

3.2.4.2.1 Rancangan Antarmuka Halaman Login

Halaman ini merupakan tampilan bagi pengguna untuk melakukan *login* pada aplikasi. Pada halaman ini pengguna memasukkan teks *username* dan *password* sesuai dengan *username* dan *password* saat melakukan *register* pada aplikasi. *Username* yang digunakan berupa email. Jika proses *login* berhasil maka pengguna dapat menggunakan halaman-halaman lain untuk melakukan proses lainnya. Rancangan antarmuka halaman Login ini dapat dilihat pada Gambar 3.8 Rancangan Antarmuka Halaman Login. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.8.



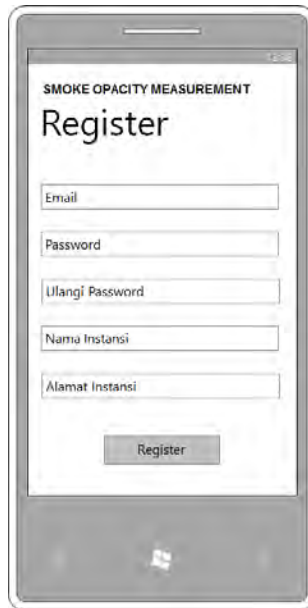
Gambar 3.8 Rancangan Antarmuka Halaman Login

Tabel 3.8 Spesifikasi Atribut Antarmuka Halaman Login

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	textUsername	<i>TextBox</i>	Memasukkan teks <i>username</i> berupa <i>email</i> dari pengguna	<i>String</i>
2	textPassword	<i>TextBox</i>	Memasukkan teks <i>password</i> dari pengguna	<i>String</i>
3	btnLogin	<i>Button</i>	Menjalankan perintah untuk melakukan proses <i>login</i>	<i>Selection Event</i>

3.2.4.2.2 Rancangan Antarmuka Halaman Register

Halaman ini merupakan tampilan bagi pengguna untuk melakukan *register* pada aplikasi. Pada halaman ini pengguna memasukkan teks *email*, *password*, konfirmasi *password*, nama instansi, dan alamat instansi. Jika proses *register* berhasil maka pengguna dapat melakukan proses *login* pada halaman Login. Rancangan halaman Register dapat dilihat pada Gambar 3.9. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.9.



The image shows a mobile application interface for registration. At the top, it says "SMOKE OPACITY MEASUREMENT" and "Register". Below this, there are five input fields: "Email", "Password", "Ulangi Password", "Nama Instansi", and "Alamat Instansi". At the bottom of the form is a "Register" button.

Gambar 3.9 Rancangan Antarmuka Halaman Register

Tabel 3.9 Spesifikasi Atribut Antarmuka Halaman Register

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	textUsername	<i>TextBox</i>	Memasukkan teks <i>username</i> berupa <i>email</i> dari pengguna	<i>String</i>
2	textPassword	<i>TextBox</i>	Memasukkan teks <i>password</i> dari pengguna	<i>String</i>
3	textConfirmPassword	<i>TextBox</i>	Memasukkan teks konfirmasi <i>password</i> dari pengguna	<i>String</i>
4	textOfficeName	<i>TextBox</i>	Memasukkan teks nama instansi	<i>String</i>
5	textOfficeAddress	<i>TextBox</i>	Memasukkan teks alamat instansi	<i>String</i>
5	btnRegister	<i>Button</i>	Menjalankan perintah untuk melakukan proses <i>register</i>	<i>Tap Event</i>

3.2.4.2.3 Rancangan Antarmuka Halaman Pilihan Menu

Halaman ini merupakan tampilan bagi pengguna untuk memilih menu aplikasi. Terdapat menu Pengukuran, menu Rekaman Data, menu Riwayat, dan menu Benchmark. Rancangan halaman Menu dapat dilihat pada Gambar 3.10. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.10.



Gambar 3.10 Rancangan Antarmuka Halaman Pilihan Menu

Tabel 3.10 Spesifikasi Atribut Antarmuka Halaman Pilihan Menu

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	borderMeasurement	<i>Border</i>	Menjalankan perintah untuk memilih menu pengukuran	<i>Tap Event</i>
2	borderDraft	<i>Border</i>	Menjalankan perintah untuk memilih menu rekaman data	<i>Tap Event</i>

3	borderHistory	<i>Border</i>	Menjalankan perintah untuk memilih menu riwayat	<i>Tap Event</i>
4	borderBenchmark	<i>Border</i>	Menjalankan perintah untuk memilih menu <i>benchmark</i>	<i>Tap Event</i>
5	appBarAbout	<i>Application Bar</i>	Menjalankan perintah untuk menampilkan halaman tentang aplikasi	<i>Tap Event</i>
6	appBarSetting	<i>Application Bar</i>	Menjalankan perintah untuk menampilkan halaman pengaturan batas aman kepekatan asap	<i>Tap Event</i>
7	appBarLogout	<i>Application Bar</i>	Menjalankan perintah untuk melakukan proses <i>logout</i>	<i>Tap Event</i>

3.2.4.2.4 Rancangan Antarmuka Halaman Ambil Gambar

Halaman ini merupakan tampilan bagi pengguna untuk mengambil gambar setelah memilih menu pengukuran. Pada halaman ini akan ditampilkan *streaming* gambar. Pengguna harus mengarahkan asap pada kotak merah. Kepekatan asap akan dihitung berdasarkan gambar pada kotak merah tersebut. Untuk mengambil gambar, pengguna harus memilih tombol Ambil Gambar. Setelah tombol tersebut diklik, sistem akan menampilkan halaman hasil kepekatan. Rancangan halaman ambil gambar dapat dilihat pada Gambar 3.11.

Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.11.



Gambar 3.11 Rancangan Antarmuka Halaman Ambil Gambar

Tabel 3.11 Spesifikasi Atribut Antarmuka Halaman Ambil Gambar

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	previewCamera	Video Brush	Menampilkan <i>streaming</i> gambar	<i>Image Stream</i>

2	btnCaptureImage	<i>Button</i>	Menjalankan perintah untuk mengambil gambar dan menampilkan halaman hasil kepekatan	<i>Tap event</i>
---	-----------------	---------------	---	------------------

3.2.4.2.5 Rancangan Antarmuka Halaman Hasil Pengukuran Kepekatan

Halaman ini merupakan tampilan bagi pengguna untuk melihat hasil nilai kepekatan asap dari gambar yang telah diambil. Yang ditampilkan pada halaman ini adalah nilai kepekatan dan status dari nilai kepekatan tersebut, apakah di atas batas aman atau di bawah batas aman. Setelah pengguna memilih tombol Lengkapi Data Pengamatan, sistem akan menampilkan daftar perusahaan. Rancangan halaman hasil pengukuran kepekatan dapat dilihat pada Gambar 3.12. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.12.



Gambar 3.12 Rancangan Antarmuka Halaman Hasil Pengukuran Kepekatan

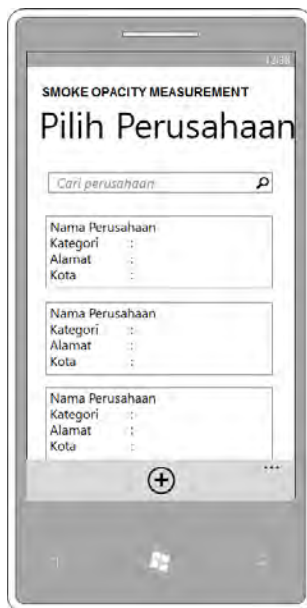
Tabel 3.12 Spesifikasi Atribut Antarmuka Halaman Hasil Pengukuran Kepekatan

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	textOpacity	<i>TextBlock</i>	Menampilkan hasil nilai kepekatan	<i>String</i>
2	textStatus	<i>TextBlock</i>	Menampilkan status nilai kepekatan	<i>String</i>

3	btnCompleteData	<i>Button</i>	Menjalankan perintah untuk menampilkan halaman pilih perusahaan	<i>Tap event</i>
---	-----------------	---------------	---	------------------

3.2.4.2.6 Rancangan Antarmuka Halaman Pilih Perusahaan pada Fitur Pengukuran

Halaman ini merupakan tampilan bagi pengguna untuk memilih perusahaan pemilik cerobong asap yang sedang diukur kepekatan asapnya. Pada halaman ini akan ditampilkan daftar perusahaan yang terletak pada kota yang sama dengan lokasi pengamatan. Pengguna dapat melakukan pencarian perusahaan pada kotak pencarian. Pengguna dapat memilih perusahaan dengan cara melakukan *tap* pada kotak yang berisi info perusahaan. Setelah pengguna memilih perusahaan, sistem akan menampilkan halaman peta cerobong. Apabila perusahaan tidak terdapat dalam daftar, pengguna dapat menambahkan perusahaan dengan melakukan *tap* pada *application bar* Tambah. Setelah itu sistem akan menampilkan halaman tambah perusahaan. Rancangan halaman pilih perusahaan pada fitur pengukuran dapat dilihat pada Gambar 3.13. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.13.



Gambar 3.13 Rancangan Antarmuka Halaman Pilih Perusahaan pada Fitur Pengukuran

Tabel 3.13 Spesifikasi Atribut Antarmuka Halaman Pilih Perusahaan pada Fitur Pengukuran

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	textSearch	<i>TextBox</i>	Memasukkan kata kunci pencarian perusahaan	<i>String</i>
2	listCompany	<i>ListBox</i>	Menampilkan daftar perusahaan	<i>String</i>

3	appBarAddCompany	<i>Application Bar</i>	Menjalankan perintah untuk menampilkan halaman tambah perusahaan	<i>Tap event</i>
---	------------------	------------------------	--	------------------

3.2.4.2.7 Rancangan Antarmuka Halaman Tambah Perusahaan

Halaman ini merupakan tampilan bagi pengguna untuk menambahkan perusahaan baru. Pengguna harus mengisi info perusahaan seperti kategori perusahaan, nama, dan lokasi. Pengguna dapat mendapatkan lokasi saat ini dengan memilih tombol Dapatkan Lokasi Saat Ini. Setelah pengguna memilih tombol Tambahkan, sistem akan menampilkan halaman peta cerobong. Rancangan halaman tambah perusahaan dapat dilihat pada Gambar 3.14. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.14.

The image shows a mobile application screen for adding a new company. The screen is titled "SMOKE OPACITY MEASUREMENT" and "Tambah Baru". It contains a form with the following fields: "Isi Data Perusahaan", "Kategori perusahaan" (dropdown), "Email", "Alamat Perusahaan", "Kota", "Provinsi", and "Kode Pos". There are two buttons: "Dapatkan Lokasi Saat Ini" and "Tambahkan".

Gambar 3.14 Rancangan Antarmuka Halaman Tambah Perusahaan

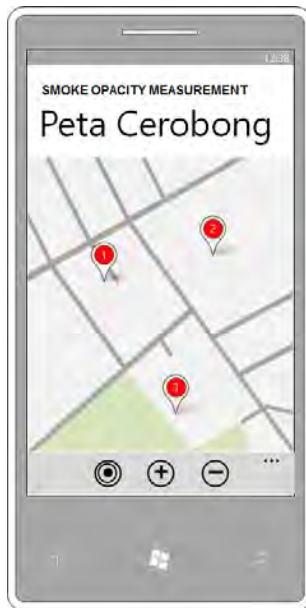
Tabel 3.14 Spesifikasi Atribut Antarmuka Halaman Tambah Perusahaan

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	dropdownCategory	<i>DropDown Box</i>	Menampilkan pilihan kategori perusahaan	<i>String</i>
2	textCompanyName	<i>TextBox</i>	Memasukkan teks nama perusahaan	<i>String</i>

3	btnGetLocation	<i>Button</i>	Menjalankan perintah untuk mendapatkan lokasi saat ini	<i>Tap event</i>
4	textCompanyAddress	<i>TextBox</i>	Memasukkan teks alamat perusahaan	<i>String</i>
5	textCompanyCity	<i>TextBox</i>	Memasukkan teks kota perusahaan	<i>String</i>
6	textProvince	<i>TextBox</i>	Memasukkan teks provinsi perusahaan	<i>String</i>
7	textPostalCode	<i>TextBox</i>	Memasukkan teks kode pos perusahaan	<i>String</i>
8	btnAddCompany	<i>Button</i>	Menjalankan perintah untuk menambahkan perusahaan	<i>Tap event</i>

3.2.4.2.8 Rancangan Antarmuka Halaman Peta Cerobong

Halaman ini berisi peta yang menampilkan lokasi cerobong dari perusahaan yang telah dipilih pengguna. Untuk memilih cerobong, pengguna dapat melakukan *tap* pada *pushpin* cerobong yang terdapat pada peta. Pengguna hanya dapat menambahkan cerobong baru jika memilih menu pengukuran. Untuk menambahkan cerobong baru, pengguna melakukan *tap* pada lokasi pada peta yang belum terdapat cerobong dan melakukan *tap* lagi pada *pushpin* yang muncul. Selanjutnya akan muncul *MessageBox* yang berisi borang pengisian cerobong baru. Rancangan halaman peta cerobong dapat dilihat pada Gambar 3.15. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.15.



Gambar 3.15 Rancangan Antarmuka Halaman Peta Cerobong

Tabel 3.15 Spesifikasi Atribut Antarmuka Halaman Peta Cerobong

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	chimneyMap	Map	Menampilkan peta cerobong	Map
2	pushpin	Pushpin	Menampilkan lokasi cerobong	Tap event

3	appBarLocation	<i>Application Bar</i>	Menjalankan perintah untuk menampilkan lokasi saat ini	<i>Tap event</i>
4	appBarZoomIn	<i>Application Bar</i>	Menjalankan perintah untuk <i>zoom in</i> peta	<i>Tap event</i>
5	appBarZoomOut	<i>Application Bar</i>	Menjalankan perintah untuk <i>zoom out</i> peta	<i>Tap event</i>

3.2.4.2.9 Rancangan Antarmuka *MessageBox* Pilih Cerobong

MessageBox ini berisi konfirmasi bagi pengguna untuk memilih cerobong. *MessageBox* akan ditampilkan oleh sistem setelah pengguna melakukan *tap* pada *pushpin* cerobong pada peta. Rancangan *MessageBox* konfirmasi untuk memilih cerobong dapat dilihat pada Gambar 3.16. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.16.



