



TUGAS AKHIR - TJ 141502

**APLIKASI BERGERAK BERBASIS GEOLOCATION DAN
GOOGLE MAPS SEBAGAI ALAT BANTU VERIFIKASI IMB
KOTA SURABAYA**

Nurfajariyawan Juniantera
NRP 2913 100 024

Dosen Pembimbing
Dr. I Ketut Eddy Purnama, ST., MT.
Muhtadin, ST., M.Sc.

DEPARTEMEN TEKNIK KOMPUTER
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2017

[Halaman ini sengaja dikosongkan].



FINAL PROJECT - TJ 141502

***IMPLEMENTATION OF GEOLOCATION AND GOOGLE
MAPS FOR VERIFYING BUILDING PERMISSION IN
SURABAYA BASED ON ANDROID APPLICATION***

Nurfajariyawan Juniantera
NRP 2913 100 024

Advisors
Dr. I Ketut Eddy Purnama, ST., MT.
Muhtadin, ST., M.Sc.

Departement of Computer Engineering
Faculty of Electrical Technology
Sepuluh Nopember Institute of Technology
Surabaya 2017

[Halaman ini sengaja dikosongkan].

LEMBAR PENGESAHAN

Aplikasi Bergerak Berbasis *Geolocation* dan *Google Maps* Sebagai Alat Bantu Verifikasi IMB Kota Surabaya

Tugas Akhir ini disusun untuk memenuhi salah satu syarat memperoleh gelar Sarjana Teknik di Institut Teknologi Sepuluh Nopember Surabaya

Oleh : Nurfajariyawan Juniantera (NRP: 2913100024)

Tanggal Ujian : 12 Juli 2017

Periode Wisuda : September 2017

Disetujui oleh:

(Pembimbing I)

Dr. I Ketut Eddy Purnama, S.T., M.T.
NIP: 196907301995121001

(Pembimbing II)

Muhtadin, S.T., M.Sc.
NIP: 198106092009121003

(Penguji I)

Prof. Dr. Ir. Mauridhi Hery P., M.Eng.
NIP: 195809161986011001

(Penguji II)

Dr. Supeno Mardi Susiki N., ST., M.T.
NIP: 197003131995121001

(Penguji III)

Dr. Adhi Dharma Wibawa, ST., MT.
NIP: 197605052008121003

Mengetahui
Kepala Departemen Teknik Komputer

Dr. I Ketut Eddy Purnama, S.T., M.T.
NIP: 196907301995121001



PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul “**Aplikasi Bergerak Berbasis Geolocation dan Google Maps Sebagai Alat Bantu Verifikasi IMB**” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juli 2017

Nurfajariyawan Juniantera

NRP. 2913100024

ABSTRAK

Nama Mahasiswa : Nurfajariyawan Juniantera
Judul Tugas Akhir : Aplikasi Bergerak Berbasis Geolocation dan Google Maps Sebagai Alat Bantu Verifikasi IMB Kota Surabaya
Pembimbing : 1. Dr. I Ketut Eddy Purnama, ST., MT.
2. Muhtadin, ST., M.Sc.

Permintaan surat IMB pada suatu daerah padat penduduk memiliki kuantitas yang cukup banyak dan selalu mengalami lonjakan jumlah pada tahun berikutnya. Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi bergerak yang dapat digunakan sebagai alat bantu untuk melakukan pemeriksaan bangunan dan sebagai wadah untuk menampung laporan-laporan pemeriksaan yang telah dilakukan petugas secara *realtime*. Melalui aplikasi ini, berkas-berkas fisik bangunan yang dibutuhkan petugas disediakan dalam bentuk digital sehingga dapat diakses kapanpun dan dimanapun. Fitur geolokasi digunakan sebagai visualisasi letak bangunan sehingga informasi bangunan yang ditampilkan sesuai dengan lokasi bangunan. Dari hasil uji lapangan yang dilakukan, aplikasi ini mampu menjalankan keseluruhan skenario sistem dengan baik ketika perangkat berada pada jaringan 4g dan 3g. Uji ketergunaan, kebenaran dan keandalan aplikasi juga dilakukan dengan memberikan kuesioner kepada petugas. Dari hasil pengujian didapatkan skor rata-rata 74.1% untuk ketiga unsur tersebut sehingga aplikasi ini mampu diterima oleh petugas sebagai alat bantu pemeriksaan bangunan dan verifikasi IMB.

Kata Kunci : Verifikasi IMB, Aplikasi Bergerak, *Android*, *Geolocation*

Halaman ini sengaja dikosongkan

ABSTRACT

Name : Nurfajariyawan Juniantera
Title : *Implementation of Geolocation and Google Maps for Verifying Building Permission in Surabaya Based on Android Application.*
Advisors : 1. Dr. I Ketut Eddy Purnama, ST., MT.
2. Muhtadin, ST., M.Sc.

The demand for an Permits Building License in a densely populated area has a considerable quantity and always experiences a surge in the number for the following year. This study aims to develop a mobile application that can be used as a tool for conducting a realtime inspection of buildings and as a container to accommodate inspection reports that have been done by the officer. Through this application, the building's physical files required by the officers are provided in digital form so that they can be accessed whenever and wherever. Geolocation feature is used as a visualization for the location of the building so that the building information shown in accordance with the physical location of the building. From the results of field tests conducted, this application is able to run well, in the overall scenario of the system when the device is using the 4g and 3g networks. Test of usability, truth and reliability of application is also done by giving questionnaire to the officer. From the test results obtained an average score of 74.1% for the three elements so that this application can be accepted by the officer as a tool of building inspection and verification Permits Building License.

Keywords : Permits Building License verification, Mobile Application, Android, Geolocation

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji dan syukur kehadiran Allah SWT atas segala limpahan berkah, rahmat, serta hidayah-Nya, penulis dapat menyelesaikan penelitian ini dengan judul **Aplikasi Bergerak Berbasis Geolocation dan Google Maps Sebagai Alat Bantu Verifikasi IMB Kota Surabaya**.

Penelitian ini disusun dalam rangka pemenuhan bidang riset di Departemen Teknik Komputer ITS, serta digunakan sebagai persyaratan menyelesaikan pendidikan S1. Penelitian ini dapat terselesaikan tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Keluarga, Ibu, dan Ayah yang telah memberikan dorongan spiritual dan material dalam penyelesaian buku penelitian ini.
2. Bapak Kepala Departemen Teknik Komputer Dr. I Ketut Eddy Purnama, ST., MT.
3. Bapak Dr. I Ketut Eddy Purnama, ST., MT. dan Bapak Muhtadin, ST., M.Sc. atas bimbingan selama mengerjakan penelitian.
4. Bapak-ibu dosen pengajar Bidang Studi Telematika, atas pengajaran, bimbingan, serta perhatian yang diberikan kepada penulis selama ini.
5. Seluruh teman-teman *B201-crew* Laboratorium Bidang Studi Teknik Komputer dan Telematika dan teman-teman asisten lab B201 yang telah memberikan dukungan yang luar biasa.

Kesempurnaan hanya milik Allah SWT, untuk itu penulis memohon segenap kritik dan saran yang membangun. Semoga penelitian ini dapat

Surabaya, 14 Juni 2017

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

Abstrak	i
Abstract	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
DAFTAR KODE	xiii
1 PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Permasalahan	2
1.3 Tujuan	3
1.4 Batasan masalah	3
1.5 Sistematika Penulisan	4
2 TINJAUAN PUSTAKA	7
2.1 Izin Mendirikan Bangunan (IMB)	7
2.2 Landasan hukum IMB Kota Surabaya	7
2.3 SOP Petugas Pengendalian	8
2.4 Landasan hukum Standard Operating Procedure (SOP) Pengawasan IMB	8
2.5 Formula Haversine	9
2.6 Standar WGS 84	10
2.7 <i>Generic List di Java</i>	11
3 DESAIN	13
3.1 Desain Sistem	13
3.1.1 Desain Aplikasi Berbasis <i>UML</i>	17
3.2 Perancangan Basis Data	23
3.2.1 Data yang Digunakan	23

3.2.2	Desain Aliran Data	23
3.2.3	Skema Relasi Basis Data	25
3.2.4	Struktur Tabel	27
3.3	Desain Antarmuka Pengguna	31
4	IMPLEMENTASI DAN HASIL	37
4.1	Implementasi	37
4.1.1	Persiapan Data	37
4.1.2	Penerimaan Data	37
4.1.3	Penghitungan Jarak Banguann	39
4.1.4	Penampilan Data Bangunan	40
4.1.5	Penyusunan Data Bangunan	40
4.1.6	Penampilan gambar bangunan	41
4.1.7	Pemeriksaan Bangunan	41
4.1.8	Verifikasi Bangunan	42
4.1.9	Laporkan Bangunan	42
4.2	Hasil	43
4.2.1	Hasil <i>User Interface</i>	43
5	PENGUJIAN	49
5.1	Pengujian Sistem	49
5.2	Pengujian Fungsional Aplikasi	53
5.3	Pengujian API Berdasarkan Tipe Jaringan	55
5.3.1	API <i>Login</i>	55
5.3.2	API <i>GetData</i>	56
5.3.3	API <i>PostLaporan</i>	58
5.3.4	API <i>PostVerifikasi</i>	59
5.4	Pengujian Sensor	61
6	PENUTUP	63
6.1	Kesimpulan	63
6.2	Penelitian Selanjutnya	63
	DAFTAR PUSTAKA	65
	LAMPIRAN	67
	Biografi Penulis	67

DAFTAR GAMBAR

2.1	Definisi sistem koordinat WGS 84	10
3.1	Gambaran umum sistem kerja	13
3.2	Alur kerja proses persiapan data	14
3.3	Alur kerja proses penerimaan data	15
3.4	Alur kerja proses pemeriksaan bangunan	16
3.5	Use Case Diagram	17
3.6	Activity diagram persiapan data	19
3.7	Activity diagram persiapan data bagian 1	20
3.8	Activity diagram persiapan data bagian 2	21
3.9	Activity diagram pemeriksaan bangunan	22
3.10	Data flow diagram keseluruhan sistem	24
3.11	Skema relasi basis data	26
3.12	Tampilan <i>Splash</i>	32
3.13	Tampilan <i>login</i> dan <i>register</i>	32
3.14	Tampilan menu utama aplikasi	33
3.15	Tampilan detail informasi dan lokasi bangunan	34
3.16	Tampilan formulir laporan	35
4.1	Algoritma proses persiapan data	38
4.2	Algoritma proses penerimaan data	40
4.3	Tampilan <i>login</i> dan <i>register</i>	43
4.4	Tampilan Menu Utama dan <i>register</i>	44
4.5	Detail bangunan	45
4.6	Lokasi bangunan	46
4.7	Formulir pelaporan bangunan	47
5.1	Hasil pengujian <i>API Login</i>	56
5.2	Hasil pengujian <i>API GetData</i>	57
5.3	Hasil pengujian <i>API PostLaporan</i>	59
5.4	Hasil pengujian <i>API PostVerifikasi</i>	60