Gambar 3.16 Rancangan Antarmuka *MessageBox* Pilih Cerobong

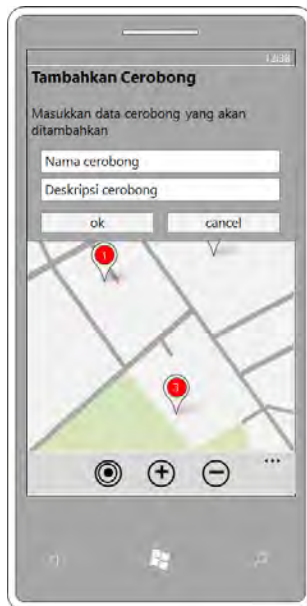
Tabel 3.16 Spesifikasi Atribut Antarmuka *MessageBox* Pilih Cerobong

No	Nama Antarmuka	Atribut	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	textChimneyName		<i>TextBox</i>	Memasukkan teks nama cerobong	<i>String</i>
2	textChimneyDescription		<i>TextBox</i>	Memasukkan teks deskripsi cerobong	<i>String</i>

3	btnOK	<i>Button</i>	Menjalankan perintah untuk memilih cerobong	<i>Tap event</i>
4	btnCancel	<i>Button</i>	Menjalankan perintah untuk memilih cerobong	<i>Tap event</i>

3.2.4.2.10 Rancangan Antarmuka *MessageBox* Tambah Cerobong

MessageBox ini berisi borang pengisian data cerobong baru. Untuk menambahkan cerobong baru, pengguna harus mengisi teks nama cerobong dan deskripsi cerobong. Setelah pengguna menekan tombol Ok, sistem akan menampilkan halaman pengisian formulir data pengamatan. Rancangan *MessageBox* untuk menambahkan cerobong dapat dilihat pada Gambar 3.17. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.17.



Gambar 3.17 Rancangan Antarmuka *MessageBox* Tambah Cerobong

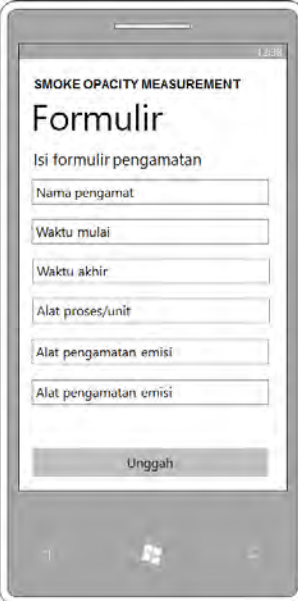
Tabel 3.17 Spesifikasi Atribut Antarmuka *MessageBox* Tambah Cerobong

No	Nama Antarmuka	Atribut	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	btnOK		<i>Button</i>	Menjalankan perintah untuk menambahkan cerobong	<i>Tap event</i>
2	btnCancel		<i>Button</i>	Menjalankan perintah untuk membatalkan	<i>Tap event</i>

			penambahan cerobong	
--	--	--	---------------------	--

3.2.4.2.11 Rancangan Antarmuka Halaman Formulir Pengamatan

Antarmuka ini berisi borang pengisian data pengamatan asap. Setelah pengguna menekan tombol Unggah, sistem akan mengunggah data pengamatan ke *server*. Apabila koneksi ke *server* terputus, data pengamatan akan disimpan sementara di basis data lokal. Rancangan antarmuka halaman formulir pengamatan dapat dilihat pada Gambar 3.18. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.18.



The image shows a mobile application interface for "SMOKE OPACITY MEASUREMENT". The screen displays a form titled "Formulir" with the subtitle "Isi formulir pengamatan". The form contains several input fields: "Nama pengamat", "Waktu mulai", "Waktu akhir", "Alat proses/unit", "Alat pengamatan emisi", and "Alat pengamatan emisi". At the bottom of the form is a button labeled "Unggah". The interface is displayed on a smartphone screen with a status bar at the top showing the time 12:38.

Gambar 3.18 Rancangan Antarmuka Halaman Formulir Pengamatan

Tabel 3.18 Spesifikasi Atribut Antarmuka Halaman Formulir Pengamatan

No	Nama Atribut	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	txtName	<i>TextBox</i>	Memasukkan teks nama pengamat.	<i>String</i>
2	txtTimeStart	<i>TextBox</i>	Memasukkan waktu mulai	<i>String</i>
3	txtTimeStop	<i>TextBox</i>	Memasukkan waktu berakhir	<i>String</i>
4	txtProcessTool	<i>TextBox</i>	Memasukkan alat proses	<i>String</i>
5	txtEmissionTool	<i>TextBox</i>	Memasukkan alat pengamatan emisi	<i>String</i>
2	btnUpload	<i>Button</i>	Menjalankan perintah untuk mengunggah data pengamatan	<i>Tap event</i>

3.2.4.2.12 Rancangan Antarmuka Halaman Rekaman Data

Halaman ini merupakan tampilan bagi pengguna untuk mengunggah rekaman data pengamatan ke *server*. Rekaman data

merupakan data pengamatan yang disimpan pada basis data lokal. Setelah pengguna menekan tombol Unggah, sistem akan mengunggah rekaman data ke *server*. Rancangan halaman rekaman data dapat dilihat pada Gambar 3.19. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.19.

The image shows a mobile application interface for 'SMOKE OPACITY MEASUREMENT'. The screen is titled 'Rekaman Data'. It features three identical data entry forms stacked vertically. Each form contains three input fields: 'Tanggal Pengamatan', 'Nama Pengamat', and 'Nilai Kepekatan'. At the bottom of the screen, there is a prominent 'Unggah' (Upload) button.

Gambar 3.19 Rancangan Antarmuka Halaman Rekaman Data

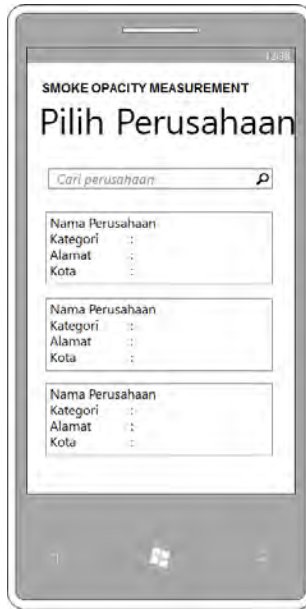
Tabel 3.19 Spesifikasi Atribut Antarmuka Halaman Rekaman Data

No	Nama Atribut	Atribut	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran

1	listDraft	<i>ListBox</i>	Menampilkan daftar rekaman data	<i>String</i>
2	btnUpload	<i>Button</i>	Menjalankan perintah untuk mengunggah rekaman data ke <i>server</i>	<i>Tap event</i>

3.2.4.2.13 Rancangan Antarmuka Halaman Pilih Perusahaan pada Fitur Riwayat

Halaman ini merupakan tampilan bagi pengguna untuk memilih perusahaan pemilik cerobong asap yang akan dilihat riwayat pengamatannya. Pada halaman ini akan ditampilkan daftar perusahaan yang terletak pada kota yang sama dengan lokasi pengamatan. Pengguna dapat melakukan pencarian perusahaan pada kotak pencarian. Pengguna dapat memilih perusahaan dengan cara melakukan *tap* pada kotak yang berisi info perusahaan. Setelah pengguna memilih perusahaan, sistem akan menampilkan halaman peta cerobong. Rancangan halaman pilih perusahaan pada fitur riwayat dapat dilihat pada Gambar 3.20. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.20.



Gambar 3.20 Rancangan Antarmuka Halaman Pilih Perusahaan pada Fitur Riwayat

Tabel 3.20 Spesifikasi Atribut Antarmuka Halaman Pilih Perusahaan pada Fitur Riwayat

No	Nama Atribut	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	textSearch	<i>TextBox</i>	Memasukkan kata kunci pencarian perusahaan	<i>String</i>
2	listCompany	<i>ListBox</i>	Menampilkan daftar perusahaan	<i>String</i>

3.2.4.2.14 Rancangan Antarmuka Halaman Riwayat

Halaman ini berisi riwayat pengamatan cerobong asap. Rancangan halaman pilih perusahaan pada fitur riwayat dapat dilihat pada Gambar 3.21. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.21.



Gambar 3.21 Rancangan Antarmuka Halaman Riwayat

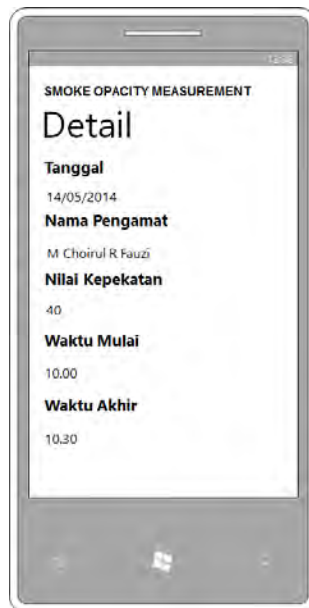
Tabel 3.21 Spesifikasi Atribut Antarmuka Halaman Riwayat

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	btnShowChart	Button	Menjalankan perintah untuk menampilkan	String

			halaman grafik riwayat	
2	listMeasurement	ListBox	Menampilkan riwayat pengukuran	String

3.2.4.2.15 Rancangan Antarmuka Halaman Detail Pengamatan

Halaman ini berisi detail dari pengamatan asap yang yang dipilih pengguna pada daftar riwayat. Rancangan halaman detail pengamatan dapat dilihat pada Gambar 3.22. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.22.



Gambar 3.22 Rancangan Antarmuka Halaman Detail Pengamatan

Tabel 3.22 Spesifikasi Atribut Antarmuka Halaman Detail Pengamatan

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	chart	<i>Chart</i>	Menampilkan riwayat pengukuran dalam bentuk grafik	<i>Chart</i>
2	toggleBenchmark	<i>Toggle</i>	Menjalankan perintah untuk menampilkan atau menyembunyikan <i>benchmark</i>	<i>Tap event</i>
3	benchmark	<i>Line</i>	Menampilkan <i>benchmark</i> nilai kepekatan asap	<i>Line</i>

3.2.4.2.16 Rancangan Antarmuka Halaman Grafik Riwayat
 Halaman ini berisi riwayat pengamatan dalam bentuk grafik. Pengguna dapat menampilkan *benchmark* nilai kepekatan asap pada grafik tersebut dengan melakukan *tap* pada *toggle benchmark*. *Benchmark* terdiri dari nilai kepekatan tertinggi, nilai kepekatan rata-rata, dan nilai kepekatan terendah. Rancangan halaman grafik riwayat dapat dilihat pada Gambar 3.23. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.23.



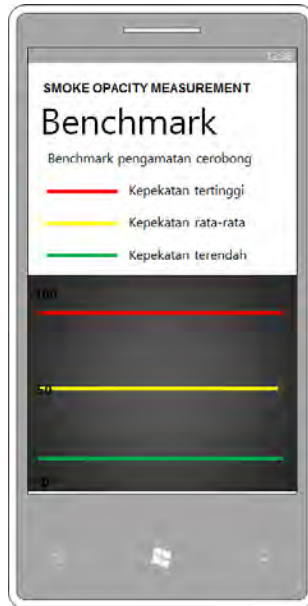
Gambar 3.23 Rancangan Antarmuka Halaman Grafik Riwayat

Tabel 3.23 Spesifikasi Atribut Antarmuka Halaman Grafik Riwayat

No	Nama Atribut Antarmuka	Atribut	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	chart		<i>Chart</i>	Menampilkan riwayat pengukuran dalam bentuk grafik	<i>Chart</i>
2	toggleBenchmark		<i>Toggle</i>	Menjalankan perintah untuk menampilkan atau menyembunyikan <i>benchmark</i>	<i>Tap event</i>
3	benchmark		<i>Line</i>	Menampilkan <i>benchmark</i> nilai kepekatan asap	<i>Line</i>

3.2.4.2.17 Rancangan Antarmuka Halaman Benchmark

Halaman ini berisi *benchmark* nilai kepekatan asap. *Benchmark* terdiri dari nilai kepekatan tertinggi, nilai kepekatan rata-rata, dan nilai kepekatan terendah. Rancangan halaman grafik riwayat dapat dilihat pada Gambar 3.24. Penjelasan mengenai atribut-atribut yang terdapat pada halaman ini dapat dilihat pada Tabel 3.24.



Gambar 3.24 Rancangan Antarmuka Halaman Benchmark

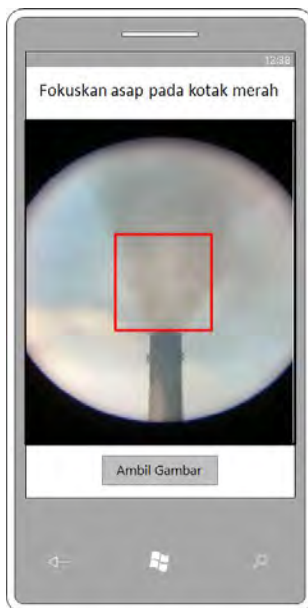
Tabel 3.24 Spesifikasi Atribut Antarmuka Halaman Benchmark

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan / Keluaran
1	chart	<i>Chart</i>	Menampilkan grafik kosong	<i>Chart</i>

2	benchmark	<i>Line</i>	Menampilkan <i>benchmark</i> nilai kepekatan asap	<i>Line</i>
---	-----------	-------------	---	-------------

3.2.4.3 Perancangan Proses Pengukuran Kepekatan Asap

Proses pengukuran tingkat kepekatan asap menggunakan metode *image retrieval based on color histogram*. Dengan menggunakan metode tersebut tingkat kepekatan asap didapatkan dengan mencari *euclidean distance* dari histogram RGB yang paling kecil antara citra asap dengan masing-masing skala pada Ringelmann Smoke Chart. Citra asap dan masing-masing skala Ringelmann Smoke Chart yang dibandingkan harus sama-sama berukuran 512x512 *pixel*. Foto asap yang didapatkan dari kamera *smartphone* berukuran 2048x1536 *pixel*. Sehingga untuk mendapatkan *euclidean distance* dari citra asap dan skala Ringelmann Smoke Chart, citra asap dipotong menjadi berukuran 512x512 *pixel*. Citra asap yang dipotong tersebut sesuai dengan citra yang terdapat pada kotak merah pada antarmuka kamera yang ditunjukkan oleh gambar Gambar 3.25.



Gambar 3.25 Antarmuka Kamera

Proses untuk mendapatkan tingkat kepekatan asap adalah sebagai berikut:

1. Memotong citra asap menjadi ukuran 512×512 *pixel* sesuai dengan posisi kotak merah pada antarmuka kamera.
2. Mengekstrak informasi histogram RGB dari citra asap yang berukuran 512×512 *pixel* tersebut.
3. Melakukan normalisasi terhadap histogram RGB dari citra asap dan memasukkannya pada *array* dengan panjang 16 pada masing-masing komponen RGB.
4. Mengekstrak informasi histogram RGB dari citra masing-masing skala Ringelmann Smoke Chart.
5. Melakukan normalisasi terhadap histogram RGB dari citra masing-masing skala Ringelmann Smoke Chart dan memasukkannya pada *array* dengan panjang 16 pada masing-masing komponen RGB.

6. Menghitung *euclidean distance* antara histogram RGB citra asap dengan histogram RGB citra masing-masing skala pada Ringelmann Smoke Chart.
7. Tingkat kepekatan asap didapatkan dari skala pada Ringelmann Smoke Chart yang memiliki *euclidean distance* terkecil terhadap citra asap.

BAB IV

IMPLEMENTASI PERANGKAT LUNAK

Pada bab ini dibahas implementasi dari rancangan perangkat lunak yang didasari pada rancangan pada Bab III. Namun, rancangan tersebut tidak menutup kemungkinan mengalami perubahan-perubahan minor jika dirasa perlu.