Halaman ini sengaja dikosongkan

DAFTAR TABEL

3.1	Penjelasan <i>Use case</i> diagram	18
3.2	Variabel yang digunakan dalam proses pemeriksaan IMB	23
3.3	Struktur tabel bangunan	27
3.4	Struktur tabel alamat	28
3.5	Struktur tabel kecamatan	28
3.6	Struktur tabel kelurahan	28
3.7	Struktur tabel ket_imb	29
3.8	Struktur tabel land_use	29
3.9	Struktur tabel pemilik	29
3.10	Struktur tabel laporan	30
3.11	Struktur tabel user	30
3.12	Struktur tabel verifikasi	31
3.13	Struktur tabel wilayah	31
5.1	Hasil Pengujian <i>Betha</i> Kepada Mahasiswa	50
5.2	Hasil Pengujian <i>Betha</i> Kepada Petugas	51
5.3	Bobot penilaian berdasarkan pilihan jawaban	52
5.4	Hasil penghitungan nilai dengan metode Linkert	52
5.5	Hasil penghitungan nilai dengan metode Linkert	53
5.6	Hasil pengujian <i>Black Box</i>	54
5.7	Hasil pengujian <i>API Login</i>	55
5.8	Hasil pengujian <i>API GetData</i>	57
5.9	Hasil pengujian <i>API PostLaporan</i>	58
5.10	Hasil pengujian <i>API PostVerifikasi</i>	60
5.11	Hasil pengujian akurasi sensor <i>GPS</i>	61

Halaman ini sengaja dikosongkan

DAFTAR KODE

2.1	Konfigurasi <i>generic list</i>	11
4.1	<i>Pseudocode</i> proses penerimaan data	39
4.2	<i>Pseudocode</i> proses pernghitungan jarak	39
4.3	<i>Pseudocode</i> proses penyusunan data bangunan . . .	40
4.4	<i>Pseudocode</i> proses penampilan gambar	41
4.5	<i>Pseudocode</i> proses pemeriksaan bangunan	41
4.6	<i>Pseudocode</i> proses verifikasi bangunan	42
4.7	<i>Pseudocode</i> proses laporkan bangunan	42

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

Penelitian ini di latar belakang oleh berbagai kondisi yang menjadi acuan. Selain itu juga terdapat beberapa permasalahan yang akan dijawab sebagai luaran dari penelitian.

1.1 Latar belakang

Unsur Pendapatan Asli Daerah (PAD) diantaranya adalah Pajak Daerah, Retribusi Daerah, Hasil Perusahaan Milik Daerah dan Hasil Pengelolaan Kekayaan Daerah. Dari beberapa sumber tersebut, pajak daerah dan retribusi daerah merupakan sumber pendapatan yang paling besar. Dasar hukum yang mengatur tentang Pajak Daerah dan Retribusi Daerah adalah Undang-undang No.34 tahun 2000 mengenai Perubahan atas Undang-undang RI No.18 tahun 1997 tentang Pajak Daerah dan Retribusi Daerah.

Pembangunan daerah merupakan sebuah proses yang berkesinambungan yang perlu didukung oleh sumber dana, sumber daya alam, dan sumber daya manusia. Salah satu upaya pemerintah meningkatkan pendapatan daerah adalah dengan mewajibkan tiap bangunan memiliki Ijin Mendirikan Bangunan (IMB).

Jumlah proyek pembangunan yang berlangsung dalam satu periode waktu tertentu di sebuah daerah padat penduduk tidaklah sedikit. Menurut data statistik yang disampaikan oleh Dinas Perumahan Rakyat dan Kawasan Pemukiman, Cipta Karya dan Tata Ruang Kota Surabaya, pada tahun 2010 tercatat sebanyak 3.145 Surat Izin Mendirikan Bangunan (IMB) yang dikeluarkan oleh Pemerintah. Dari jumlah tersebut, 1.855 diantaranya adalah surat izin yang diterbitkan untuk bangunan baru. Hal ini merupakan sebuah indikasi bahwa setidaknya ada ribuan bangunan baik itu dalam kategori tempat tinggal maupun bukan tempat tinggal yang harus diperiksa berdasarkan surat IMB setiap tahun. Kondisi saat ini, petugas dari Dinas Perumahan Rakyat dan Kawasan Pemukiman, Cipta Karya dan Tata Ruang Kota Surabaya melakukan pengawasan IMB melalui pemeriksaan dan penyesuaian kondisi fisik bangunan dengan berkas-berkas yang telah didaftarkan, sehingga pada proses pemeriksaan satu bangunan petugas harus membawa berkas yang

banyak (contohnya : Kartu Tanda Penduduk, Sertifikat Kepemilikan Tanah, Gambar Konstruksi Bangunan dan lain-lain) selain itu penentuan jadwal pemeriksaan tidak fleksibel sehingga harus dilakukan penjadwalan sebelum waktu pengecekan dilaksanakan. Hal ini berakibat pada kurangnya performa petugas dalam proses pemeriksaan karena berkas-berkas yang dibawa petugas menjadi banyak dan membutuhkan waktu yang cukup lama.

1.2 Permasalahan

Sistem pemeriksaan IMB yang dijalankan oleh dinas saat ini masih konvensional. Berkas-berkas pemeriksaan IMB yang dibawa petugas disajikan dalam bentuk fisik, hal tersebut dapat menimbulkan terjadinya penumpukan berkas ketika proses pemeriksaan bangunan berlangsung. Selain itu belum ada tempat penampungan laporan pemeriksaan IMB yang dapat menghasilkannya sebuah laporan yang cepat, baik dan terstruktur. Berdasarkan sistem pemeriksaan IMB yang diterapkan saat ini, maka dapat dirumuskan permasalahan sebagai berikut:

1. Informasi bangunan yang telah mengajukan Izin Mendirikan Bangunan sudah tersimpan dalam bentuk digital di dalam server internal milik Dinas Perumahan Rakyat dan Kawasan Pemukiman, Cipta Karya dan Tata Ruang, namun informasi tersebut belum disajikan secara digital dan belum dapat diakses oleh petugas di lapangan untuk digunakan dalam proses pemeriksaan bangunan.
2. Belum ada wadah untuk menampung data hasil dari proses pemeriksaan IMB, sehingga data tersebut tidak dapat disajikan dalam bentuk informasi yang terstruktur.
3. Laporan tertulis pemeriksaan IMB tidak dibuat langsung pada saat proses pemeriksaan berlangsung, namun laporan tersebut dikerjakan pada saat petugas telah kembali ke kantor atau pada saat lain, hal ini bisa memicu terjadinya kesalahan pencatatan karena faktor *human error* dan juga bisa memicu terjadinya kecurangan dalam pelaporan IMB.

Dengan adanya permasalahan-permasalahan tersebut, maka aplikasi yang akan dibuat bertujuan untuk menyajikan informasi-informasi

yang dibutuhkan oleh petugas serta membuat sebuah wadah untuk menampung hasil pemeriksaan agar hasil tersebut dapat diolah dan disajikan menjadi informasi yang jelas dan terstruktur.

1.3 Tujuan

Tujuan utama dari tugas akhir ini adalah mengembangkan sebuah aplikasi *mobile* berbasis *Android* yang dimanfaatkan untuk:

1. Menyajikan *dummy data* yang dirancang berdasarkan kebutuhan petugas pemeriksaan IMB dalam bentuk digital, sehingga petugas tidak perlu membawa berkas fisik bangunan.
2. Memastikan data akurat dengan memanfaatkan *geolocation* dan *google maps*, sehingga hasil pemeriksaan dapat dipertanggungjawabkan.
3. Menyediakan sebuah wadah yang berfungsi untuk menampung hasil pemeriksaan IMB dan menyediakan data laporan yang terstruktur, sehingga data tersebut dapat dimanfaatkan untuk membuat sebuah laporan yang cepat dan terstruktur dengan baik.

1.4 Batasan masalah

Untuk memfokuskan permasalahan yang akan diangkat maka dilakukan pembatasan masalah. Batasan-batasan masalah tersebut diantaranya adalah:

1. Data yang digunakan adalah *dummy data* yang berdasar pada lokasi bangunan sebenarnya dengan informasi yang dibuat sedemikian sehingga sesuai dengan peraturan pemerintah kota surabaya.
2. Aplikasi dikembangkan dan dijalankan pada Framework Android dengan minimal *Standard Development Kit* versi 21.
3. Aplikasi dijalankan pada perangkat Android yang dilengkapi dengan sensor GPS.
4. Perangkat yang digunakan untuk pengujian memerlukan koneksi dengan jaringan internet untuk mengakses data yang tersimpan di dalam basis data.
5. Pengujian dilakukan pada cuaca cerah dan tidak di dalam ruangan untuk menjaga keakurasian sensor GPS.

6. Kompleksitas data pada tugas akhir ini hanya dapat digunakan sebagai acuan pemeriksaan kondisi fisik bangunan dengan data yang ditampilkan.