4.1. Lingkungan Implementasi

Dalam merancang perangkat lunak ini digunakan beberapa perangkat pendukung sebagai berikut.

4.1.1. Lingkungan Implementasi Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan sistem adalah sebuah komputer dan Leap Motion. Spesifikasi dari komputer yang digunakan adalah *notebook* dengan prosesor Intel® Core™ i7-2670QM CPU (2.20GHz) dan RAM 4 GB.

4.1.2. Lingkungan Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pengembangan sistem adalah sebagai berikut:

- Microsoft Windows 8.1 sebagai Sistem Operasi.
- Visual Studio 2012 Ultimate.
- Windows Phone SDK.

4.2. Implementasi Antarmuka Pengguna

Pada subbab ini akan dibahas tentang implementasi antarmuka pengguna yang sudah dirancang pada Bab III. Antarmuka pengguna yang akan dibahas adalah antarmuka aplikasi berbasis Windows Phone.

4.3.1 Implementasi Antarmuka Halaman Login

Halaman Login merupakan halaman awal ketika pengguna memasuki aplikasi. Pada halaman ini pengguna harus memasukkan *email* dan *password* untuk bisa mengakses fitur yang dimiliki aplikasi. Apabila *email* dan *password* yang dimasukkan pengguna salah, aplikasi akan menampilkan *messagebox* yang berisi pesan bahwa pengguna salah memasukkan *email* dan *password*. Implementasi antarmuka halaman Login ditunjukkan oleh Gambar 4.1.



Gambar 4.1 Implementasi Halaman *Login*

4.3.2 Implementasi Antarmuka Halaman Register

Halaman Register merupakan antarmuka yang digunakan oleh pengguna untuk melakukan proses registrasi apabila pengguna belum memiliki akun. Pengguna harus mengisi *email*, *password*, konfirmasi *password*, nama instansi, dan alamat instansi.

Password yang dimiliki pengguna harus terdiri dari minimal 8 karakter. Apabila terdapat kesalahan dalam pengisian *field* pada proses registrasi ini, aplikasi akan menampilkan peringatan yang terletak di kanan atas *textbox*. Apabila proses registrasi berhasil, aplikasi akan menampilkan *messagebox* yang berisi pesan untuk melakukan proses *login*. Implementasi antarmuka halaman Register ditunjukkan oleh Gambar 4.2.

The image shows a mobile application interface for registration. At the top, there is a dark green header with the text 'Pengamatan Asap' in white. Below the header, the title 'Register Login' is displayed in a large, light grey font. The form consists of several input fields: an email field with the placeholder 'email@domain.com', a password field with a red error message 'Minimal 8 karakter' to its right, a confirm password field with a red error message 'Minimal 8 karakter' to its right, a name field with the placeholder 'Nama instansi', and an address field with the placeholder 'Alamat instansi'. At the bottom of the form, there is a dark green button with the text 'Register' in white. The status bar at the top right shows the time '11:49'.

Gambar 4.2 Implementasi Halaman Register

4.3.3 Implementasi Antarmuka Halaman Pilihan Menu

Halaman pilihan menu merupakan halaman yang ditampilkan apabila proses *login* berhasil. Terdapat 4 menu yang dapat diakses oleh pengguna, yaitu:

8. Pengukuran, merupakan menu untuk melakukan pengukuran tingkat kecepatan asap dan unggah data pengukuran.

9. Rekaman data, merupakan menu untuk mengunggah data pengukuran yang disimpan sementara pada basis data lokal.
10. Riwayat, merupakan menu untuk menampilkan riwayat pengukuran tingkat kepekatan asap yang dilakukan oleh pengguna.
11. Benchmark, merupakan menu untuk menampilkan *benchmark* nilai kepekatan asap yang terdiri dari nilai kepekatan tertinggi, nilai kepekatan rata-rata, dan nilai kepekataan terendah.

Implementasi halaman pilihan menu ditunjukkan oleh Gambar 4.3.



Gambar 4.3 Implementasi Halaman Pilihan Menu

4.3.4 Implementasi Antarmuka Halaman Ambil Gambar

Halaman ambil gambar merupakan halaman yang ditampilkan oleh aplikasi setelah pengguna memilih menu Pengukuran. Halaman ini

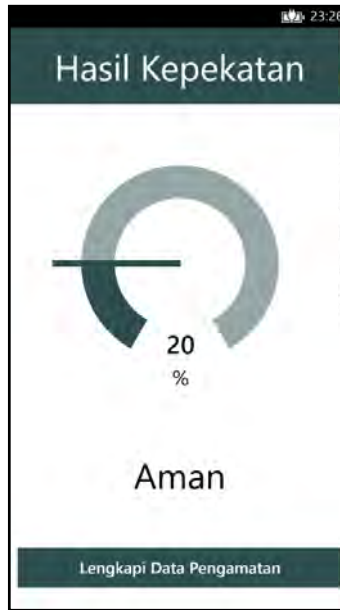
berisi *streaming* gambar dari kamera. Ketika melakukan pengukuran, pengguna harus memfokuskan asap pada kotak merah yang terletak di tengah-tengah halaman. Setelah menekan tombol Ambil Gambar, aplikasi akan menampilkan halaman Hasil Pengukuran. Implementasi antarmuka halaman Ambil Gambar ditunjukkan oleh Gambar 4.4.



Gambar 4.4 Implementasi Halaman Ambil Gambar

4.3.5 Implementasi Antarmuka Halaman Hasil Pengukuran

Halaman Hasil Pengukuran merupakan halaman yang ditampilkan setelah pengguna menekan tombol Ambil Gambar pada halaman Ambil Gambar. Pada halaman ini terdapat hasil pengukuran dalam bentuk angka dan status aman atau tidaknya. Setelah pengguna menekan tombol Lengkapi Data Pengamatan, aplikasi akan menampilkan halaman Pilih Perusahaan. Implementasi antarmuka halaman Hasil Pengukuran ditunjukkan oleh Gambar 4.5.



Gambar 4.5 Implementasi Halaman Hasil Pengukuran

4.3.6 Implementasi Antarmuka Halaman Pilih Perusahaan pada Fitur Pengukuran

Halaman Pilih Perusahaan pada fitur pengukuran merupakan halaman yang ditampilkan setelah pengguna menekan tombol Lengkapi Data Pengamatan pada halaman Ambil Gambar. Halaman ini menampilkan daftar perusahaan beserta infonya. Pengguna bisa mencari perusahaan yang diamati cerobongnya dengan mengetikkan nama perusahaan pada kotak pencarian Cari perusahaan. Jika perusahaan yang cerobongnya diamati terdapat pada daftar, pengguna bisa melakukan *tap* pada kotak info perusahaan. Setelah itu aplikasi akan menampilkan halaman Peta Cerobong. Jika perusahaan tidak terdapat pada daftar, pengguna bisa menambahkan perusahaan baru dengan menekan *application bar* tambah. Kemudian aplikasi akan menampilkan halaman

Tambah Perusahaan. Implementasi halaman Pilih Perusahaan pada fitur pengukuran ditunjukkan oleh Gambar 4.6.



Gambar 4.6 Implementasi Halaman Pilih Perusahaan pada Fitur Pengukuran

4.3.7 Implementasi Antarmuka Halaman Tambah Perusahaan

Halaman Tambah Perusahaan merupakan halaman yang ditampilkan setelah pengguna menekan *application bar* tambah pada halaman Pilih Perusahaan pada fitur pengukuran. Pada halaman ini pengguna harus mengisi nama perusahaan, alamat perusahaan, kota, provinsi, dan kode pos. Pengguna dapat mengisi lokasi secara otomatis sesuai posisi pengguna dengan menekan tombol Dapatkan Lokasi Saat Ini. Setelah pengguna mengisi data perusahaan dan menekan tombol Tambahkan, aplikasi akan

menampilkan halaman Peta Cerobong. Implementasi halaman Tambah Perusahaan ditunjukkan oleh Gambar 4.7.

The screenshot shows a mobile application interface for adding a company. At the top, there is a dark green header with the title "Tambah Perusahaan" in white. Below the header, the form is organized into several sections:

- Nama Perusahaan:** A text input field containing "PT Kalbe Farma".
- Dapatkan Lokasi Saat Ini:** A dark green button with white text.
- Alamat Perusahaan:** A text input field containing "Jalan Keputih Tegal Timur, Desa Keputih, Kecamatan Sukolilo".
- Kota:** A text input field containing "Surabaya".
- Provinsi:** A text input field containing "Jawa Timur".
- Kode Pos:** A text input field containing "60111".
- Tambahkan:** A dark green button with white text at the bottom of the form.

The status bar at the top right shows the time as 23:02.

Gambar 4.7 Implementasi Halaman Tambah Perusahaan

4.3.8 Implementasi Antarmuka Halaman Peta Cerobong

Halaman Peta Cerobong merupakan halaman yang ditampilkan setelah pengguna memilih perusahaan pada fitur pengukuran maupun pada fitur riwayat dan juga setelah menambahkan perusahaan pada fitur pengukuran. Halaman ini menampilkan peta yang berisi *pushpin* cerobong. Untuk memilih cerobong, pengguna melakukan *tap* pada *pushpin* cerobong. Setelah itu aplikasi akan menampilkan *messagebox* yang berisi pesan konfirmasi memilih cerobong. Untuk menambahkan cerobong baru, pengguna melakukan *tap* pada lokasi cerobong yang ingin ditambahkan. Setelah itu aplikasi akan menampilkan *messagebox* yang berisi borang pengisian data cerobong baru.

Pada peta ini terdapat 2 mode peta, yaitu mode *road* dan mode *hybrid*. Pengguna dapat mengubah mode tersebut pada *menu items* pada *application bar*. Pada *application bar* juga terdapat 3 *icon button* yang terdiri dari tunjukkan lokasi saat ini, *zoom in* peta, dan *zoom out* peta. Implementasi antarmuka halaman Peta Cerobong ditunjukkan oleh Gambar 4.8.

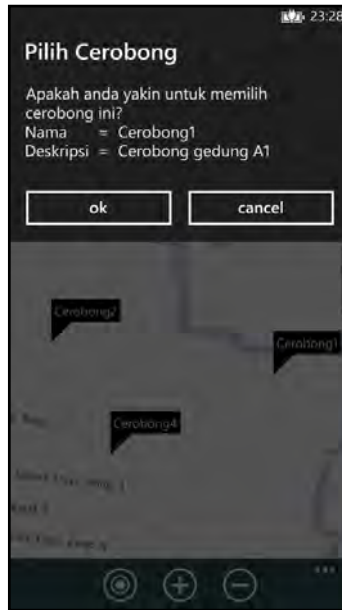


Gambar 4.8 Implementasi Halaman Peta Cerobong

4.3.9 Implementasi Antarmuka *MessageBox* Pilih Cerobong

Antarmuka *messagebox* Pilih Cerobong merupakan antarmuka yang ditampilkan setelah pengguna memilih cerobong pada halaman Peta Cerobong. *MessageBox* ini berisi pesan konfirmasi untuk memilih cerobong. Pada fitur pengukuran, jika pengguna menekan tombol ok, aplikasi akan menampilkan halaman Formulir Pengamatan. Sedangkan pada fitur riwayat, jika pengguna menekan tombol ok, aplikasi akan menampilkan halaman Riwayat.


Implementasi antarmuka *messagebox* Pilih Cerobong ditunjukkan oleh Gambar 4.9.



Gambar 4.9 Implementasi *MessageBox* Pilih Cerobong

4.3.10 Implementasi Antarmuka *MessageBox* Tambah Cerobong

Antarmuka *messagebox* Tambah Cerobong merupakan antarmuka yang ditampilkan setelah pengguna melakukan *tap* pada peta cerobong untuk menambahkan cerobong baru. *MessageBox* Tambah Cerobong ini berisi borang pengisian data cerobong baru. Setelah pengguna mengisi data cerobong dan menekan tombol ok, aplikasi akan menampilkan halaman Formulir Pengamatan. Implementasi antarmuka *messagebox* Tambah Cerobong ditunjukkan oleh Gambar 4.10.



The image shows a mobile application dialog box with a dark background. At the top, the title "Tambahkan Cerobong" is displayed in white. Below the title, there is a subtitle "Masukkan data cerobong yang akan ditambahkan." (Enter data for the chimney to be added). There are two text input fields: "Nama cerobong" (Chimney name) and "Deskripsi cerobong" (Chimney description). At the bottom of the dialog box, there are two buttons: "ok" and "cancel". The status bar at the top right of the screen shows the time "23:28".

Gambar 4.10 Implementasi *MessageBox* Tambah Cerobong

4.3.11 Implementasi Antarmuka Halaman Formulir Pengamatan

Halaman Formulir Pengamatan merupakan halaman yang ditampilkan setelah pengguna memilih cerobong atau menambahkan cerobong. Halaman ini berisi borang pengisian data pendukung pengamatan. Setelah pengguna mengisi data pengamatan dan menekan tombol Unggah, aplikasi akan menyimpan data pengamatan di *server* dan menampilkan *messagebox* status unggah data pengamatan. Setelah itu aplikasi akan menampilkan halaman Pilihan Menu. Implementasi antarmuka halaman Formulir Pengamatan ditunjukkan oleh Gambar 4.11.



Formulir Pengamatan

Nama pengamat
Fauzi

Waktu mulai
11:00

Waktu akhir
11:15

Alat proses/unit
YK3050

Alat pengamatan emisi
YK3050

Ketinggian terhadap tanah
10

Ketinggian relatif terhadap tanah

Gambar 4.11 Implementasi Halaman Formulir Pengamatan

4.3.12 Implementasi Antarmuka Halaman Rekaman Data

Halaman Rekaman Data merupakan halaman yang ditampilkan setelah pengguna memilih menu Rekaman Data pada halaman Pilihan Menu. Halaman ini berisi data pengamatan yang tersimpan sementara pada basis data lokal. Pengguna dapat mengunggah rekaman data ke *server* dengan menekan tombol Unggah. Aplikasi akan menampilkan *messagebox* yang berisi status unggah rekaman data. Implementasi antarmuka halaman Rekaman Data ditunjukkan oleh Gambar 4.12.



Gambar 4.12 Implementasi Halaman Rekaman Data

4.3.13 Implementasi Antarmuka Halaman Pilih Perusahaan pada Fitur Riwayat

Halaman Pilih Perusahaan pada fitur riwayat merupakan halaman yang ditampilkan setelah pengguna memilih menu Riwayat pada halaman Pilihan Menu. Halaman ini menampilkan daftar perusahaan beserta infonya. Pengguna bisa mencari perusahaan yang akan dilihat riwayat cerobongnya dengan mengetikkan nama perusahaan pada kotak pencarian Cari perusahaan. Jika perusahaan yang cerobongnya diamati terdapat pada daftar, pengguna bisa melakukan *tap* pada kotak info perusahaan. Setelah itu aplikasi akan menampilkan halaman Peta Cerobong. Implementasi antarmuka halaman Pilih Perusahaan pada fitur riwayat ditunjukkan oleh Gambar 4.13.



Gambar 4.13 Implementasi Halaman Pilih Perusahaan pada Fitur Riwayat

4.3.14 Implementasi Antarmuka Halaman Riwayat

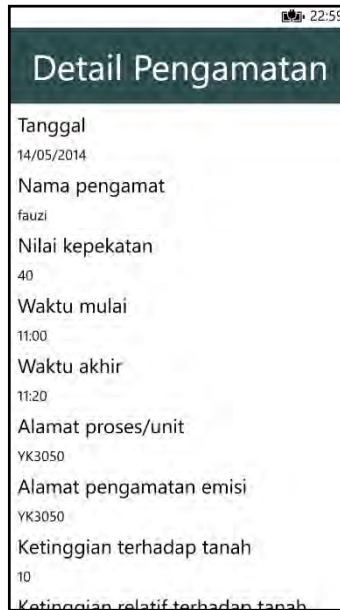
Halaman Riwayat merupakan halaman yang ditampilkan setelah pengguna memilih cerobong pada halaman Peta Cerobong. Halaman ini berisi daftar pengamatan cerobong yang dipilih. Pengguna dapat melihat detail pengamatan dengan melakukan *tap* pada kotak info pengamatan. Pengguna dapat melihat riwayat pengamatan dalam bentuk grafik dengan menekan tombol Lihat Grafik. Implementasi antarmuka halaman Riwayat ditunjukkan oleh Gambar 4.14.



Gambar 4.14 Implementasi Halaman Riwayat

4.3.15 Implementasi Antarmuka Halaman Detail Pengamatan

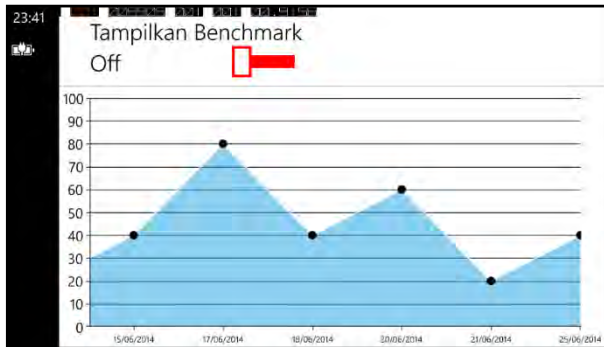
Halaman Detail Pengamatan merupakan halaman yang ditampilkan setelah pengguna melakukan *tap* pada kotak info pengamatan pada halaman Riwayat. Halaman ini berisi info data pendukung pengamatan. Implementasi halaman Detail Pengamatan ditunjukkan oleh gambar Gambar 4.15.



Gambar 4.15 Implementasi Halaman Detail Pengamatan

4.3.16 Implementasi Antarmuka Halaman Grafik Riwayat

Halaman Grafik Riwayat merupakan halaman yang ditampilkan setelah pengguna menekan tombol Lihat Grafik pada halaman Riwayat. Halaman ini berisi grafik riwayat pengamatan asap dari cerobong yang dipilih. Sumbu X pada grafik menunjukkan tanggal pengamatan, sedangkan Sumbu Y menunjukkan nilai kepekatan asap. Pada halaman ini pengguna juga bisa melihat *benchmark* nilai kepekatan asap dengan melakukan *tap* pada *toggle switch* Tampilkan Benchmark. Implementasi antarmuka halaman Grafik Riwayat ditunjukkan oleh Gambar 4.16.



Gambar 4.16 Implementasi Halaman Grafik Riwayat

4.3.17 Implementasi Antarmuka Halaman Benchmark

Halaman Benchmark merupakan halaman yang ditampilkan setelah pengguna memilih menu Benchmark pada halaman Pilihan Menu. Halaman ini berisi *benchmark* nilai kepekatan asap yang ditunjukkan oleh garis merah yang menunjukkan nilai tertinggi, garis kuning yang menunjukkan nilai rata-rata, dan garis hijau yang menunjukkan nilai terendah. Implementasi antarmuka halaman Grafik Riwayat ditunjukkan oleh Gambar 4.17.



Gambar 4.17 Implementasi Halaman *Benchmark*

4.3. Implementasi Kasus Penggunaan

Pada subbab ini dijelaskan bagaimana implementasi dari proses-proses aplikasi yang dapat ditemukan pada bagian perancangan sistem.

4.4.1 Implementasi Kasus Penggunaan Mengukur Tingkat Kepekatan Asap

Implementasi pengukuran tingkat kepekatan asap ditunjukkan oleh Kode Sumber 4.1. Fungsi *CropImage* merupakan fungsi untuk memotong citra menjadi berukuran 512×512 *pixel*. Fungsi *ConvertToWbmp* merupakan fungsi untuk mengubah skala Ringelmann Smoke Chart berupa *.png* menjadi bertipe *Writeable Bitmap*. Fungsi *getQuant* merupakan fungsi untuk menormalisasikan histogram RGB ke dalam *array* dengan panjang 16. Fungsi *GetCumulativeHistogram* merupakan fungsi untuk

menggabungkan masing komponen RGB ke dalam 1 *array*. Fungsi *GetEuclideanDistance* merupakan fungsi untuk menghitung *euclidean distance* antara 2 histogram RGB. Fungsi *GetOpacity* merupakan fungsi untuk mendapatkan nilai kepekatan asap berdasarkan *euclidean distance* yang terkecil.

```

1. private const int QUANT = 16;
2.
3. public static void GetOpacity(WriteableBitmap
   ImageSource)
4. {
5.     DataSingleton.Instance.ChartHistogram = new int[5][];
6.     DataSingleton.Instance.ChartHistogram[0] =
   OpacityMeasurement.GetCumulativeHistogram(OpacityMeasurem
   ent.ConvertToWbmp("20.png"));
7.     DataSingleton.Instance.ChartHistogram[1] =
   OpacityMeasurement.GetCumulativeHistogram(OpacityMeasurem
   ent.ConvertToWbmp("40.png"));
8.     DataSingleton.Instance.ChartHistogram[2] =
   OpacityMeasurement.GetCumulativeHistogram(OpacityMeasurem
   ent.ConvertToWbmp("60.png"));
9.     DataSingleton.Instance.ChartHistogram[3] =
   OpacityMeasurement.GetCumulativeHistogram(OpacityMeasurem
   ent.ConvertToWbmp("80.png"));
10.    DataSingleton.Instance.ChartHistogram[4] =
   OpacityMeasurement.GetCumulativeHistogram(OpacityMeasurem
   ent.ConvertToWbmp("100.png"));
11.    int[] chart = { 20, 40, 60, 80, 100 };
12.
13.    int[] smokeImage =
   GetCumulativeHistogram(CropImage(ImageSource, 768, 512,
   512, 512));
14.
15.    double min = Double.MaxValue;
16.    int index = -1;
17.    for (int i = 0; i < 5; i++)
18.    {
19.        double has =
   GetEuclideanDistance(DataSingleton.Instance.ChartHistogram[i],
   smokeImage);
20.        if (has < min)
21.        {
22.            min = has;
23.            index = i;
24.        }
25.    }

```

```

26.     DataSingleton.Instance.RingelmannOpacity =
    chart[index];
27. }
28. private static WriteableBitmap CropImage(WriteableBitmap
    source,      int xOffset, int yOffset, int width, int
    height)
29. {
30.     var sourceWidth = source.PixelWidth;
31.     var result = new WriteableBitmap(width, height);
32.     for (var x = 0; x <= height - 1; x++)
33.     {
34.         var sourceIndex = xOffset + (yOffset + x) *
    sourceWidth;
35.         var destinationIndex = x * width;
36.         Array.Copy(source.Pixels, sourceIndex,
    result.Pixels, destinationIndex, width);
37.     }
38.
39.     DataSingleton.Instance.Pic = result;
40.     return result;
41. }
42.
43. public static WriteableBitmap ConvertToWbmp(string
    filename)
44. {
45.     StreamResourceInfo sri =
    Application.GetResourceStream(new Uri("Images/"+filename,
    UriKind.Relative));
46.     BitmapImage src = new BitmapImage();
47.     src.SetSource(sri.Stream);
48.     WriteableBitmap bitmapImage = new
    WriteableBitmap(src);
49.     return bitmapImage;
50. }
51.
52. public static int[]
    GetCumulativeHistogram(WriteableBitmap bitmapImage)
53. {
54.     int[][] arr = getQuant(bitmapImage);
55.     int[] r = new int[QUANT];
56.     int[] g = new int[QUANT];
57.     int[] b = new int[QUANT];
58.     for (int j = 0; j < arr[0].Length; j++)
59.     {
60.         r[arr[0][j]]++;
61.         g[arr[1][j]]++;
62.         b[arr[2][j]]++;
63.     }

```

```

64.     int[] output = new int[r.Length + g.Length +
        b.Length];
65.     r.CopyTo(output, 0);
66.     g.CopyTo(output, r.Length);
67.     b.CopyTo(output, r.Length + g.Length);
68.     return output;
69. }
70.
71. private static int[][] getQuant(WritableBitmap img)
72. {
73.     int[][] feature = new int[3][];
74.     int[] r = new int[img.Pixels.Length];
75.     int[] g = new int[img.Pixels.Length];
76.     int[] b = new int[img.Pixels.Length];
77.     for (int i = 0; i < img.Pixels.Length; i++)
78.     {
79.         var pixelbytes =
            BitConverter.GetBytes(img.Pixels[i]);
80.         r[i] = pixelbytes[0] * (QUANT - 1) / 255;
81.         g[i] = pixelbytes[1] * (QUANT - 1) / 255;
82.         b[i] = pixelbytes[2] * (QUANT - 1) / 255;
83.     }
84.
85.     feature[0] = r;
86.     feature[1] = g;
87.     feature[2] = b;
88.     return feature;
89. }
90.
91. private static double GetEuclideanDistance(int[] array1,
        int[] array2)
92. {
93.     double Sum = 0.0;
94.     for (int i = 0; i < array1.Length; i++)
95.     {
96.         Sum = Sum + Math.Pow((array1[i] - array2[i]),
            2.0);
97.     }
98.     return Math.Sqrt(Sum);
99. }

```

Kode Sumber 4.1 Sintaks untuk Mengukur Tingkat Kepekatan Asap

4.4.2 Implementasi Kasus Penggunaan Mengunggah Rekaman Data ke *Server*

Implementasi mengunggah rekaman data ke server ditunjukkan oleh Kode Sumber 4.2. Fungsi *Draft* merupakan fungsi *main* untuk

menjalankan memanggil fungsi *LoadMeasurement* yang berisi sintaks untuk mendapatkan *list* rekaman data dari basis data lokal. Fungsi *UploadMeasurement* merupakan fungsi untuk mengunggah rekaman data ke *server*. Fungsi *AddMeasurementCompleted* merupakan fungsi yang dijalankan ketika proses unggah rekaman data telah dilakukan, berisi sintaks untuk menampilkan pesan status unggah. Fungsi *DeleteDraft* merupakan fungsi untuk menghapus rekaman data pada basis data lokal yang telah singgah ke *server*.

```

1. private MeasurementDBDataContext
   _measurementDBDataContext;
2. private ObservableCollection<MeasurementDB>
   listMeasurementDb;
3.
4. public Draft()
5. {
6.     InitializeComponent();
7.     listMeasurementDb = new
   ObservableCollection<MeasurementDB>();
8.     _measurementDBDataContext = new
   MeasurementDBDataContext (MeasurementDBDataContext.DBConne
   ctionString);
9.     LoadMeasurement();
10. }
11.
12. private void LoadMeasurement()
13. {
14.     var measurementDBs = from MeasurementDB m in
   _measurementDBDataContext.MeasurementTable select m;
15.     foreach (var measurement in measurementDBs)
16.     {
17.         listMeasurementDb.Add(measurement);
18.     }
19.
20.     if (listMeasurementDb.Count > 0)
21.     {
22.         listDraft.ItemsSource = listMeasurementDb;
23.         btnUpload.IsEnabled = true;
24.     }
25.     else
26.     {
27.         txtEmpty.Visibility = Visibility.Visible;
28.     }
29. }
30.

```

```

31. private void UploadMeasurement(MeasurementDB m)
32. {
33.     SmokeCalcClient client = new SmokeCalcClient();
34.     client.AddMeasurementCompleted +=
AddMeasurementCompleted;
35.     client.AddMeasurementAsync(m.ChimneyId, m.UserId,
m.MeasurementObserver, m.MeasurementDate,
m.MeasurementTime1,
36.         m.MeasurementTime2,
m.MeasurementResultRingelmann, m.MeasurementUnitAddr,
37.         m.MeasurementEmissionAddr,
m.MeasurementHeight, m.MeasurementRelativeHeight,
m.MeasurementDistance,
38.         m.MeasurementDir1, m.MeasurementDir2,
m.MeasurementEmissionDes1, m.MeasurementEmissionDes2,
39.         m.MeasurementEmissionColor1,
m.MeasurementEmissionColor2, m.MeasurementPoint1,
m.MeasurementPoint2,
40.         m.MeasurementBgDes1, m.MeasurementBgDes2,
m.MeasurementBgColor1, m.MeasurementBgColor2,
m.MeasurementSky1,
41.         m.MeasurementSky2, m.MeasurementWind1,
m.MeasurementWind2, m.MeasurementWindDir1,
m.MeasurementWindDir2,
42.         m.MeasurementTemp1, m.MeasurementTemp2,
m.MeasurementWetTemp, m.MeasurementRh);
43. }
44.
45. private void AddMeasurementCompleted(object sender,
AddMeasurementCompletedEventArgs e)
46. {
47.     if (e.Error != null)
48.     {
49.         MessageBox.Show("Tidak dapat melakukan
koneksi ke server.\r\nCoba periksa koneksi internet
anda.");
50.         NavigationService.Navigate(new
Uri("/Menu.xaml", UriKind.Relative));
51.     }
52.     else
53.     {
54.         MessageBox.Show(e.Result);
55.     }
56. }
57.
58. private void UploadDraft(object sender, RoutedEventArgs
e)
59. {

```

```

60.     var measurementDBs = from MeasurementDB m in
61.     _measurementDBDataContext.MeasurementTable select m;
62.     foreach (var measurement in measurementDBs)
63.     {
64.         UploadMeasurement(measurement);
65.         DeleteDraft(measurement);
66.     }
67.     MessageBox.Show("Data formulir pengamatan telah
        diunggah ke server.");
68.     NavigationService.Navigate(new Uri("/Menu.xaml",
        UriKind.Relative));
69. }
70. private void DeleteDraft(MeasurementDB measurement)
71. {
72.     _measurementDBDataContext.MeasurementTable.DeleteOnSu
        bmit(measurement);
73.     _measurementDBDataContext.SubmitChanges();
74. }