1.5 Sistematika Penulisan

Laporan penelitian Tugas akhir ini tersusun dalam sistematika dan terstruktur sehingga mudah dipahami dan dipelajari oleh pembaca maupun seseorang yang ingin melanjutkan penelitian ini. Alur sistematika penulisan laporan penelitian ini yaitu :

1. BAB I Pendahuluan
Bab ini berisi uraian mengenai latar belakang permasalahan, penegasan dan alasan pemilihan judul, sistematika laporan, tujuan dan metodologi penelitian.
2. BAB II Dasar Teori
Pada bab ini berisi uraian secara sistematis teori-teori yang berhubungan dengan permasalahan yang dibahas pada penelitian ini. Teori-teori ini digunakan sebagai dasar dalam penelitian, yaitu informasi terkait fitur apache spark, teknik-teknik visualisasi, dan teori-teori penunjang lainnya.
3. BAB III Desain
Bab ini berisi tentang penjelasan-penjelasan terkait perancangan sistem dan perancangan basis data. Sistem yang akan digunakan dalam tugas akhir ini diinterpretasikan menggunakan perancangan berbasis UML, sedangkan basis data yang akan digunakan dirancang dengan menyusun pola-pola aliran data, skema relasi basis data, struktur tabel
4. BAB IV Implementasi dan Hasil
Bab ini berisi tentang penjelasan tentang metode yang digunakan untuk mengimplementasikan rancangan yang telah dibuat sebelumnya. Selain itu pada bab ini juga dijelaskan mengenai hasil implementasi tersebut.
5. BAB V Pengujian Bab ini menjelaskan tentang pengujian eksperimen yang dilakukan terhadap data dan analisisnya. Beberapa teknik visualisasi akan ditunjukkan hasilnya pada bab ini dan dilakukan analisa terhadap hasil visualisasi dan informasi yang didapat dari hasil mengamati visualisasi yang tersaji.

6. BAB VI Penutup Bab ini merupakan penutup yang berisi kesimpulan yang diambil dari penelitian dan pengujian yang telah dilakukan. Saran dan kritik yang membangun untuk pengembangan lebih lanjut juga dituliskan pada bab ini.

Halaman ini sengaja dikosongkan

BAB 2

TINJAUAN PUSTAKA

Demi memperoleh hasil akhir yang sesuai dengan tujuan penelitian, dibutuhkan beberapa teori penunjang sebagai referensi. Dengan menerapkan referensi yang berlatar belakang keilmuan serta teknologi yang spesifik dengan judul penelitian, diharapkan peneliti mampu memahami prosedur serta variabel yang digunakan. Kajian teori yang digunakan dapat diuraikan sebagai berikut :

2.1 Izin Mendirikan Bangunan (IMB)

IMB adalah perizinan yang diberikan oleh Kepala Daerah kepada pemilik bangunan untuk membangun baru, mengubah, memperluas, mengurangi, dan/atau merawat bangunan sesuai dengan persyaratan administratif dan persyaratan teknis yang berlaku[1]. Persyaratan tersebut mengharuskan pemilik bangunan melaksanakan pembangunan yang mengacu pada Nilai Dasar Bangunan (NDB), Nilai Luas Bangunan (NLB) serta Ketinggian Bangunan (KB) yang ditetapkan sesuai dengan syarat-syarat keselamatan bagi penghuni bangunan, orang lain dan lingkungan[2].

2.2 Landasan hukum IMB Kota Surabaya

Dalam penerapan IMB kota Surabaya, pemerintah mengacu pada beberapa landasan Hukum berikut [3] :

1. Undang-undang no. 28 tahun 2002 tentang Bangunan Gedung, yang isinya :
Bangunan gedung adalah wujud fisik hasil pekerjaan konstruksi yang menyatu dengan tempat kedudukannya, sebagian atau seluruhnya berada di atas dan/atau di dalam tanah dan/atau air, yang berfungsi sebagai tempat manusia melakukan kegiatannya, baik untuk hunian atau tempat tinggal, kegiatan keagamaan, kegiatan usaha, kegiatan sosial, budaya, maupun kegiatan khusus [4].
2. Undang-undang no. 26 tahun 2007 tentang Penataan Ruang, yang isinya :
Ruang adalah wadah yang meliputi ruang darat, ruang laut,

dan ruang udara, termasuk ruang di dalam bumi sebagai satu kesatuan wilayah, tempat manusia dan makhluk lain hidup, melakukan kegiatan, dan memelihara kelangsungan hidupnya [5].

2.3 SOP Petugas Pengendalian

Petugas pengendalian memiliki prosedur pelaksanaan standar yang diterapkan dalam proses pengawasan IMB[3]. Prosedur tersebut yaitu :

1. Membuat konsep surat panggilan/peringatan/bantuan penertiban kepada pemilik atau penanggung jawab bangunan yang dimaksud
2. Mengecek pada peta pengendalian apakah pada persil yang dimaksud sudah pernah ada permohonan IMB
3. Membuat konsep surat penghentian pekerjaan fisik bangunan
4. Membuat konsep surat pernyataan kesanggupan menghentikan pelaksanaan kegiatan pembangunan di persil yang dimaksud
5. Menyerahkan surat panggilan yang telah ditandatangani pimpinan kepada pemilik/penanggung jawab bangunan
6. Menyerahkan surat peringatan yang telah ditandatangani pimpinan kepada pemilik/penanggung jawab bangunan dengan tembusan Walikota, Bagian Hukum dan Kecamatan
7. Menyerahkan surat bantuan penertiban yang telah ditandatangani pimpinan kepada Satuan Polisi Pamong Praja dengan tembusan Walikota, Asisten Perekonomian dan Pembangunan, Bagian Hukum dan Kecamatan

2.4 Landasan hukum Standard Operating Procedure (SOP) Pengawasan IMB

Prosedur standar yang dilakukan oleh Dinas Cipta Karya dan Tata Ruang Kota Surabaya berlandaskan pada beberapa sumber hukum berikut[3] :

1. Peraturan Menteri Pendayagunaan Aparatur Negara dan Reformasi Birokrasi Nomor 35 Tahun 2012 tentang Pedoman Penyusunan Standar Operasional Prosedur Administrasi Pemerintahan.
2. Peraturan Daerah Kota Surabaya Nomor 6 Tahun 2013 tentang Perubahan Atas Peraturan Daerah Kota Surabaya Nomor 7 Tahun 2009 tentang Bangunan.
3. Peraturan Daerah Kota Surabaya Nomor 9 Tahun 2013 tentang Perubahan atas Peraturan Daerah Kota Surabaya Nomor 12 Tahun 2012 tentang Retribusi Izin Mendirikan Bangunan.
4. Peraturan Daerah Kota Surabaya No. 7 Tahun 2009 tentang Bangunan sebagaimana telah diubah dengan Peraturan daerah Kota Surabaya No. 6 tahun 2013.

2.5 Formula Haversine

Formula Haversine adalah sebuah rumus matematika yang dapat digunakan untuk menghitung jarak antar dua koordinat geografis bumi.[6] Rumus perhitungan Haversine dapat didefinisikan pada persamaan 2.1 :

$$hav() = \sin^2(\Theta/2) = (1 - \cos(\Theta))/2 \quad (2.1)$$

nilai jarak dapat diperoleh dari persamaan 2.2 :

$$d = 2r \arcsin(\sqrt{hav(\varphi_2 - \varphi_1) + \cos(\varphi_2)\cos(\varphi_1)hav(\lambda_2 - \lambda_1)}) \quad (2.2)$$

Dengan :

1. φ_1, φ_2 sebagai latitude titik pertama dan kedua dalam radian.
2. λ_1, λ_2 sebagai longitude titik pertama dan kedua dalam radian.
3. hav sebagai fungsi haversine.
4. r sebagai jari- jari bumi.
5. dan d sebagai jarak dua kordinat.

2.6 Standar WGS 84

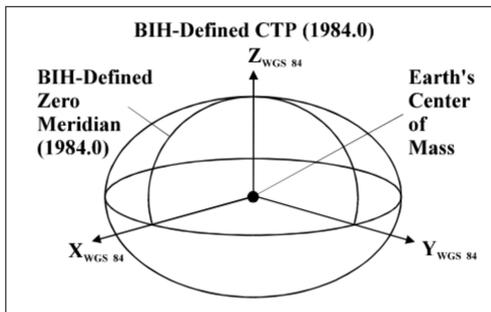
Google maps menggunakan standar WGS 84 dalam menentukan kartografi, geodesi, dan navigasi. WGS 84 adalah salah satu referensi sistem koordinat yang digunakan oleh *GPS* dalam menentukan garis lintang dan garis bujur bumi dengan akurasi ± 1 meter.

WGS84 terdiri dari bingkai koordinat standar Bumi, Datum geodetik, (referensi permukaan standar bulat (acuan atau referensi elipsoid) untuk data ketinggian mentah, dan permukaan ekuipotensial gravitasi (geoid) dipakai sebagai pendefinisian tingkat nominal laut [7].

Titik asal dan sumbu sistem koordinat WGS 84 didefinisikan sebagai berikut :

1. Titik Asal sebaga Pusat massa bumi.
2. *Z-Axis* sebagai Arah Kutub Terrestrial Konvensional (CTP) untuk kutub gerak.
3. *X-Axis* sebagai Titik temu bidang referensi meridian WGS 84 dan pesawat dari khatulistiwa CTP.
4. *Y-Axis* sebagai Sistem koordinat ortogonal *Earth Centered, Earth Fixed (ECEF)*, diukur di bidang CTP Khatulistiwa, 90 BT dari sumbu *x*.

Gambar ilustrasi titik asal dan sumbu kordinat WGS 84 disajikan pada gambar 2.1



Gambar 2.1: Definisi sistem koordinat WGS 84

2.7 *Generic List di Java*

List interface pada java merupakan sebuah tipe data array yang mengizinkan nilai yang sama dalam index yang berbeda. Sebuah List interface dapat digenerasikan, sehingga List dapat dimasuki maupun diambil datanya. Contoh konfigurasi Generic List disajikan pada kode 2.1.