```

Kode Sumber 4.2 Sintaks untuk Mengunggah Rekaman Data ke Server

4.4.3 Implementasi Kasus Penggunaan Menampilkan Riwayat Pengukuran

Implementasi menampilkan riwayat pengukuran ditunjukkan oleh Kode Sumber 4.3. Fungsi *History* merupakan fungsi *main* untuk menampilkan riwayat pengukuran. Fungsi *GetHistory* merupakan fungsi untuk mendapatkan *list* data pengukuran dari *server*. Fungsi *GetMeasurementListCompleted* merupakan fungsi yang dipanggil setelah proses *GetHistory* telah dijalankan, berisi sintaks untuk mengisi grafik sesuai data pengukuran.

```

1. public History()
2. {
3.     InitializeComponent();
4.     listHistory.SelectedIndex = -1;
5.     listHistory.SetValue(TiltEffect.IsTiltEnabledProperty
        , true);
6.     if (listHistory.Items.Count == 0)
7.         busyIndicator.IsRunning = true;
8.     GetHistory();
9. }
10. private void GetHistory()
11. {

```



```

12.     SmokeCalcClient client = new SmokeCalcClient();
13.     client.GetMeasurementListCompleted +=
    GetMeasurementListCompleted;
14.     client.GetMeasurementListAsync("history",
    DataSingleton.Instance.ChimneyId);
15. }
16.
17. private void GetMeasurementListCompleted(object sender,
    GetMeasurementListCompletedEventArgs e)
18. {
19.     busyIndicator.IsRunning = false;
20.     Thread.Sleep(500);
21.     DataSingleton.Instance.MeasurementList = e.Result;
22.     if (e.Result.Count == 0)
23.     {
24.         MessageBox.Show("Tidak terdapat histori
    pengamatan.");
25.     }
26.     else
27.     {
28.         listHistory.ItemsSource = e.Result;
29.         btnViewChart.Visibility = Visibility.Visible;
30.     }
31. }
32.
33. private void listHistory_SelectionChanged(object sender,
    SelectionChangedEventArgs e)
34. {
35.     if (listHistory.SelectedIndex >= 0)
36.     {
37.         Measurement selectedMeasurement =
    listHistory.SelectedItem as Measurement;
38.         DataSingleton.Instance.MeasurementId =
    selectedMeasurement.MeasurementId;
39.         NavigationService.Navigate(new
    Uri("/MeasurementDetail.xaml", UriKind.Relative));
40.     }
41. }

```

Kode Sumber 4.3 Sintaks untuk Menampilkan Riwayat Pengukuran

4.4.4 Implementasi Kasus Penggunaan Menampilkan *Benchmark* Nilai Kepekatan Asap

Implementasi menampilkan riwayat pengukuran ditunjukkan oleh Kode Sumber 4.4. Fungsi *Benchmark* merupakan fungsi *main* untuk menampilkan *benchmark*. Fungsi *GetBenchmark* merupakan

fungsi untuk mendapatkan nilai *benchmark* dari *server*. Fungsi *GetBenchmarkCompleted* merupakan fungsi yang dipanggil setelah proses *GetBenchmark* telah dijalankan, berisi sintaks untuk memanggil fungsi *SetBenchmarkValue* yang berfungsi untuk menampilkan *benchmark* pada *chart*.

```
1. private double[] value;
2.
3. public Benchmark()
4. {
5.     GetBenchmark();
6. }
7.
8. private void GetBenchmark()
9. {
10.    SmokeCalcClient client = new SmokeCalcClient();
11.    client.GetBenchmarkCompleted +=
    GetBenchmarkCompleted;
12.    client.GetBenchmarkAsync();
13. }
14.
15. private void SetBenchmarkValue(double[] value)
16. {
17.    double topHighest =
    CalculateTopMargin(chart.radChart.ActualHeight,
    value[0]);
18.    chart.lineHighest.Margin = new Thickness(43, 12 +
    topHighest, 0, 0);
19.    chart.lineHighest.Visibility = Visibility.Visible;
20.
21.    double topAverage =
    CalculateTopMargin(chart.radChart.ActualHeight,
    value[2]);
22.    chart.lineAverage.Margin = new Thickness(43, 12 +
    topAverage, 0, 0);
23.    chart.lineAverage.Visibility = Visibility.Visible;
24.
25.    double topLowest =
    CalculateTopMargin(chart.radChart.ActualHeight,
    value[1]);
26.    chart.lineLowest.Margin = new Thickness(43, 12 +
    topLowest, 0, 0);
27.    chart.lineLowest.Visibility = Visibility.Visible;
28. }
29.
30. private double CalculateTopMargin(double height, double
    val)
```

```
31. {
32.     return (height - 23.0) / 100.0 * (100.0 - val);
33. }
34.
35. private void GetBenchmarkCompleted(object sender,
    GetOpacityStatisticCompletedEventArgs e)
36. {
37.     if (e.Error != null)
38.     {
39.         MessageBox.Show("Tidak dapat melakukan
    koneksi ke server.\r\nCoba periksa koneksi internet
    anda.");
40.         NavigationService.Navigate(new
    Uri("/Menu.xaml", UriKind.Relative));
41.     }
42.     else if(e.Result != null)
43.     {
44.         value = new double[]{e.Result[0],
    e.Result[1], e.Result[2]};
45.         SetBenchmarkValue(value);
46.     }
47. }
```

**Kode Sumber 4.4 Sintaks untuk Menampilkan Benchmark
Nilai Kepekatan Asap**

BAB V

PENGUJIAN DAN EVALUASI

Bab ini membahas rangkaian pengujian dan evaluasi terhadap perangkat lunak yang dibangun pada Tugas Akhir ini. Pengujian dilakukan secara menyeluruh menggunakan metode kotak hitam untuk mengecek apakah fungsi berjalan sesuai dengan yang diinginkan. Pengujian dilaksanakan dengan berbagai skenario uji. Bab ini mencakup lingkungan, skenario, hasil, serta evaluasi terhadap pengujian.

5.1 Lingkungan Pengujian

Pengujian terhadap aplikasi dilakukan menggunakan *smartphone* dengan spesifikasi sebagai berikut:

- Prosesor: Qualcomm Snapdragon S4 Dual-core 1.7 GHz (MSM8930)
- Memori: 1 GB
- Kartu Grafis: Adreno 305
- Sistem Operasi: Windows Phone 8.0

Pengambilan gambar asap dilakukan dengan memenuhi syarat-syarat berikut ini:

1. Jarak antara pengamat dan cerobong asap sejauh 3 kali tinggi cerobong.
2. Latar belakang langit biru.
3. Posisi matahari di belakang pengamat.
4. Arah angin pada sudut 90° .
5. Tidak ada halangan yang mempengaruhi pengamatan.

5.2 Skenario Pengujian

Pada bagian ini akan dijelaskan tentang skenario pengujian yang dilakukan. Pengujian yang dilakukan adalah pengujian kebutuhan fungsionalitas. Pengujian fungsionalitas menggunakan metode *black box*. Metode ini menekankan pada kesesuaian hasil keluaran sistem.

Pengujian terhadap hasil pengukuran tingkat kepekatan asap dilakukan dengan membandingkan hasil pengukuran oleh sistem dengan hasil pengukuran manual yang dilakukan oleh pengamat ahli. Dengan membandingkan hasil pengukuran tingkat kepekatan oleh sistem dan pengukuran secara manual akan didapatkan kesimpulan terhadap hasil pengukuran tingkat kepekatan asap oleh sistem.

5.2.1 Pengujian Kasus Penggunaan

Pengujian fungsionalitas sistem dilakukan dengan menyiapkan sejumlah skenario sebagai tolak ukur keberhasilan pengujian. Pengujian fungsionalitas dilakukan dengan mengacu pada kasus penggunaan yang telah dijelaskan pada subbab 3.1.3.2. Pengujian pada kebutuhan fungsionalitas dapat dijabarkan pada subbab berikut.

5.2.1.1 Pengujian Fitur Mengukur Tingkat Kepekatan Asap

Pengujian fitur mengukur tingkat kepekatan asap dilakukan untuk menguji apakah aplikasi dapat mengukur tingkat kepekatan asap. Terdapat dua skenario pengujian untuk mengukur tingkat kepekatan asap. Skenario pengujian untuk skenario pertama dapat dilihat pada Tabel 5.1. Skenario pengujian untuk skenario pertama dapat dilihat pada

Tabel 5.2. Skenario pengujian untuk skenario pertama dapat dilihat pada Tabel 5.3. Hasil tampilan pada program untuk pengujian pertama dapat dilihat pada Gambar 5.1, untuk pengujian kedua dapat dilihat pada Gambar 5.2, dan untuk pengujian ketiga dapat dilihat pada Gambar 5.3.

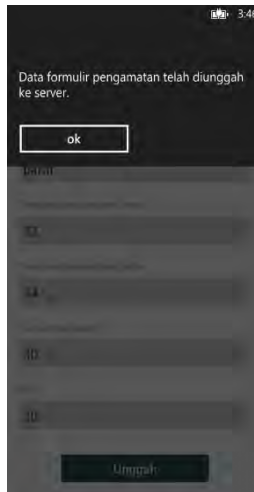
Tabel 5.1 Pengujian Mengukur Tingkat Kepekatan Asap Skenario Pertama

ID	UJ.UC-0001
Referensi Kasus Penggunaan	UC-0001
Nama	Pengujian fitur mengukur tingkat kepekatan asap
Tujuan Pengujian	Menguji fitur untuk mengukur tingkat kepekatan asap sampai dengan proses mengunggah data pengamatannya.
Skenario 1	Memilih perusahaan dan cerobong yang sudah terdaftar pada <i>server</i> sebelum mengisi formulir data pengamatan.
Kondisi Awal	Pengguna berada pada halaman pilihan menu aplikasi.
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih menu pengukuran. 2. Pengguna memfokuskan asap pada kotak merah pada antarmuka kamera dan menekan tombol Ambil Gambar. 3. Pengguna menekan Lengkapi data pengamatan. 4. Pengguna memilih perusahaan dengan melakukan <i>tap</i> pada kotak info perusahaan. 5. Pengguna memilih cerobong dengan melakukan <i>tap</i> pada <i>pushpin</i> cerobong pada <i>map</i>. 6. Pengguna menekan tombol Ok pada <i>messagebox</i> konfirmasi memilih cerobong. 7. Pengguna mengisi formulir data pengamatan. 8. Pengguna menekan tombol Unggah.
Hasil Yang Diharapkan	Muncul <i>messagebox</i> yang menampilkan pesan berhasil mengunggah data pengamatan.
Hasil Yang Didapat	Muncul <i>messagebox</i> yang menampilkan pesan berhasil mengunggah data pengamatan seperti yang terlihat pada Gambar 5.1.
Hasil Pengujian	Berhasil

**Tabel 5.2 Pengujian Mengukur Tingkat Kepekatan Asap
Skenario Kedua**

ID	UJ.UC-0001
Referensi Kasus Penggunaan	UC-0001
Nama	Pengujian fitur mengukur tingkat kepekatan asap
Tujuan Pengujian	Menguji fitur untuk mengukur tingkat kepekatan asap sampai dengan proses mengunggah data pengamatannya.
Skenario 2	Menambahkan perusahaan dan cerobong yang belum terdaftar pada <i>server</i> sebelum mengisi formulir data pengamatan.
Kondisi Awal	Pengguna berada pada halaman pilihan menu aplikasi.
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih menu pengukuran. 2. Pengguna memfokuskan asap pada kotak merah pada antarmuka kamera dan menekan tombol Ambil Gambar. 3. Pengguna menekan Lengkapi data pengamatan. 4. Pengguna menambahkan perusahaan baru dengan menekan <i>application bar</i> Tambah. 5. Pengguna mengisi data perusahaan. 6. Pengguna menekan tombol Tambahkan Perusahaan. 7. Pengguna menambahkan cerobong dengan melakukan <i>tap</i> pada lokasi yang belum terdapat <i>pushpin</i> pada <i>map</i>. 8. Pengguna mengisi data cerobong. 9. Pengguna menekan tombol Ok. 10. Pengguna mengisi formulir data pengamatan. 11. Pengguna menekan tombol Unggah.
Hasil Yang Diharapkan	Muncul <i>messagebox</i> yang menampilkan pesan berhasil mengunggah data pengamatan.

Hasil Yang Didapat	Muncul <i>messagebox</i> yang menampilkan pesan berhasil mengunggah data pengamatan seperti yang terlihat pada Gambar 5.2.
Hasil Pengujian	Berhasil



Gambar 5.1 Hasil Pengujian Fitur Mengukur Tingkat Kepekatan Asap Untuk Skenario Pertama

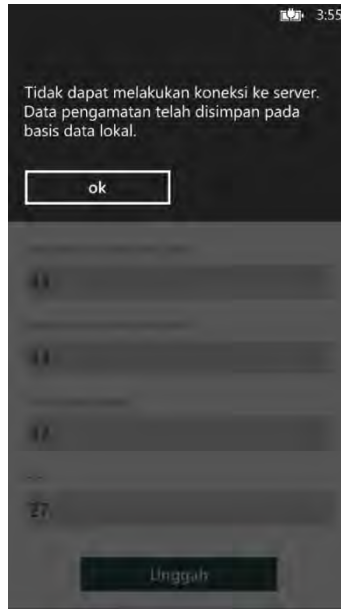


Gambar 5.2 Hasil Pengujian Fitur Mengukur Tingkat Kepekatan Asap Untuk Skenario Kedua

Tabel 5.3 Pengujian Mengukur Tingkat Kepekatan Asap Skenario Ketiga

ID	UJ.UC-0001
Referensi Kasus Penggunaan	UC-0001
Nama	Pengujian fitur mengukur tingkat kepekatan asap
Tujuan Pengujian	Menguji fitur untuk mengukur tingkat kepekatan asap sampai dengan proses mengunggah data pengamatannya.
Skenario 3	Memilih perusahaan dan cerobong yang sudah terdaftar pada <i>server</i> dan mengisi formulir data pengamatan pada kondisi koneksi internet tidak tersambung.

Kondisi Awal	Pengguna berada pada halaman pilihan menu aplikasi.
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih menu pengukuran. 2. Pengguna memfokuskan asap pada kotak merah pada antarmuka kamera dan menekan tombol Ambil Gambar. 3. Pengguna menekan Lengkapi data pengamatan. 4. Pengguna memilih perusahaan dengan melakukan <i>tap</i> pada kotak info perusahaan. 5. Pengguna memilih cerobong dengan melakukan <i>tap</i> pada <i>pushpin</i> cerobong pada <i>map</i>. 6. Pengguna menekan tombol Ok pada <i>messagebox</i> konfirmasi memilih cerobong. 7. Pengguna mengisi formulir data pengamatan. 8. Koneksi internet pada <i>smartphone</i> dimatikan dengan melakukan <i>minimize</i> aplikasi. 9. Pengguna membuka kembali aplikasi. 10. Pengguna menekan tombol Unggah.
Hasil Yang Diharapkan	Muncul <i>messagebox</i> yang menampilkan pesan tidak tersambung ke <i>server</i> dan data pengamatan disimpan pada basis data lokal.
Hasil Yang Didapat	Muncul <i>messagebox</i> yang menampilkan pesan tidak tersambung ke <i>server</i> dan data pengamatan disimpan pada basis data lokal seperti yang terlihat pada Gambar 5.3.
Hasil Pengujian	Berhasil



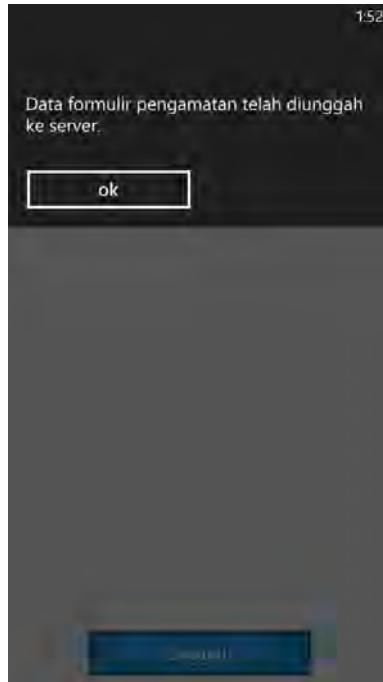
Gambar 5.3 Hasil Pengujian Fitur Mengukur Tingkat Kepekatan Asap Untuk Skenario Ketiga

5.2.1.2 Pengujian Fitur Mengunggah Rekaman Data ke *Server*

Pengujian fitur mengunggah rekaman data ke *server* bertujuan untuk mengecek apakah aplikasi dapat mengunggah rekaman data pengamatan yang tersimpan pada basis data lokal ke *server*. Terdapat satu skenario pengujian untuk mengukur tingkat kepekatan asap. Skenario pengujian dapat dilihat pada Tabel 5.4. Hasil tampilan pada pengujian dapat dilihat pada Gambar 5.4.

Tabel 5.4 Skenario Pengujian Mengunggah Rekaman Data ke *Server*

ID	UJ.UC-0002
Referensi Kasus Penggunaan	UC-0002
Nama	Pengujian fitur mengunggah rekaman data ke <i>server</i> .
Tujuan Pengujian	Menguji fitur untuk mengunggah rekaman data pengamatan ke <i>server</i> .
Skenario	Mengunggah rekaman data ke <i>server</i>.
Kondisi Awal	Pengguna berada pada halaman pilihan menu aplikasi dan terdapat 1 rekaman data pada basis data lokal.
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih menu rekaman data. 2. Pengguna menekan tombol Unggah.
Hasil Yang Diharapkan	Muncul <i>messagebox</i> yang menampilkan pesan berhasil mengunggah rekaman data.
Hasil Yang Didapat	Muncul <i>messagebox</i> yang menampilkan pesan berhasil mengunggah rekaman data seperti yang terlihat pada Gambar 5.4.
Hasil Pengujian	Berhasil



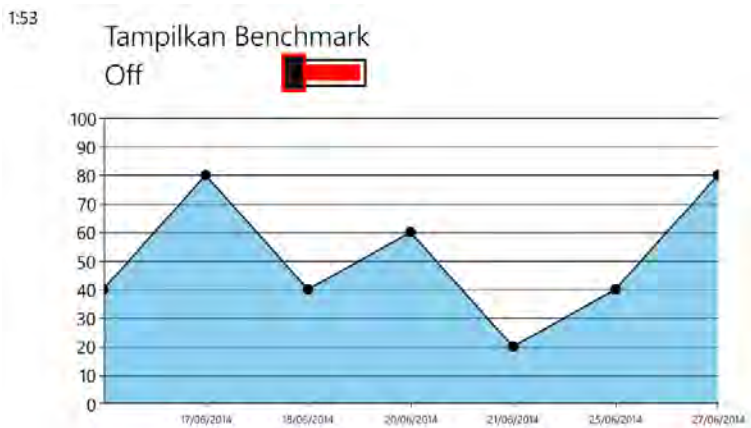
Gambar 5.4 Hasil Pengujian Fitur Mengunggah Rekaman Data ke *Server*

5.2.1.3 Pengujian Fitur Menampilkan Riwayat Pengukuran

Pengujian fitur menampilkan riwayat pengukuran bertujuan untuk menguji apakah aplikasi dapat menampilkan riwayat pengukuran tingkat kepekatan asap. Terdapat satu skenario pengujian untuk mengukur tingkat kepekatan asap. Skenario pengujian dapat dilihat pada Tabel 5.5. Hasil tampilan pada pengujian dapat dilihat pada Gambar 5.5.

Tabel 5.5 Skenario Pengujian Menampilkan Riwayat Pengukuran

ID	UJ.UC-0003
Referensi Kasus Penggunaan	UC-0003
Nama	Pengujian fitur menampilkan riwayat pengukuran
Tujuan Pengujian	Menguji fitur untuk menampilkan riwayat pengukuran.
Skenario	Menampilkan riwayat pengukuran dari cerobong yang dipilih.
Kondisi Awal	Pengguna berada pada halaman pilihan menu aplikasi.
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengguna memilih menu riwayat. 2. Pengguna memilih perusahaan dengan melakukan <i>tap</i> pada kotak info perusahaan. 3. Pengguna memilih cerobong dengan melakukan <i>tap</i> pada <i>pushpin</i> cerobong pada <i>map</i>. 4. Pengguna menekan tombol Ok pada <i>messagebox</i> konfirmasi memilih cerobong. 5. Pengguna menekan tombol Lihat Grafik.
Hasil Yang Diharapkan	Muncul grafik riwayat pengamatan dari cerobong yang dipilih.
Hasil Yang Didapat	Muncul grafik riwayat pengamatan dari cerobong yang dipilih seperti yang terlihat pada Gambar 5.5.
Hasil Pengujian	Berhasil



Gambar 5.5 Hasil Pengujian Fitur Menampilkan Riwayat Pengukuran

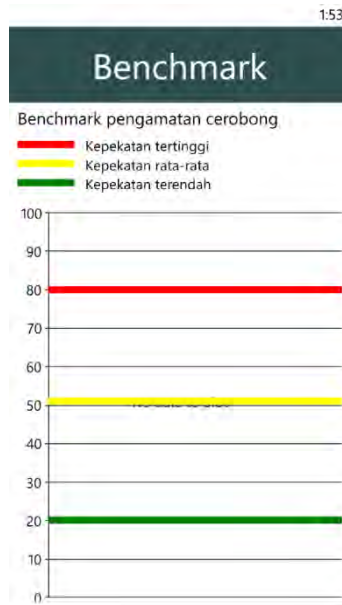
5.2.1.4 Pengujian Fitur Menampilkan *Benchmark* Nilai Kepekatan Asap

Pengujian fitur menampilkan *benchmark* nilai kepekatan asap bertujuan untuk menguji apakah aplikasi dapat menampilkan *benchmark* nilai kepekatan asap. Terdapat satu skenario pengujian untuk mengukur tingkat kepekatan asap. Skenario pengujian dapat dilihat pada Tabel 5.6. Hasil tampilan pada pengujian dapat dilihat pada Gambar 5.6 Hasil Pengujian Fitur Menampilkan *Benchmark* Nilai Kepekatan Asap.

Tabel 5.6 Skenario Pengujian Menampilkan *Benchmark* Nilai Kepekatan Asap

ID	UJ.UC-0004
Referensi Kasus Penggunaan	UC-0004
Nama	Pengujian fitur menampilkan <i>benchmark</i> nilai kepekatan asap.

Tujuan Pengujian	Menguji fitur untuk menampilkan <i>benchmark</i> nilai kepekatan asap.
Skenario	Menampilkan <i>benchmark</i> nilai kepekatan asap.
Kondisi Awal	Pengguna berada pada halaman pilihan menu aplikasi.
Langkah Pengujian	1. Pengguna memilih menu <i>benchmark</i> .
Hasil Yang Diharapkan	Muncul <i>benchmark</i> nilai kepekatan asap yang terdiri dari nilai kepekatan tertinggi, nilai kepekatan rata-rata, dan nilai kepekatan terendah.
Hasil Yang Didapat	Muncul grafik riwayat pengamatan dari cerobong yang dipilih seperti yang terlihat pada Gambar 5.5.
Hasil Pengujian	Berhasil



Gambar 5.6 Hasil Pengujian Fitur Menampilkan *Benchmark* Nilai Kepekatan Asap

5.2.2 Pengujian Kegunaan & Kemudahan Aplikasi

Pengujian kegunaan dan kemudahan aplikasi dilakukan dengan menyiapkan kuesioner yang ditunjukkan oleh Tabel 5.7. Pengujian ini dilakukan dengan mengacu pada hasil dari respon pengguna terhadap aplikasi dengan menggunakan skala 1-5 untuk nilai kesesuaian dengan pertanyaan. Angka 1 menunjukkan bahwa responden sangat tidak setuju. Angka 2 menunjukkan bahwa responden tidak setuju. Angka 3 menunjukkan bahwa responden netral. Angka 4 menunjukkan bahwa responden setuju. Angka 5 menunjukkan bahwa responden sangat setuju. Hasil keseluruhan dari kuesioner yang diisi responden dilampirkan pada Lampiran C. Hasil Kuesioner.

Tabel 5.7 Kuesioner Pendapat Pengguna

No	Pertanyaan Kegunaan Aplikasi	Penilaian				
		1	2	3	4	5
1.	Pengukuran tingkat kepekatan asap dapat dilakukan dengan mudah menggunakan aplikasi ini.					
2.	Pengisian data pengamatan melalui aplikasi ini lebih mudah dan cepat dibandingkan dengan pengisian secara manual pada kertas laporan pengamatan.					
3.	Fitur rekaman data membantu pengguna untuk tidak mengulang proses pengukuran dan pengisian data pengamatan, apabila koneksi internet sedang bermasalah.					
4.	Fitur histori / riwayat mempermudah pengguna untuk <i>tracking</i> perkembangan kepekatan asap cerobong.					
5.	<i>Benchmark</i> dapat ditampilkan dengan baik pada aplikasi ini.					

Hasil Persentase didapatkan dari jumlah nilai yang diberikan terhadap pertanyaan dikalikan dengan jumlah responden yang menjawab pada nilai lalu dikalikan dengan 100 dan dibagi nilai maksimal. Contoh:

- Persentase dari tampilan yang menarik.
 - Jumlah responden keseluruhan = 10 orang.
 - Jumlah responden yang menilai 1 = 0 orang.
 - Jumlah responden yang menilai 2 = 0 orang.
 - Jumlah responden yang menilai 3 = 0 orang.
 - Jumlah responden yang menilai 4 = 2 orang.
 - Jumlah responden yang menilai 5 = 8 orang.
$$\text{Persentase} = ((1 \times 0) + (2 \times 0) + (3 \times 0) + (4 \times 2) + (5 \times 8)) \times 100 / 50$$

$$\text{Persentase} = 96\%$$

5.3 Evaluasi Pengujian

Pada subbab ini akan diberikan hasil evaluasi dari pengujian-pengujian yang telah dilakukan. Evaluasi yang diberikan meliputi evaluasi pengujian kebutuhan fungsional, evaluasi pengujian kegunaan & kemudahan, dan evaluasi hasil pengukuran tingkat kepekatan asap.

5.3.1 Evaluasi Pengujian Fungsionalitas

Rangkuman mengenai hasil pengujian fungsionalitas dapat dilihat pada Tabel 5.8. Berdasarkan data pada tabel tersebut, semua skenario pengujian berhasil dan program berjalan dengan baik. Sehingga bisa ditarik disimpulkan bahwa fungsionalitas dari program telah bisa bekerja sesuai dengan yang diharapkan.

Tabel 5.8 Rangkuman Hasil Pengujian

ID	Nama	Skenario	Hasil
UJ.UC-0001	Pengujian fitur mengukur tingkat kepekatan asap	Skenario 1	Berhasil
		Skenario 2	Berhasil
		Skenario 3	Berhasil
UJ.UC-0002	Pengujian fitur mengunggah rekaman data ke <i>server</i>	Skenario 1	Berhasil

ID	Nama	Skenario	Hasil
UJ.UC-0003	Pengujian fitur menampilkan riwayat pengukuran	Skenario 1	Berhasil
UJ.UC-0004	Pengujian fitur menampilkan <i>benchmark</i> nilai kepekatan asap	Skenario 1	Berhasil

5.3.2 Evaluasi Pengujian Kegunaan & Kemudahan

Berdasarkan hasil dari kuesioner yang diisi oleh pengguna maka didapatkan kesimpulan bahwa aplikasi ini cukup membantu pengguna dalam mengukur tingkat kepekatan asap, mencatat data pengamatan, melihat riwayat pengamatan, dan melihat *benchmark* kepekatan asap. Persentase persentase dari hasil pengujian kegunaan dapat dilihat pada Tabel 5.9.

Tabel 5.9 Hasil Pengujian Kegunaan & Kemudahan

No.	Kegunaan	Persentase
1.	Pengukuran tingkat kepekatan asap dapat dilakukan dengan mudah menggunakan aplikasi ini.	80%
2.	Pengisian data pengamatan melalui aplikasi ini lebih mudah dan cepat dibandingkan dengan pengisian secara manual pada kertas laporan pengamatan.	88%
3.	Fitur rekaman data membantu pengguna untuk tidak mengulang proses pengukuran dan pengisian data pengamatan, apabila koneksi internet sedang bermasalah.	88%

4.	Fitur histori / riwayat mempermudah pengguna untuk <i>tracking</i> perkembangan kepekatan asap cerobong.	87%
5.	<i>Benchmark</i> dapat ditampilkan dengan baik pada aplikasi ini.	81%
Rata-rata		85%

5.3.3 Evaluasi Hasil Pengukuran Tingkat Kepekatan Asap

Evaluasi hasil pengukuran tingkat kepekatan asap dilakukan dengan membandingkan hasil pengukuran yang dilakukan oleh aplikasi dengan hasil pengukuran yang dilakukan dengan membandingkan antara foto asap dan skala pada Ringelmann Smoke Chart secara manual. Pengukuran secara manual dilakukan oleh 5 pengamat. Dari masing-masing pengukuran oleh pengamat didapatkan akurasi dengan membagi jumlah foto yang memiliki hasil yang sama antara pengukuran aplikasi dan pengukuran manual dengan jumlah foto yang diuji. Dari akurasi tersebut dirata-rata dan didapatkan hasil akurasinya adalah 72%. Dengan nilai akurasi tersebut, dapat dikatakan aplikasi dapat mengimplementasikan metode Image Retrieval Based on Color Histogram dengan baik, di mana akurasi hasil pengujian metode tersebut pada paper sebesar 66% [7]. Hasil akurasi pengukuran aplikasi dibandingkan dengan pengukuran secara manual ditunjukkan oleh Tabel 5.10.

Tabel 5.10 Akurasi Pengukuran Oleh Aplikasi

Pengamat	Jumlah Pengukuran yang Sama dengan Aplikasi	Jumlah Foto	Akurasi
Pengamat 1	30	42	71%
Pengamat 2	31	42	74%
Pengamat 3	31	42	74%
Pengamat 4	30	42	71%
Pengamat 5	30	42	71%
Rata-rata			72%

Tabel perbandingan antara hasil pengukuran oleh aplikasi dan hasil pengukuran secara manual secara lengkap dilampirkan pada Tabel 11.1 Lampiran D. Hasil Pengukuran.