Kode 2.1 Konfigurasi *generic list*

```
// Inisiasi Generic List
List<MyObject> list = new ArrayList<MyObject>();

//menambah elemen pada list
list.add("element 0");
list.add("element 1");
list.add("element 2");

//akses via index
String element0 = list.get(0);
String element1 = list.get(1);
String element3 = list.get(2);
```

Halaman ini sengaja dikosongkan

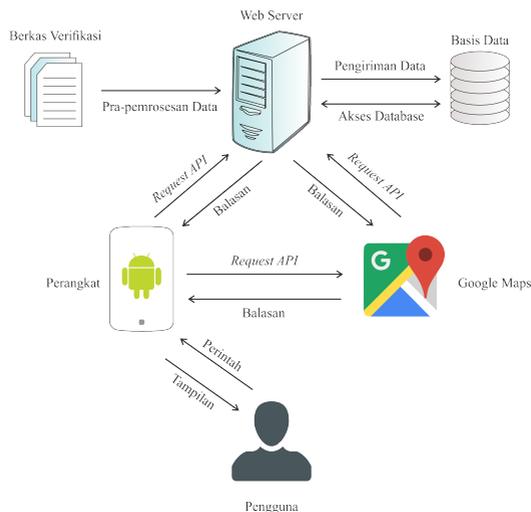
BAB 3

DESAIN

Tugas akhir ini dilaksanakan sesuai dengan desain sistem berikut. Desain sistem merupakan konsep dari pembuatan dan perancangan infrastruktur dan kemudian diwujudkan dalam bentuk blok-blok alur yang harus dikerjakan.

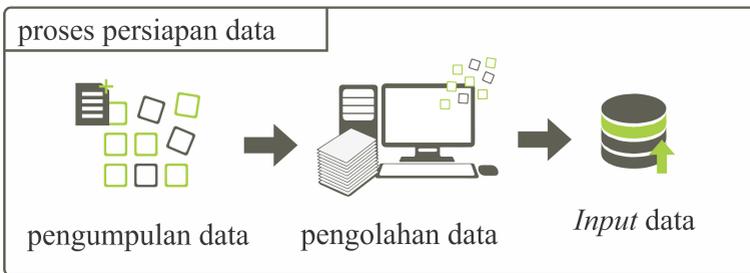
3.1 Desain Sistem

Tugas akhir ini bertujuan untuk mengembangkan sebuah aplikasi yang dapat menyediakan informasi tentang bangunan, serta menyediakan suatu wadah sebagai penampung data hasil pemeriksaan bangunan, sehingga data tersebut dapat diolah menjadi suatu informasi yang terstruktur. Pengerjaan aplikasi pada tugas akhir ini berbasis perangkat lunak *Android Studio* dengan *application program interface* yang dibangun menggunakan bahasa *php*. Gambaran umum sistem kerja pada tugas akhir ini dapat dilihat pada gambar 3.1.



Gambar 3.1: Gambaran umum sistem kerja

Sebelum data bangunan dapat digunakan sebagai acuan dalam proses pemeriksaan bangunan, data tersebut harus melalui proses pengumpulan terlebih dahulu. Proses tersebut dikerjakan oleh petugas administrasi Dinas Cipta Karya dan dilakukan pada saat pemilik bangunan telah melengkapi seluruh berkas pendaftaran Surat permohonan IMB dan menyerahkan berkas pendaftaran. Setelah berkas terkumpul, dilakukan proses pengubahan data dari bentuk fisik ke dalam bentuk digital. Setelah itu, data disimpan di dalam sebuah basis data yang ditempatkan pada *public server*, dengan demikian, data siap untuk dipergunakan pada proses selanjutnya. Alur proses persiapan data ditunjukkan pada gambar 3.2.



Gambar 3.2: Alur kerja proses persiapan data

Setelah data siap digunakan, pengguna dapat mengakses data bangunan melalui aplikasi. Proses penerimaan data dilakukan dengan cara memanggil *HTTP API* yang disediakan oleh *Web Server*. *API* yang dipanggil akan memberikan balasan berupa *JSON*. *JSON* tersebut kemudian diuraikan menjadi sebuah *Java Object*. Objek-objek java dari bangunan kemudian ditampilkan ke dalam dua jenis tampilan. Kedua jenis tersebut berupa :

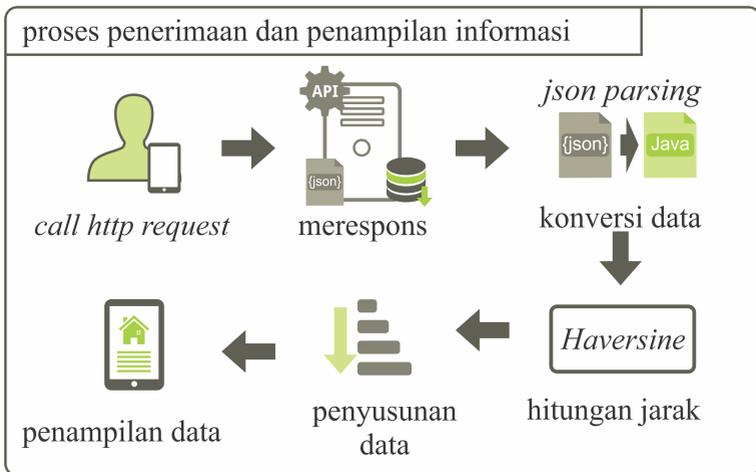
1. Tampilan Daftar

Pada tampilan ini informasi bangunan ditampilkan dalam bentuk tampilan daftar yang telah diurutkan berdasarkan jarak bangunan dengan jarak pengguna. Variabel-variabel bangunan dipetakan kedalam teks maupun gambar kemudian ditampilkan ke layar.

2. Tampilan Peta (*Google Maps*)

Pada tampilan ini informasi bangunan yang disajikan berupa marker yang disematkan berdasarkan letak bangunan dan status IMB.

Setelah pengguna memilih salah satu objek bangunan yang ingin diperiksa, pengguna akan diarahkan kepada sebuah tampilan yang memuat detail informasi bangunan yang dipilih. Dengan demikian tampilan tersebut dapat digunakan sebagai acuan dalam proses pemeriksaan bangunan. Alur kerja pada proses penerimaan data dapat dilihat pada gambar 3.3.



Gambar 3.3: Alur kerja proses penerimaan data

Proses pemeriksaan bangunan dilakukan dengan cara membandingkan kondisi fisik bangunan dengan informasi bangunan yang diberikan oleh aplikasi. Dari proses tersebut akan dihasilkan dua buah keluaran yang berupa :

1. Verifikasi Bangunan

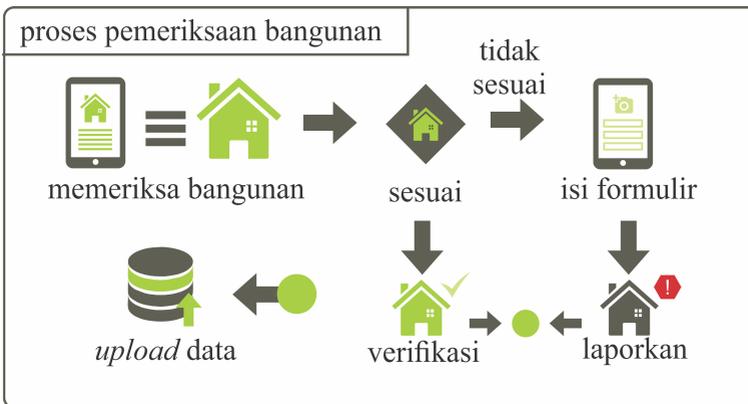
Tindakan verifikasi dilakukan pengguna apabila data fisik bangunan yang diperiksa telah sesuai dengan data yang ditampilkan pada aplikasi. Dengan melakukan verifikasi bangunan, pengguna akan membuat sebuah panggilan terhadap *HTTP*

API yang akan menginisiasi sebuah fungsi untuk memperbarui status bangunan dan menambahkan data verifikasi ke dalam basis data.

2. Pelaporan Bangunan

Tindakan pelaporan dilakukan pengguna apabila ditemukan ketidaksesuaian antara data fisik dengan data bangunan yang ditampilkan pada Aplikasi. Keadaan tersebut harus ditindaklanjuti dengan melaporkan bangunan yang melanggar aturan IMB. Oleh karena itu, petugas harus mengisi formulir pelaporan bangunan yang telah disediakan di dalam aplikasi dan mengirimkan laporan tersebut ke *Web Server*. Pada saat petugas mengirimkan laporan, aplikasi akan membuat sebuah panggilan terhadap *HTTP API* yang akan menginisiasi salah satu fungsi untuk memperbarui status bangunan dan menambahkan data laporan ke dalam basis data.

Alur kerja proses pemeriksaan bangunan dapat dilihat pada gambar 3.4.



Gambar 3.4: Alur kerja proses pemeriksaan bangunan

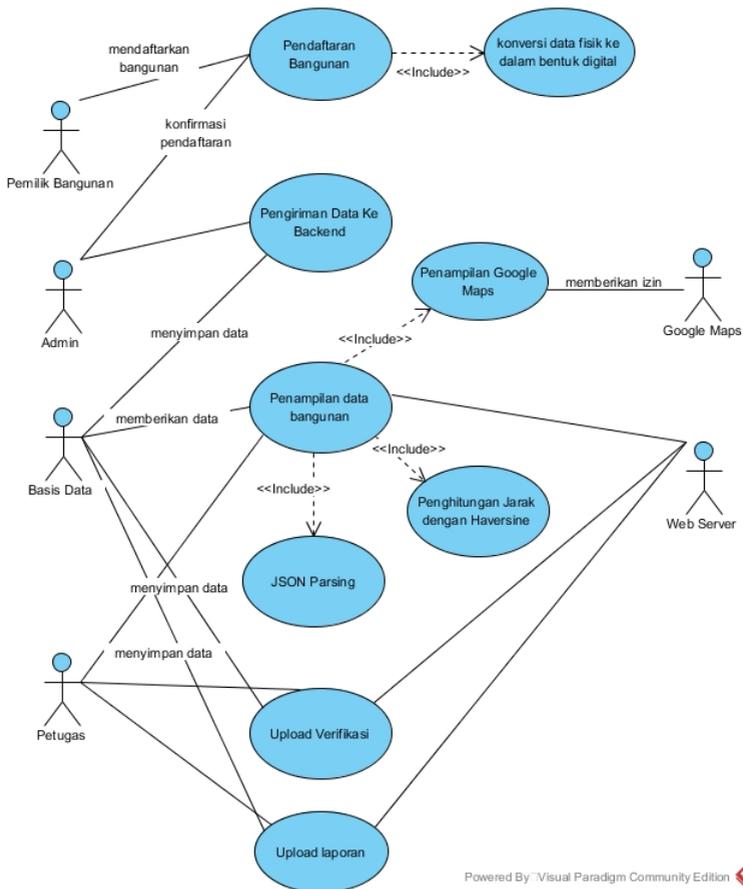
Data-data pelaporan dan verifikasi yang telah terkirim ke *server* melalui aplikasi, disimpan secara terstruktur di dalam sebuah basis data relasional yang telah disediakan.