BAB VI PENUTUP

Bab ini membahas kesimpulan yang diperoleh dari pengerjaan Tugas Akhir ini dan saran-saran mengenai pengembangan Tugas Akhir ini.

6.1 Kesimpulan

Berdasarkan proses perancangan, implementasi dan pengujian yang dilakukan, maka kesimpulan-kesimpulan yang dapat diambil adalah:

1. Pengukuran tingkat kepekatan asap berdasarkan Ringelmann Smoke Chart dapat dilakukan dengan mudah menggunakan aplikasi ini. Aplikasi dapat mengimplementasikan metode pengukuran *Image Retrieval Based on Color Histogram* dengan akurasi sebesar 72%.
2. Penyimpanan data pengukuran tingkat kepekatan asap dapat dilakukan dengan mudah menggunakan aplikasi ini. Selain fitur penyimpanan, aplikasi juga dapat mengimplementasikan fitur riwayat pengukuran dan *benchmark* nilai kepekatan asap.

6.2 Saran

Saran untuk pengembangan dan perbaikan sistem di masa yang akan datang sesuai hasil perancangan, implementasi dan uji coba yang telah dikerjakan adalah aplikasi dapat dijalankan pada platform lain selain Windows Phone.

DAFTAR PUSTAKA

- [1] BAPEDAL, “Kep-205/BAPEDAL/07/1996 tentang Pedoman Teknis Pengendalian Pencemaran Udara,” 1996.
- [2] Software Testing Fundamentals, “Black-box testing,” [Online]. Available: <http://softwaretestingfundamentals.com/black-box-testing/>. [Diakses 4 Maret 2014].
- [3] T. Kurniawan, “Pengukuran kepekatan asap dengan kartu Ringelmann,” [Online]. Available: <http://infokelistrikan.blogspot.com/>. [Diakses 4 Maret 2014].
- [4] Microsoft, “Download Windows Phone SDK 8.0 from Official Microsoft Download Center,” 30 October 2012. [Online]. Available: Available: <http://www.microsoft.com/en-us/download/details.aspx?id=35471>. [Diakses 4 Maret 2014].
- [5] Microsoft, “What Is Windows Communication Foundation,” [Online]. Available: [http://msdn.microsoft.com/en-us/library/ms731082\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/ms731082(v=vs.110).aspx). [Diakses 4 Maret 2014].
- [6] Techopedia, “Internet Information Services (IIS),” [Online]. Available: <http://www.techopedia.com/definition/24953/internet-information-services-iis>. [Diakses 5 Maret 2014].
- [7] R. Chakravarti dan X. Meng, “A Study of Color Histogram Based Image Retrieval,” *2009 Sixth International Conference on Information Technology: New Generations*, 2009.

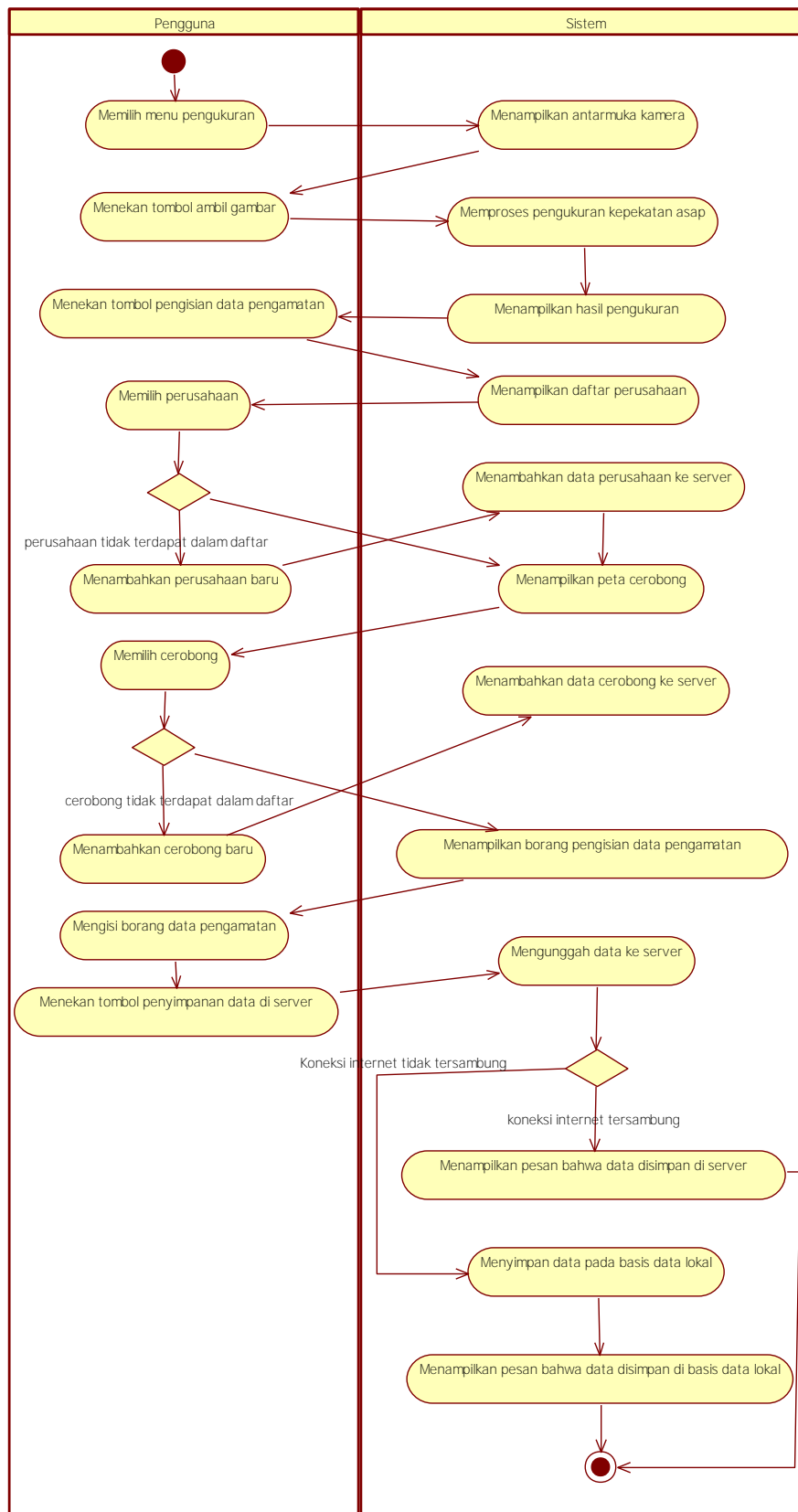
BIODATA PENULIS



Penulis, Mohammad Choirul Rochman Fauzi lahir di Bojonegoro pada 14 Mei 1992. Penulis adalah anak keempat dari empat bersaudara dan dibesarkan di Bojonegoro, Jawa Timur. Penulis menempuh pendidikan formal di SD Negeri Panjuran 2 (1998-2004), SMPN 1 Kalitidu (2004-2007), SMAN 1 Bojonegoro (2007-2010). Pada tahun 2010, penulis memulai pendidikan S1 jurusan Teknik Informatika Fakultas Teknologi Informasi di Institut Teknologi Sepuluh Nopember, Surabaya, Jawa Timur.

Di jurusan Teknik Informatika, penulis mengambil bidang minat Rekayasa Perangkat Lunak dan memiliki ketertarikan di bidang *Mobile Programming* dan *Desktop Software Development*. Penulis juga aktif dalam organisasi kemahasiswaan seperti Himpunan Mahasiswa Teknik Computer (HMTC). Penulis dapat dihubungi melalui alamat email mcr.fauzie@gmail.com.

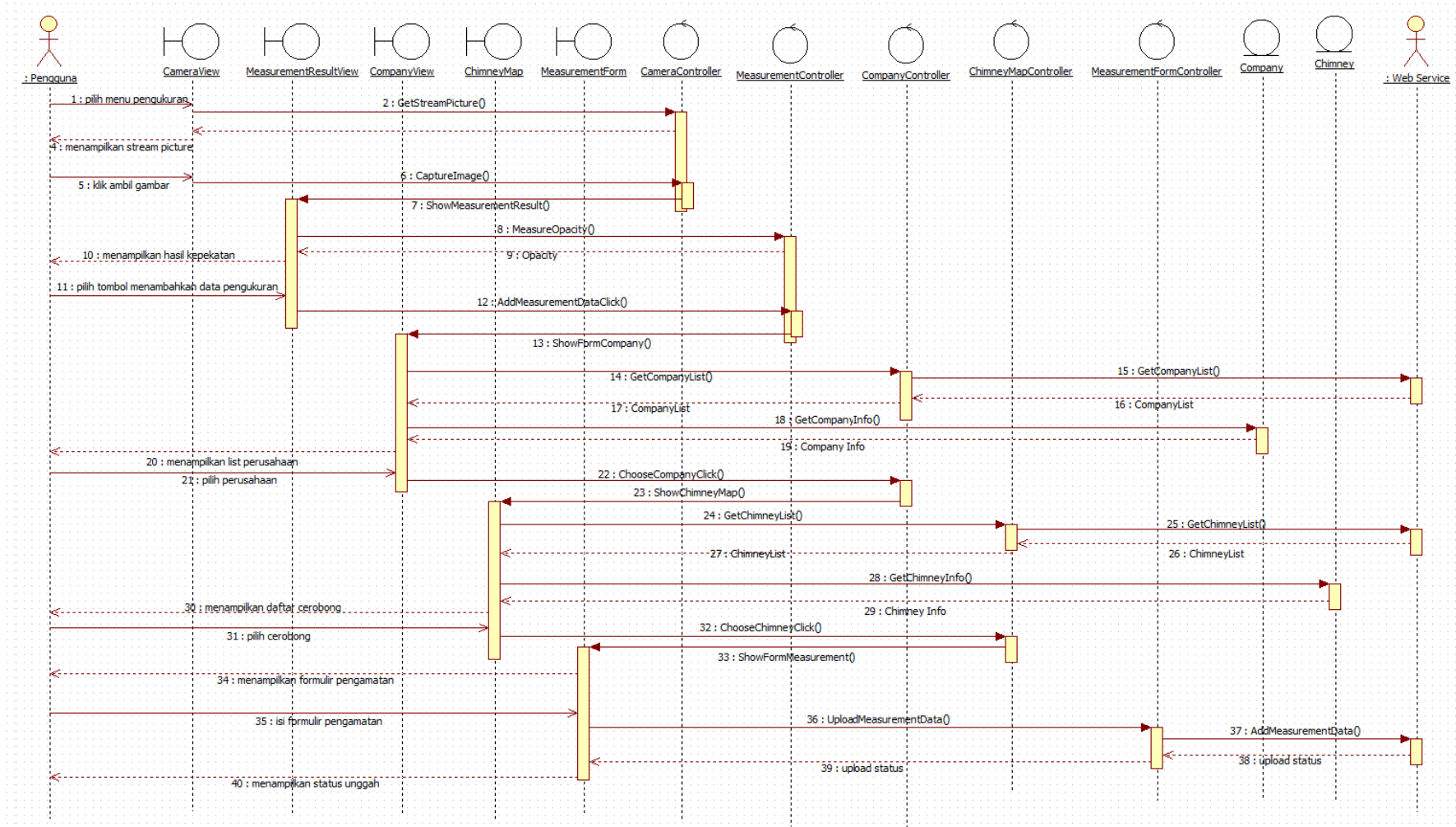
LAMPIRAN A. Diagram Aktivitas Kasus Penggunaan



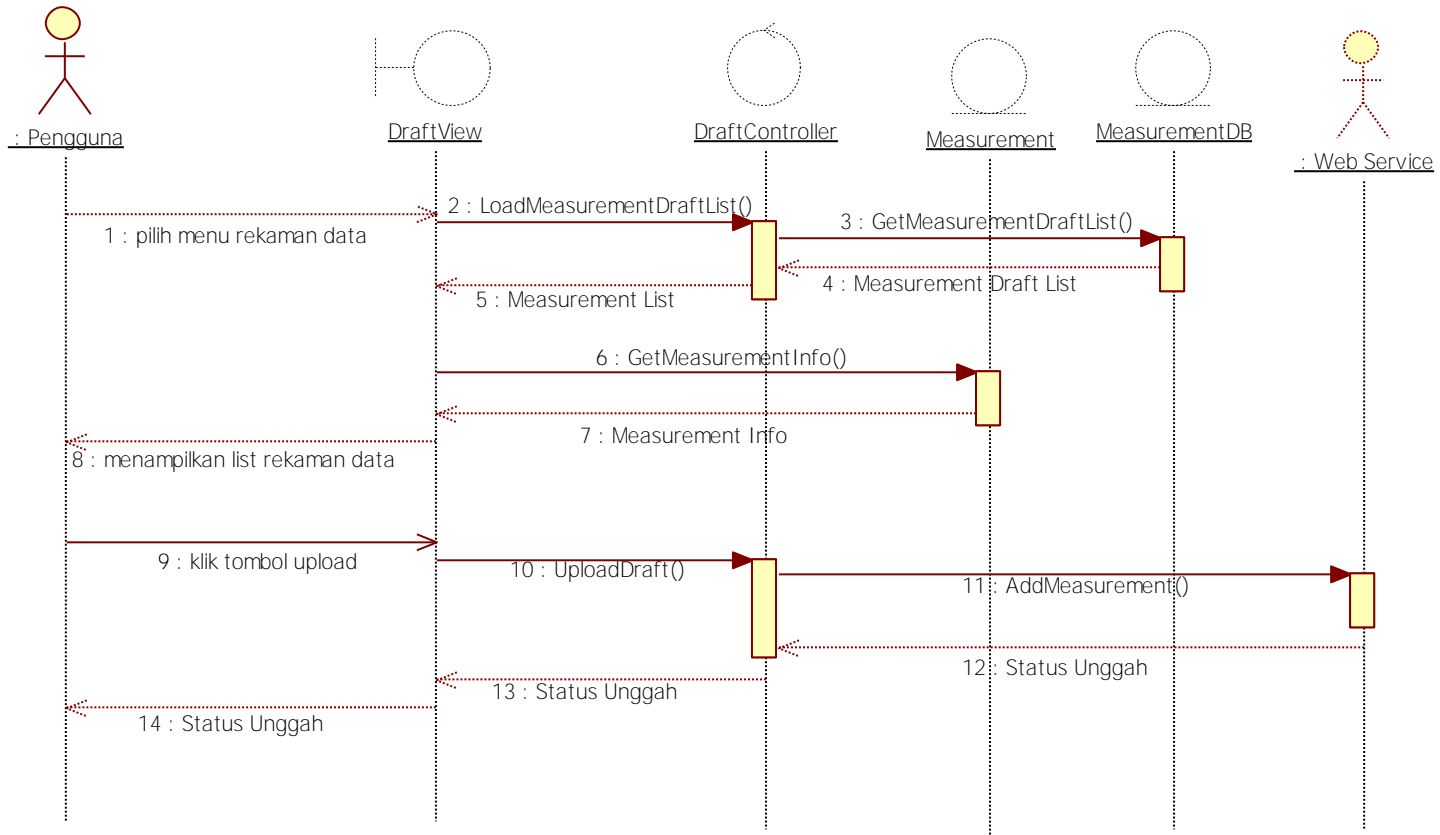
Gambar 7.1 Diagram Aktivitas Mengukur Tingkat Kepekatan Asap