3.1.1 Desain Aplikasi Berbasis UML

Desain sistem berdasarkan konsep berorientasi object pada tugas akhir ini akan dijelaskan dalam beberapa diagram berikut :

1. Use Case Diagram

Hubungan antara aktor dan *use case* yang ada dalam tugas akhir ini dapat dilihat pada diagram berikut :



Gambar 3.5: Use Case Diagram

penjelasan *Use case* diagram yang ditunjukkan pada gambar 3.5 disajikan pada tabel 3.1

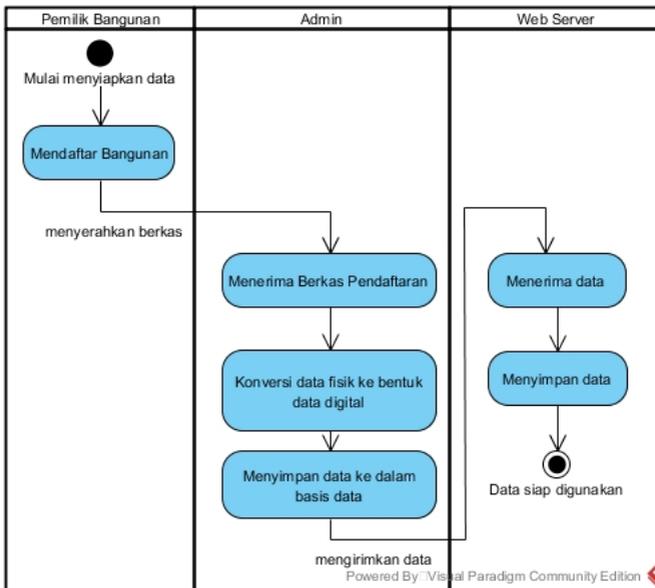
Tabel 3.1: Penjelasan *Use case* diagram

<i>Use case</i>	<i>Actor</i>	Deskripsi
Pendaftaran bangunan	pemilik bangunan, admin	<i>Use case</i> ini menjelaskan mengenai interaksi pemilik bangunan dengan admin pada saat pendaftaran bangunan
Pengiriman data ke <i>backend</i>	admin, basis data	<i>User case</i> ini menjelaskan tentang pemiraman data oleh admin ke dalam basis data
Penampilan data bangunan	petugas, <i>web server</i> , basis data	<i>Use case</i> ini menjelaskan tentang bagaimana data bangunan diambil dari basis data dan ditampilkan kepada pengguna
<i>Upload</i> verifikasi	petugas, <i>web server</i> , basis data	<i>Use case</i> ini menjelaskan interaksi antara petugas, <i>web server</i> dan basis data ketika petugas sedang melakukan tindakan verifikasi bangunan
<i>Upload</i> laporan	petugas, <i>web server</i> , basis data	<i>Use case</i> ini menjelaskan interaksi antara petugas, <i>web server</i> dan basis data ketika petugas sedang melakukan tindakan pelaporan bangunan
Penampilan <i>google maps</i>	<i>google maps server</i>	<i>Use case</i> ini menjelaskan tentang proses penampilan peta <i>google</i>

2. Activity Diagram

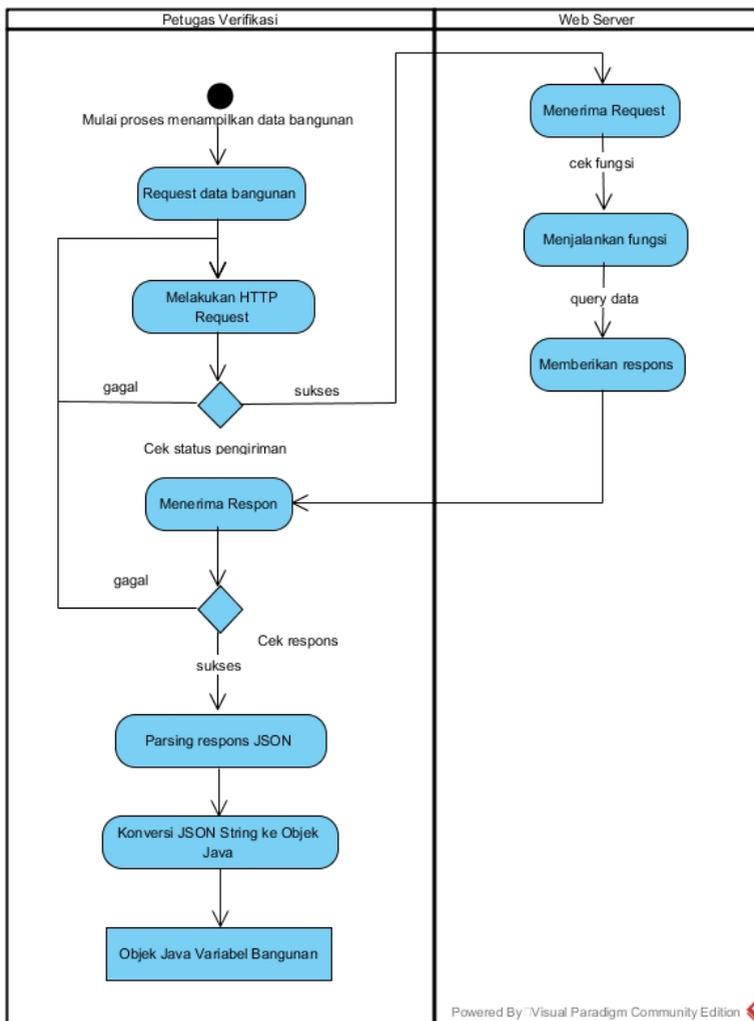
Activity diagram memberikan gambaran bagaimana sistem berinteraksi dengan pengguna dan/atau dengan sistem lain. Berikut merupakan *activity diagram* dari fungsi-fungsi yang ada pada tugas akhir ini. Aktivitas dimulai dari pengumpulan data oleh petugas administrasi. Kemudian dilakukan pra-pemrosesan data sebelum disimpan ke dalam basis data. Setelah data tersimpan di dalam basis data, pengguna dapat melakukan aktivitas pengambilan data dan penampilan informasi. Setelah data ditampilkan, pengguna dapat memilih salah satu aktivitas antara Pelaporan bangunan atau Verifikasi bangunan. Aktivitas-aktivitas tersebut ditunjukkan dalam diagram yang ditunjukkan pada gambar 3.6, 3.7, 3.8 dan 3.9 :

- (a) *Activity Diagram* untuk menjelaskan segala aktivitas yang dilakukan dalam proses Persiapan Data ditunjukkan pada gambar 3.6.

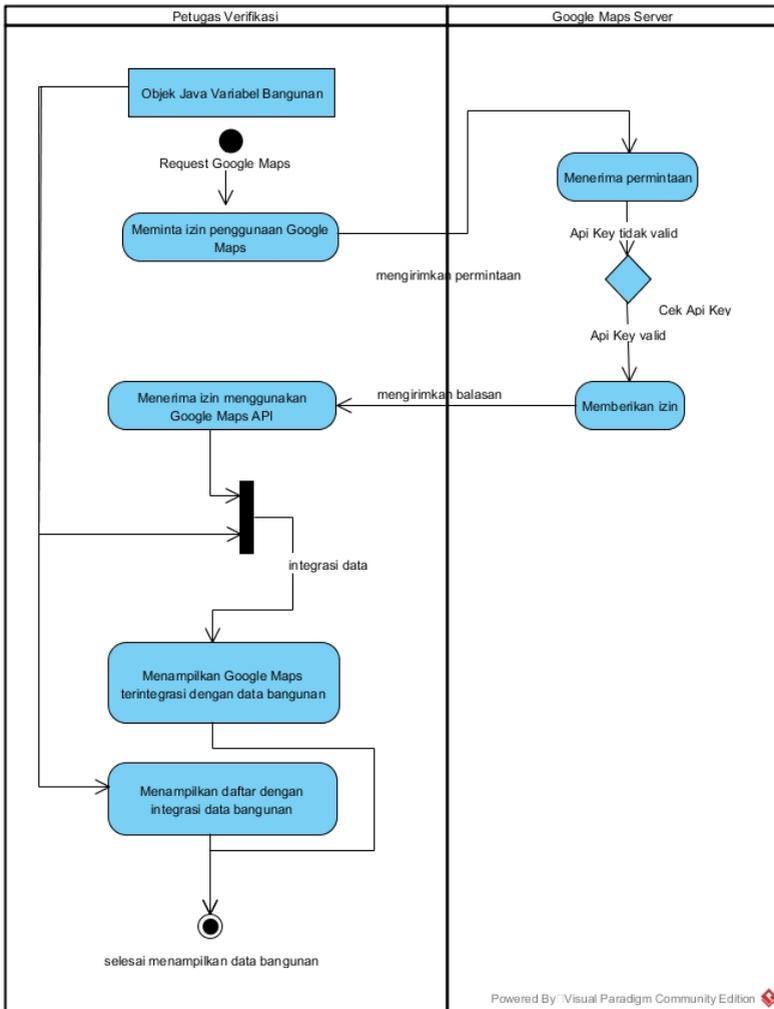


Gambar 3.6: Activity diagram persiapan data

(b) *Activity Diagram* untuk menjelaskan segala aktivitas yang dilakukan dalam Proses Penampilan Data ditunjukkan pada gambar 3.7.

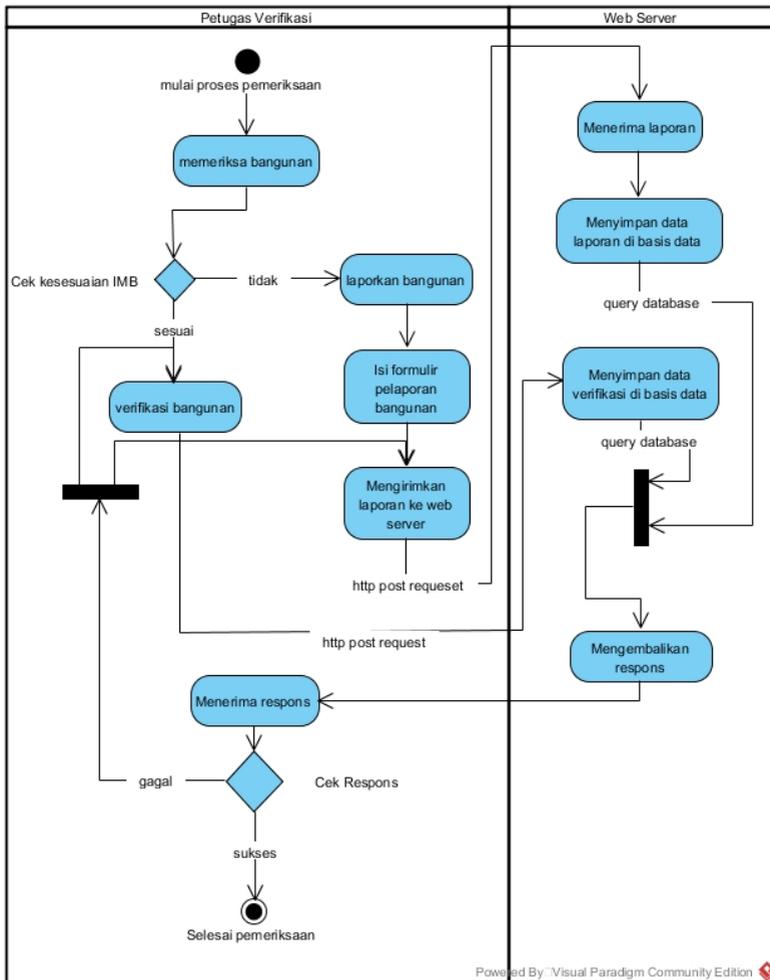


Gambar 3.7: Activity diagram persiapan data bagian 1



Gambar 3.8: Activity diagram persiapan data bagian 2

- (c) Activity Diagram untuk menjelaskan segala aktivitas yang dilakukan dalam Proses Pemeriksaan Bangunan ditunjukkan pada gambar 3.9.



Gambar 3.9: Activity diagram pemeriksaan bangunan

3.2 Perancangan Basis Data

Pada bagian ini akan dijelaskan mengenai data yang digunakan, struktur tabel dan skema relasi.

3.2.1 Data yang Digunakan

Data bangunan yang didapat dari berkas pendaftaran IMB tidak semuanya dipakai dalam proses pemeriksaan bangunan. Beberapa variabel yang digunakan beserta tipe datanya disajikan pada tabel 3.2.

Tabel 3.2: Variabel yang digunakan dalam proses pemeriksaan IMB

Nama Variabel	Keterangan
fid	Feature ID
nib	Nomor induk bangunan
nomor sk	Nomor urut SK bangunan
nomor persil	Nomor persil bangunan
nama site	Nama tempat kedudukan
pemilik	Nama pemilik
wilayah	Nama wilayah kota
ket imb	Keterangan status IMB
land use	Keperuntukan lahan
nama jalan	Nama jalan
gang	Gang rumah
nomor	Nomor rumah
kelurahan	Nama kelurahan
kecamatan	Nama kecamatan

3.2.2 Desain Aliran Data

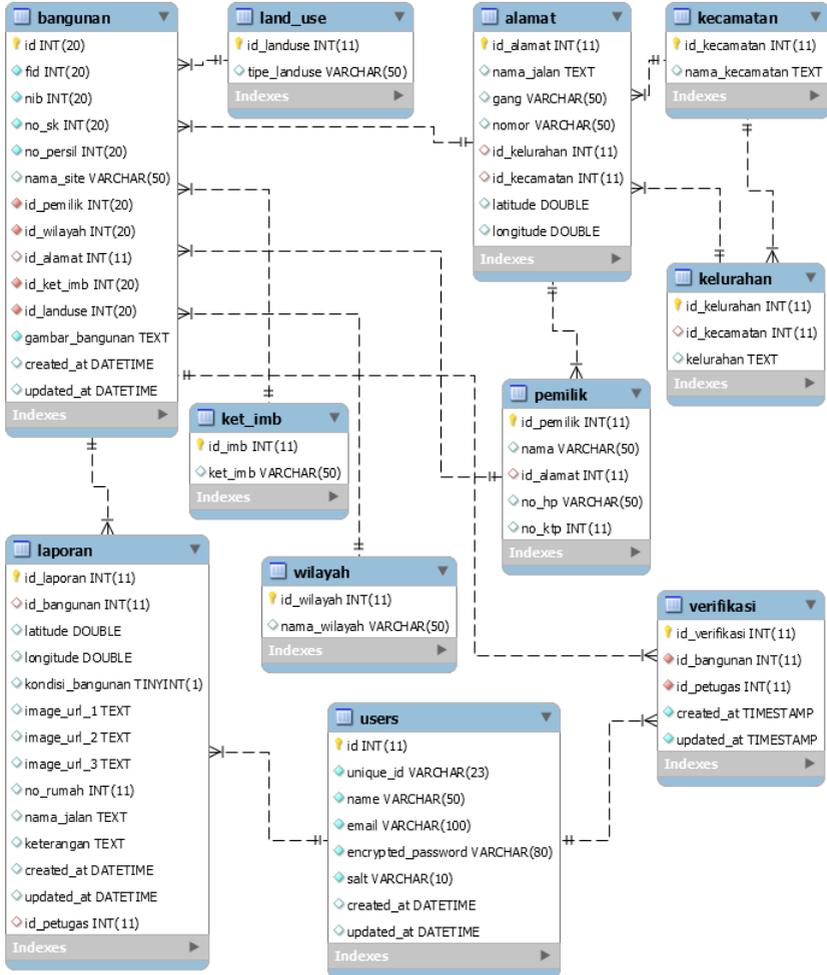
Desain aliran data merupakan penggambaran aliran data yang masuk dan keluar dari sistem. Aliran data pada tugas akhir ini dirancang menggunakan *Data Flow Diagram (DFD)* yang disajikan pada gambar 3.10.

1. Pendaftaran IMB
Pemilik menyerahkan berkas-berkas pendaftaran IMB yang memuat informasi bangunan kepada petugas administrasi.
2. Persiapan data
Berkas fisik yang terkumpul dari proses pendaftaran, diubah menjadi digital dan disimpan ke dalam basis data.
3. Penerimaan Data
Setelah berhasil melakukan login aplikasi, pengguna akan diarahkan kepada tampilan menu utama. Kemudian aplikasi akan melakukan permintaan data kepada web server dan mendapatkan balasan berupa informasi bangunan dalam format JSON.
4. Penampilan informasi bangunan
Pengguna mendapatkan tampilan informasi bangunan dari aplikasi android dengan mengakses salah satu bangunan yang tampil pada menu utama dan diarahkan menuju menu detail bangunan.
5. Pelaporan bangunan
Pengguna mengisi formulir laporan bangunan dan mengirimkan formulir tersebut ke dalam basis data.
6. Verifikasi bangunan
Pengguna memverifikasi bangunan dan mengirimkan data verifikasi.

3.2.3 Skema Relasi Basis Data

Skema relasi dari basis data yang dibangun pada tugas akhir ini disajikan pada gambar 3.11. Tabel bangunan berelasi dengan beberapa tabel yaitu tabel *landuse*(untuk data kegunaan lahan), alamat (untuk data alamat bangunan), pemilik (untuk data pemilik), *ket_imb* (untuk keterangan status IMB), verifikasi (untuk data verifikasi), laporan (untuk data laporan) dan wilayah (untuk data wilayah). Tabel alamat juga berelasi dengan beberapa tabel yaitu tabel kecamatan (untuk nama kecamatan), tabel kelurahan

(untuk nama kelurahan). Tabel verifikasi juga berelasi dengan beberapa tabel yaitu tabel *users* (untuk data petugas) dan bangunan (untuk data bangunan).



Gambar 3.11: Skema relasi basis data

3.2.4 Struktur Tabel

Tabel-tabel yang digunakan dalam tugas akhir ini terdiri dari 11 tabel. Berikut adalah penjelasan tentang tabel-tabel tersebut :

1. Tabel bangunan

Tabel bangunan digunakan untuk menyimpan data keterangan bangunan yang disusun berdasarkan tabel 3.2. Struktur tabel bangunan dapat dilihat pada tabel 3.3.

Tabel 3.3: Struktur tabel bangunan

Nama Field	Tipe Data	Ukuran	Keterangan
id	<i>int</i>	11	<i>ID</i> Bangunan
fid	<i>int</i>	11	<i>FID</i> Bangunan
nib	<i>int</i>	11	Nomor induk bangunan
no_sk	<i>int</i>	11	Nomor SK bangunan
no_persil	<i>int</i>	11	Nomor persil bangunan
nama_site	<i>varchar</i>	11	Nama Daerah
id_pemilik	<i>int</i>	11	<i>ID</i>

2. Tabel alamat

Tabel alamat digunakan untuk menyimpan data alamat bangunan yang diajukan oleh pemilik bangunan. Struktur tabel alamat dapat dilihat pada tabel 3.4.

3. Tabel kecamatan

Tabel kecamatan digunakan untuk menyimpan keterangan seluruh kecamatan yang ada di kota Surabaya. Struktur tabel kecamatan dapat dilihat pada tabel 3.5

Tabel 3.4: Struktur tabel alamat

Nama Field	Tipe Data	Ukuran	Keterangan
id_alamat	<i>int</i>	11	<i>ID</i> alamat
nama_jalan	<i>varchar</i>	50	Nama jalan bangunan
gang	<i>varchar</i>	50	Nama gang
nomor	<i>varchar</i>	50	Nomor rumah yang terdata
id_kelurahan	<i>int</i>	11	<i>ID</i> kelurahan
id_kecamatan	<i>int</i>	11	<i>ID</i> kecamatan
latitude	<i>double</i>	20	Koordinat garis lintang
longitude	<i>double</i>	20	Koordinat garis bujur

Tabel 3.5: Struktur tabel kecamatan

Nama Field	Tipe Data	Ukuran	Keterangan
id_kecamatan	<i>int</i>	11	<i>ID</i> Kecamatan
kecamatan	<i>varchar</i>	50	Nama Kecamatan

4. Tabel kelurahan

Tabel kelurahan digunakan untuk menyimpan keterangan seluruh kelurahan yang ada di kota Surabaya. Struktur tabel kelurahan dapat dilihat pada tabel 3.6

Tabel 3.6: Struktur tabel kelurahan

Nama Field	Tipe Data	Ukuran	Keterangan
id_kelurahan	<i>int</i>	11	<i>ID</i> Kelurahan
id_kecamatan	<i>int</i>	11	<i>ID</i> Kecamatan
kelurahan	<i>varchar</i>	50	Nama Kelurahan

5. Tabel ket_imb

Tabel ket_imb digunakan untuk menyimpan keterangan status IMB bangunan. Struktur tabel ket_imb dapat dilihat pada tabel 3.7.