(Halaman ini sengaja dikosongkan)

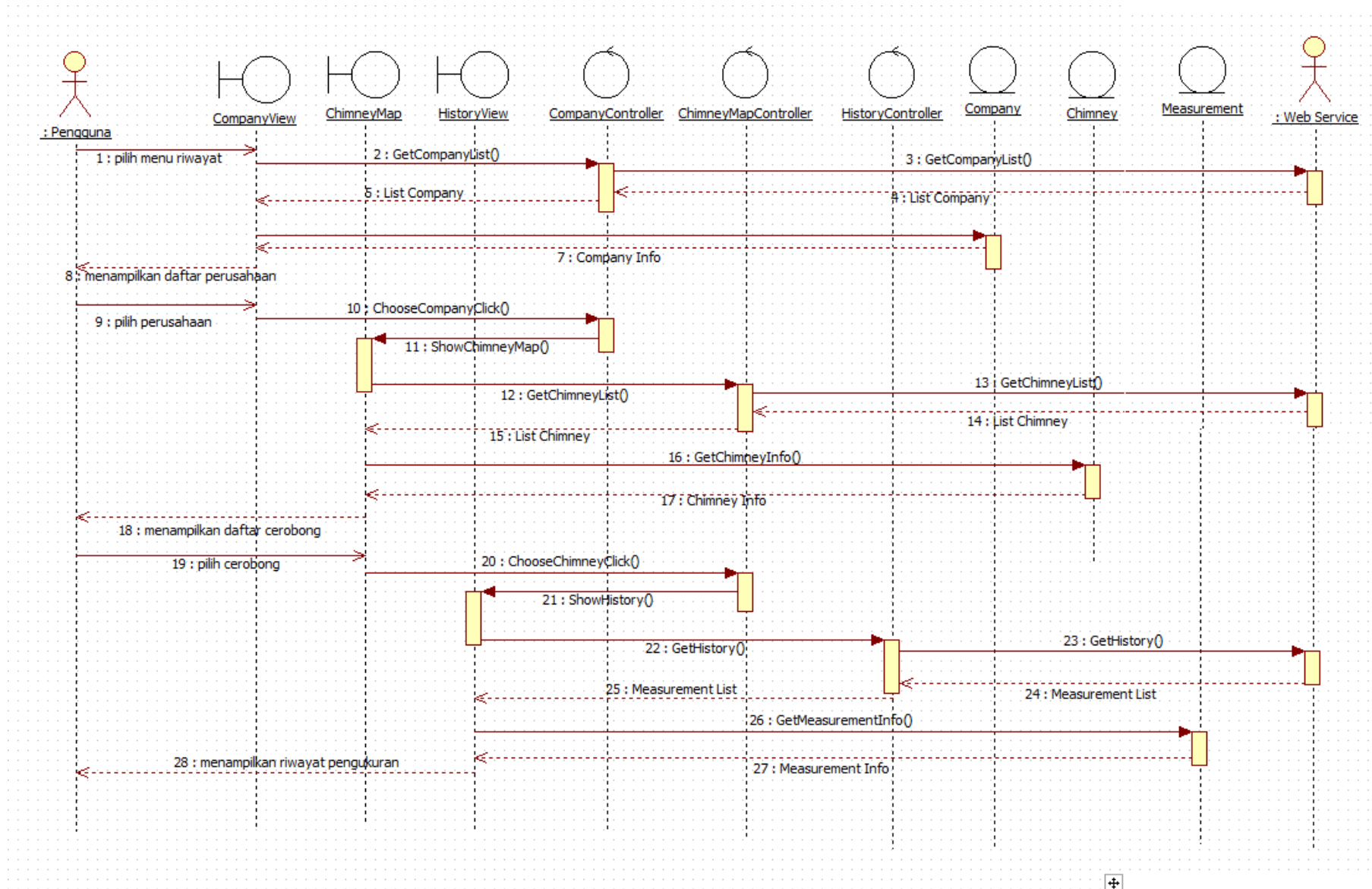
LAMPIRAN B. Diagram Urutan Kasus Penggunaan



Gambar 8.1 Diagram Urutan Mengukur Tingkat Kepekatan Asap



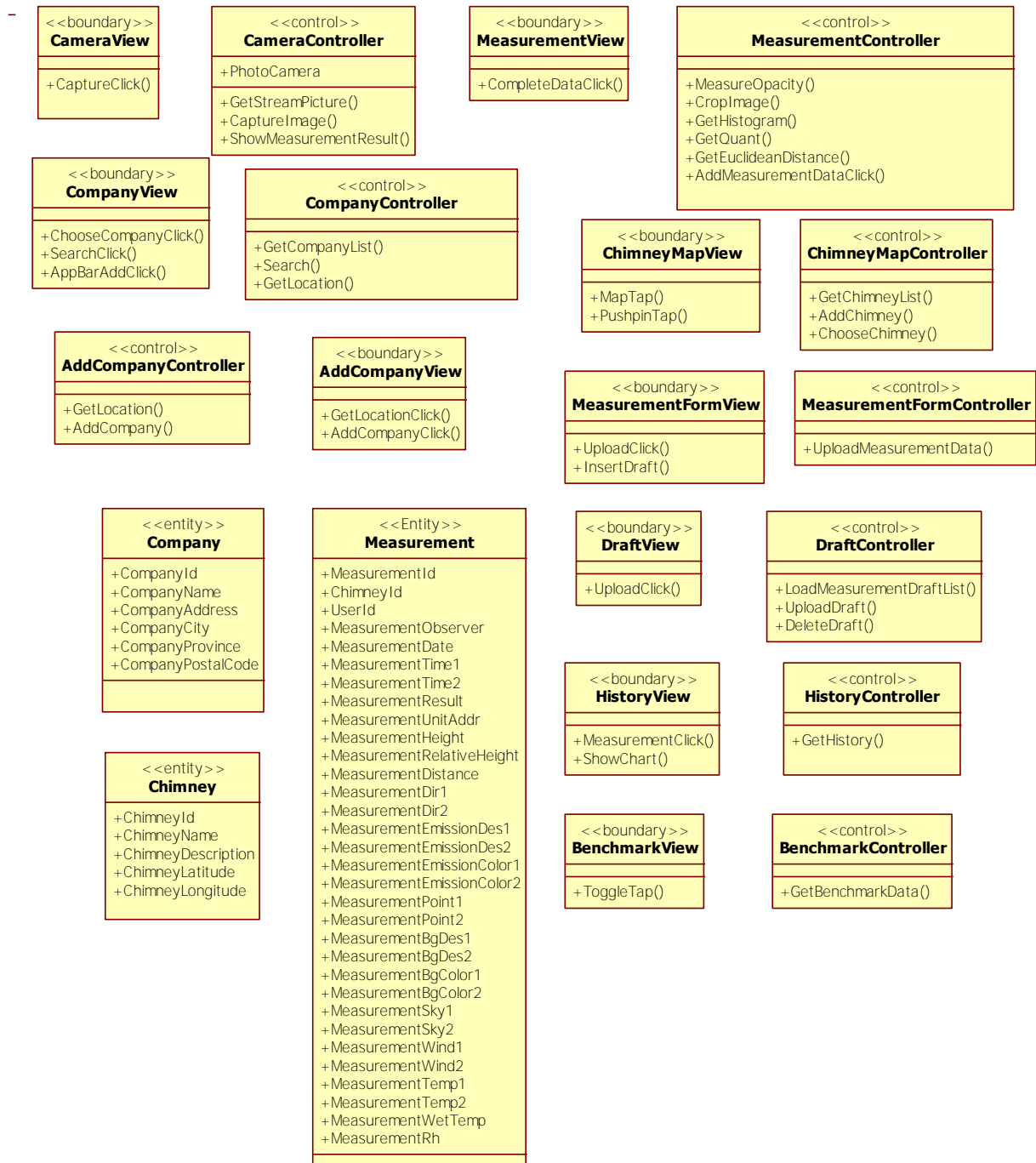
Gambar 8.2 Diagram Urutan Mengunggah Rekaman Data ke Server



Gambar 8.3 Diagram Urutan Menampilkan Riwayat Pengukuran

(Halaman ini sengaja dikosongkan)

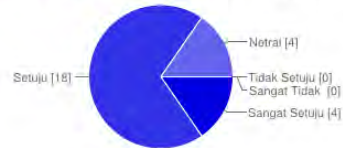
LAMPIRAN C. Diagram Kelas



(Halaman ini sengaja dikosongkan)

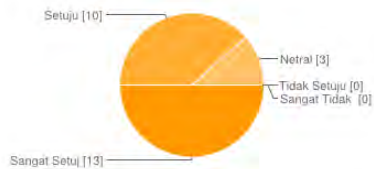
LAMPIRAN D. Hasil Kuesioner

Pengukuran tingkat kepekatan asap dapat dilakukan dengan mudah menggunakan aplikasi ini.



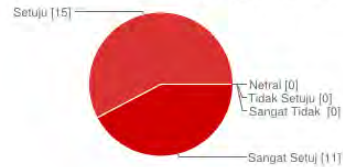
Sangat Setuju	4	15%
Setuju	18	69%
Netral	4	15%
Tidak Setuju	0	0%
Sangat Tidak Setuju	0	0%

Pengisian data pengamatan melalui aplikasi ini lebih mudah dan cepat dibandingkan dengan pengisian secara manual pada kertas laporan pengamatan.



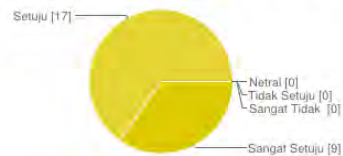
Sangat Setuju	13	50%
Setuju	10	38%
Netral	3	12%
Tidak Setuju	0	0%
Sangat Tidak Setuju	0	0%

Fitur rekaman data membantu pengguna untuk tidak mengulang proses pengukuran dan pengisian data pengamatan, apabila koneksi internet sedang bermasalah.



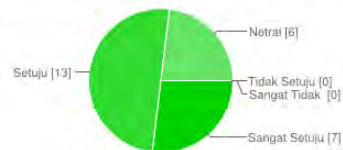
Sangat Setuju	11	42%
Setuju	15	58%
Netral	0	0%
Tidak Setuju	0	0%
Sangat Tidak Setuju	0	0%

Fitur histori / riwayat mempermudah pengguna untuk tracking perkembangan kepekatan asap cerobong.



Sangat Setuju	9	35%
Setuju	17	65%
Netral	0	0%
Tidak Setuju	0	0%
Sangat Tidak Setuju	0	0%

Benchmark dapat ditampilkan dengan baik pada aplikasi ini.



Sangat Setuju	7	27%
Setuju	13	50%
Netral	6	23%
Tidak Setuju	0	0%
Sangat Tidak Setuju	0	0%

(Halaman ini sengaja dikosongkan)

LAMPIRAN E. Hasil Pengukuran

Tabel 11.1 Perbandingan Hasil Pengukuran Oleh Aplikasi dengan Hasil Pengukuran Oleh Pengamat Secara Manual

No	Tanggal	Waktu	Lokasi	Hasil Aplikasi	Hasil Manual 1	Hasil Manual 2	Hasil Manual 3	Hasil Manual 4	Hasil Manual 5
1	22/05/2014	11:05	Taman, Sidoarjo	20%	20%	20%	20%	20%	20%
2	22/05/2014	11:06	Taman, Sidoarjo	20%	20%	20%	20%	20%	20%
3	22/05/2014	11:07	Taman, Sidoarjo	20%	20%	20%	20%	20%	20%
4	22/05/2014	11:09	Taman, Sidoarjo	20%	20%	20%	20%	20%	20%
5	22/05/2014	11:10	Taman, Sidoarjo	20%	20%	20%	20%	20%	20%
6	22/05/2014	11:11	Taman, Sidoarjo	20%	20%	20%	20%	20%	20%
7	22/05/2014	12:45	Krian, Sidoarjo	40%	60%	60%	60%	60%	60%
8	22/05/2014	12:46	Krian, Sidoarjo	40%	60%	60%	60%	60%	60%
9	22/05/2014	12:47	Krian, Sidoarjo	40%	60%	60%	60%	60%	60%
10	22/05/2014	12:48	Krian, Sidoarjo	40%	60%	60%	60%	60%	60%
11	22/05/2014	12:49	Krian, Sidoarjo	40%	60%	60%	60%	60%	60%

12	22/05/2014	12:50	Krian, Sidoarjo	40%	60%	60%	60%	60%	60%
13	22/05/2014	12:51	Krian, Sidoarjo	40%	60%	60%	60%	60%	60%
14	22/05/2014	13:01	Krian, Sidoarjo	40%	40%	40%	40%	40%	40%
15	22/05/2014	13:02	Krian, Sidoarjo	40%	40%	40%	40%	40%	40%
16	22/05/2014	13:03	Krian, Sidoarjo	40%	40%	40%	40%	40%	40%
17	22/05/2014	13:04	Krian, Sidoarjo	40%	40%	40%	40%	40%	40%
18	24/06/2014	11:01	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
19	24/06/2014	11:02	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
20	24/06/2014	11:03	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
21	24/06/2014	11:05	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
22	24/06/2014	11:06	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
23	24/06/2014	11:07	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
24	24/06/2014	11:08	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
25	24/06/2014	11:09	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%

26	24/06/2014	11:10	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
27	24/06/2014	11:11	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
28	24/06/2014	11:12	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
29	24/06/2014	11:13	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
30	24/06/2014	11:14	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
31	24/06/2014	11:15	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
32	24/06/2014	11:16	Rungkut Industri 3, Surabaya	20%	20%	20%	20%	20%	20%
33	24/06/2014	13:10	Krian, Sidoarjo	20%	20%	20%	20%	20%	20%
34	24/06/2014	13:11	Krian, Sidoarjo	20%	20%	20%	20%	20%	20%
35	24/06/2014	13:12	Krian, Sidoarjo	20%	20%	20%	20%	20%	20%
36	24/06/2014	13:13	Krian, Sidoarjo	20%	20%	20%	20%	20%	20%
37	24/06/2014	13:14	Krian, Sidoarjo	20%	40%	40%	40%	40%	40%
38	24/06/2014	13:15	Krian, Sidoarjo	20%	20%	20%	20%	20%	20%
39	24/06/2014	13:16	Krian, Sidoarjo	20%	40%	20%	20%	40%	40%
40	24/06/2014	13:17	Krian, Sidoarjo	20%	40%	40%	40%	40%	40%

41	24/06/2014	13:18	Krian, Sidoarjo	20%	40%	40%	40%	40%	40%
42	24/06/2014	13:19	Krian, Sidoarjo	20%	40%	40%	40%	40%	40%