Tabel 3.7: Struktur tabel ket_imb

Nama Field	Tipe Data	Ukuran	Keterangan
id_imb	<i>int</i>	11	<i>ID</i> Ket IMB
ket_imb	<i>varchar</i>	50	Keterangan IMB

6. Tabel land_use

Tabel land_use digunakan untuk menyimpan keterangan pemanfaatan bangunan. Struktur tabel land_use dapat dilihat pada tabel 3.8

Tabel 3.8: Struktur tabel land_use

Nama Field	Tipe Data	Ukuran	Keterangan
id_landuse	<i>int</i>	11	<i>ID Land Use</i>
tipe_landuse	<i>varchar</i>	50	Tipe penggunaan bangunan

7. Tabel pemilik

Tabel pemilik digunakan untuk menyimpan data identitas pemilik bangunan. Struktur tabel bangunan dapat dilihat pada tabel 3.9.

Tabel 3.9: Struktur tabel pemilik

Nama Field	Tipe Data	Ukuran	Keterangan
id_pemilik	<i>int</i>	11	<i>ID</i> Pemilik (primary key)
nama	<i>varchar</i>	50	Nama pemilik
id_alamat	<i>int</i>	11	<i>ID</i> alamat
no_hp	<i>varchar</i>	50	Nomor Hp pemilik
no_ktp	<i>int</i>	11	Nomor KTP pemilik

8. Tabel laporan

Tabel laporan digunakan untuk menyimpan hasil pelaporan petugas. Struktur tabel laporan dapat dilihat pada tabel 3.10.

Tabel 3.10: Struktur tabel laporan

Nama Field	Tipe Data	Ukuran	Keterangan
id_laporan	<i>int</i>	11	<i>ID</i> Laporan
id_bangunan	<i>int</i>	11	<i>ID</i> Bangunan
latitude	<i>double</i>	20	Latitude pengguna
longitude	<i>double</i>	20	Longitude pengguna
kondisi_bangunan	<i>tinyint</i>	1	Kondisi bangunan
image_url_1	<i>text</i>		<i>URL</i> dokumentasi-1
image_url_2	<i>text</i>		<i>URL</i> dokumentasi-2
image_url_3	<i>text</i>		<i>URL</i> dokumentasi-3
no_rumah	<i>int</i>	11	Nomor bangunan
nama_jalan	<i>text</i>		Nama jalan bangunan
keterangan	<i>text</i>		Berita Acara
created_at	<i>datetime</i>		Waktu data dibuat
updated_at	<i>datetime</i>		Waktu data diperbarui

9. Tabel users

Tabel user digunakan untuk menyimpan keterangan pengguna/petugas pengendalian. Struktur tabel user dapat dilihat pada tabel 3.11.

Tabel 3.11: Struktur tabel user

Nama Field	Tipe Data	Ukuran	Keterangan
id	<i>int</i>	11	<i>ID</i> User
unique_id	<i>varchar</i>	23	<i>Unique ID</i> user
name	<i>varchar</i>	50	Nama pengguna
email	<i>varchar</i>	100	Email pengguna
encrypted_password	<i>varchar</i>	80	Password terenkripsi
salt	<i>varchar</i>	10	<i>Salt</i>
created_at	<i>datetime</i>		Waktu data dibuat
updated_at	<i>datetime</i>		Waktu data diperbarui

10. Tabel verifikasi

Tabel verifikasi digunakan untuk menyimpan data hasil verifikasi bangunan. Struktur tabel verifikasi dapat dilihat pada tabel 3.12.

Tabel 3.12: Struktur tabel verifikasi

Nama Field	Tipe Data	Ukuran	Keterangan
id_verifikasi	<i>int</i>	11	<i>ID</i> Verifikasi
id_bangunan	<i>int</i>	11	<i>ID</i> Bangunan
id_petugas	<i>int</i>	11	<i>ID</i> pengguna
created_at	<i>datetime</i>		Waktu pembuatan field
updated_at	<i>datetime</i>		Waktu field diperbarui

11. Tabel wilayah

Tabel wilayah digunakan untuk menyimpan keterangan wilayah yang terdapat pada kota Surabaya. Struktur tabel wilayah dapat dilihat pada tabel 3.13.

Tabel 3.13: Struktur tabel wilayah

Nama Field	Tipe Data	Ukuran	Keterangan
id_wilayah	<i>int</i>	11	<i>ID</i> Wilayah
nama_wilayah	<i>varchar</i>	50	Nama wilayah

3.3 Desain Antarmuka Pengguna

Aplikasi yang dikembangkan pada tugas akhir ini memiliki desain antarmuka seperti yang akan dijelaskan pada bagian ini.

1. Tampilan *Splash*

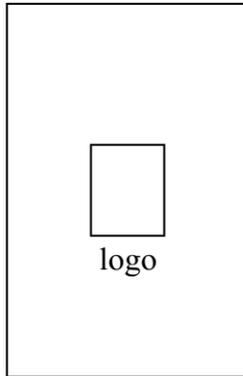
Tampilan splash merupakan tampilan paling awal pada aplikasi ini. Di dalam tampilan ini terdapat logo aplikasi yang bertujuan untuk menarik perhatian pengguna. Rancangan antarmuka splash disajikan pada gambar 3.12.

2. Tampilan *Login*

Tampilan login merupakan tampilan kedua setelah splash. Tampilan ini bertujuan untuk membatasi pengguna agar tidak semua pengguna dapat mengakses fitur yang dimiliki aplikasi. Rancangan antarmuka Login disajikan pada gambar 3.13a.

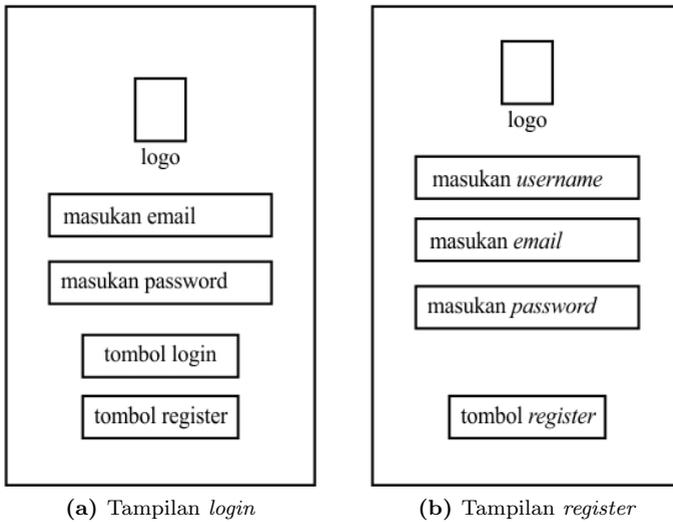
3. Tampilan *Register*

Tampilan register merupakan tampilan untuk mendaftarkan



Gambar 3.12: Tampilan *Splash*

pengguna baru agar mendapatkan hak akses fitur pada aplikasi ini. Rancangan antarmuka register disajikan pada gambar 3.13b.



(a) Tampilan *login*

(b) Tampilan *register*

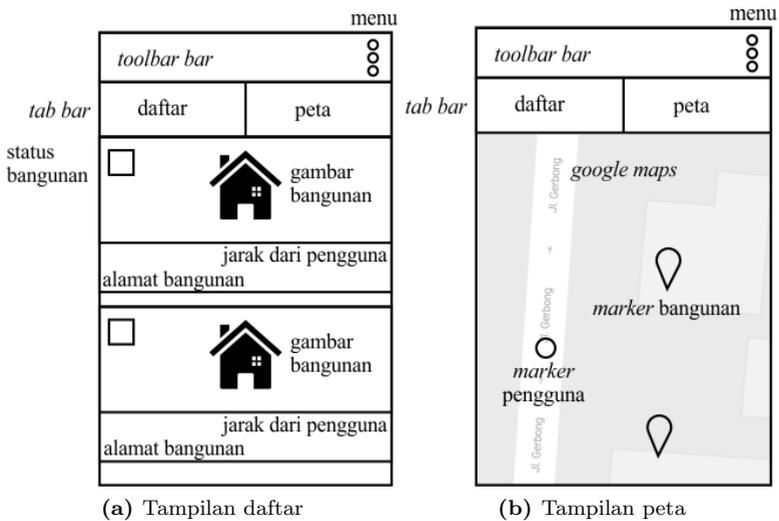
Gambar 3.13: Tampilan *login* dan *register*

4. Tampilan Menu Utama - Daftar

Tampilan daftar merupakan tampilan yang menampilkan informasi gambar bangunan, status IMB bangunan, jarak terhadap pengguna dan alamat bangunan. Rancangan antarmuka tampilan daftar disajikan pada gambar 3.14a.

5. Tampilan Menu Utama - Peta

Tampilan peta merupakan tampilan berbasis google maps yang memuat informasi lokasi pengguna dan informasi bangunan. Lokasi bangunan diwakili oleh marker berwarna untuk menandai peta berdasarkan koordinat geografis dan warna tertentu untuk menunjukkan status IMB. Rancangan tampilan peta disajikan pada gambar 3.14b.



Gambar 3.14: Tampilan menu utama aplikasi

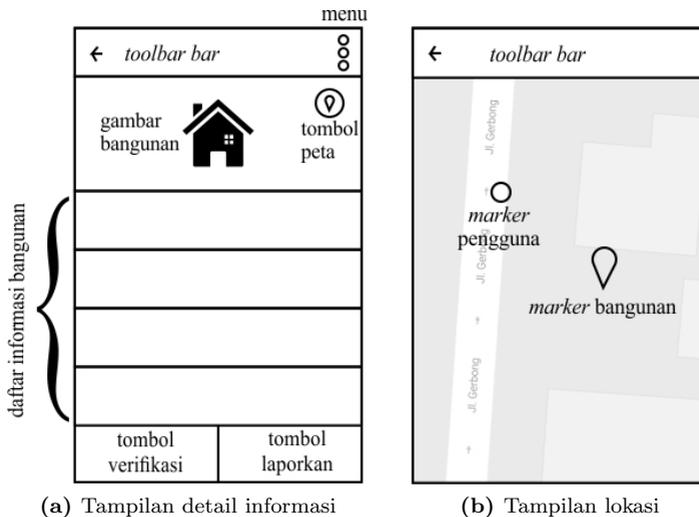
6. Tampilan Detail Bangunan

Tampilan detail bangunan merupakan tampilan yang memuat informasi detail bangunan berdasarkan IMB seperti nib, gambar bangunan, nomor sk, nomor persil dan lain-lain. Selain informasi bangunan terdapat tombol lihat peta, tombol verifika-

si dan tombol laporkan. Rancangan tampilan detail bangunan disajikan pada gambar 3.15a.

7. Tampilan Lokasi Bangunan

Tampilan lokasi bangunan merupakan tampilan yang bertujuan memberikan informasi letak bangunan yang ditunjukkan oleh marker diatas peta dan menampilkan lokasi pengguna dengan disimbolkan lingkaran berwarna biru, sehingga pengguna dapat memperkirakan letak bangunan dari posisi ia berdiri. Tampilan ini dapat diakses pengguna dengan cara menekan tombol lihat peta pada tampilan detail bangunan. Rancangan tampilan lokasi bangunan disajikan pada gambar 3.15b.



Gambar 3.15: Tampilan detail informasi dan lokasi bangunan

8. Tampilan Laporkan Bangunan

Tampilan laporkan Bangunan merupakan sebuah tampilan yang bertujuan untuk memberikan laporan terhadap bangunan yang tidak sesuai data IMB maupun belum memiliki IMB. Pada tampilan ini pengguna harus melampirkan foto bangunan yang diperiksa dan mengisi formulir yang tertera sesuai

kondisi sebenarnya. Pengguna dapat mengakses tampilan ini dengan menekan tombol laporkan bangunan pada tampilan detail bangunan. Rancangan tampilan laporkan bangunan disajikan pada gambar 3.16.

The diagram illustrates a vertical form layout. At the top is a toolbar bar containing a left-pointing arrow and the text "toolbar bar". Below this is a row of three camera icons, each with a plus sign in the top-left corner, labeled "tombol tambah dokumentasi". A large curly bracket on the left side of the form encompasses the three empty rows below the document icons, with the label "formulir laporan" written vertically next to it. At the bottom of the form is a row containing the text "tombol upload laporan".

Gambar 3.16: Tampilan formulir laporan

Halaman ini sengaja dikosongkan

BAB 4

IMPLEMENTASI DAN HASIL

4.1 Implementasi

Pada bab ini dijelaskan mengenai implementasi program berdasarkan proses kerja sistem.

4.1.1 Persiapan Data

Berkas bangunan harus melewati beberapa tahapan sebelum disajikan kepada pengguna. Tahapan-tahapan tersebut yaitu :

1. Pengumpulan data

Data awal didapatkan dari berkas-berkas pendaftaran IMB yang telah diajukan pemilik bangunan kepada petugas administrasi Dinas Cipta Karya dan Tata Ruang.

2. Pra-pemrosesan data

Setelah semua berkas pendaftaran terkumpul, dilakukan proses pengubahan bentuk data dari bentuk fisik ke dalam bentuk digital.

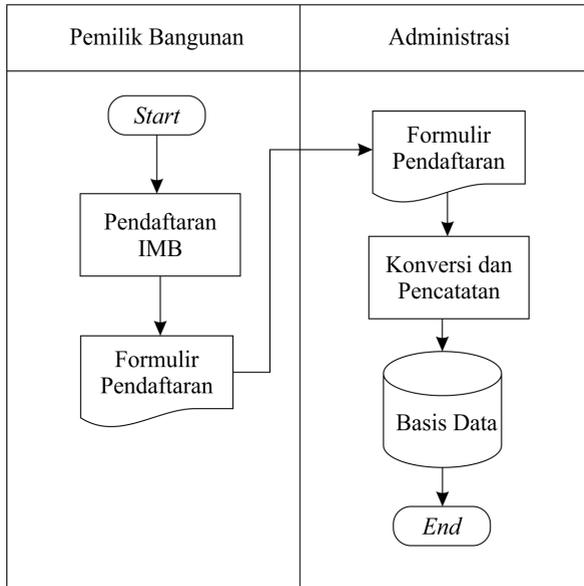
3. Pengiriman data ke basis data

Data digital yang telah siap akan disimpan ke dalam basis data relasional yang terdapat pada *public server*.

Algoritma proses persiapan data disajikan pada gambar 4.1.

4.1.2 Penerimaan Data

Pengguna melakukan *request* terhadap *HTTP API* yang telah tersedia di dalam *Web server*. Selanjutnya *Web server* akan memberikan balasan berupa *JSON* sesuai dengan *HTTP API* yang dipanggil. Setelah mendapatkan respons tersebut, dilakukan proses penguraian dari bentuk *JSON* ke dalam bentuk objek *java*. Proses penguraian tersebut dilakukan dengan menggunakan pustaka sehingga *JSON* dapat langsung tersimpan di dalam *generic list*. Variabel-variabel bangunan yang dimiliki *JSON* akan dipetakan secara otomatis ke dalam atribut-atribut yang dimiliki Objek.



Gambar 4.1: Algoritma proses persiapan data

Pada proses penerimaan data, diperlukan beberapa implementasi dari pustaka-pustaka berikut :

1. *com.squareup.okhttp3:logging-interceptor:3.0.1*
Pustaka ini digunakan untuk membuat sebuah *Interceptors* dalam suatu percobaan pemanggilan terhadap *HTTP API* tertentu dan membuat sebuah *HTTP client* berbasis *OKHTTP*.
2. *com.squareup.retrofit2:retrofit:2.0.2'*
Pustaka ini digunakan untuk mengubah suatu *HTTP API* ke dalam sebuah *interface*.
3. *com.squareup.retrofit2:converter-gson:2.0.2'*
Pustaka ini digunakan untuk mengubah *JSON String* ke dalam suatu objek java.

Aplikasi *android* mengimplementasikan beberapa pustaka diatas dengan cara menambahkan kode seperti yang ditunjukkan pada kode

4.1 ke dalam kode *java*.

Kode 4.1 *Pseudocode* proses penerimaan data

```
1: begin function GetDataBangunan(koneksi, URLGetBangunanAPI):
2: if koneksi == 'OK', then:
3:   HTTPpostRequest(URLGetBangunanAPI)
4:   if respons == 'OK', then:
5:     ParseJSON(respons.body())
6:   else:
7:     print "server tidak merespons"
8:   end if
9: else:
10:  print "tidak ada koneksi internet"
11: end if
12: end function
```

4.1.3 Penghitungan Jarak Banguann

Jarak antara lokasi bangunan terhadap lokasi pengguna perlu diketahui untuk difungsikan sebagai acuan pengurutan daftar bangunan yang akan ditampilkan kepada pengguna, sehingga daftar informasi bangunan akan ditampilkan berurutan berdasarkan jarak terdekat dengan pengguna.

Jarak tersebut dapat dihitung menggunakan fungsi haversine dengan masukan berupa koordinat geografis bangunan dan pengguna.

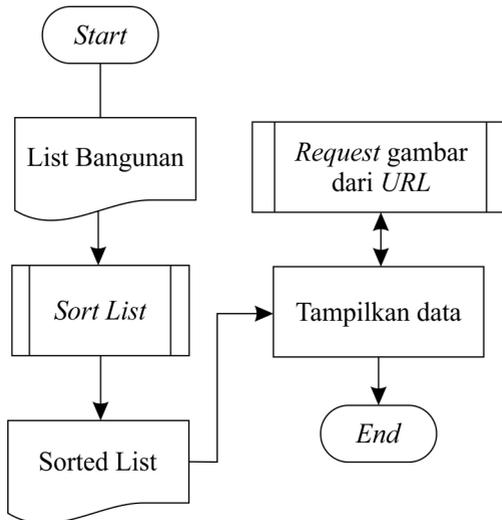
Kode 4.2 *Pseudocode* proses pernghitungan jarak

```
1: function radians(degree) #degrees to radians
2: rad = degree .* pi / 180
3: end
4:
5: function haversine(lat1, lon1, lat2, lon2)
6: dlat = radians(lat2-lat1)
7: dlon = radians(lon2-lon1)
8: lat1 = radians(lat1)
9: lat2 = radians(lat2)
10: a = (sin(dlat./2)).^2 + cos(lat1).*cos(lat2).*(sin(dlon./2)).^2
11: c = 2 .* asin(sqrt(a)) %jarak dalam meter
12: end
13:
14: begin function HitungJarak(lat_user, lon_user, list_bangunan):
15: lat_bangunan = list_bangunan.getLatitude()
16: lon_bangunan = list_bangunan.getLongitude()
17: haversine(lat_user, lon_user, lat_bangunan, lon_bangunan)
18: end function
```

Jarak yang didapat kemudian disimpan kembali ke dalam list berdasarkan posisi bangunan masing-masing.

4.1.4 Penampilan Data Bangunan

Data-data bangunan yang telah tersimpan pada generic list kemudian dipetakan ke dalam suatu objek penampil teks di dalam sebuah *layout*. Algoritma proses penampilan data disajikan pada gambar 4.2.



Gambar 4.2: Algoritma proses persiapan data

4.1.5 Penyusunan Data Bangunan

Sebelum data yang tersimpan di dalam list ditampilkan kepada pengguna, dilakukan proses penyusunan data bangunan berdasarkan jarak terdekat bangunan dengan pengguna. Proses penyusunan data dilakukan dengan menerapkan fungsi *Comparator* untuk membandingkan jarak dan mengembalikan urutannya berdasarkan jarak terkecil, sehingga data di dalam list bangunan tersusun kembali berdasarkan jarak bangunan terhadap jarak pengguna.

Kode 4.3 *Pseudocode* proses penyusunan data bangunan

```

1: begin function compareBangunan(List Bangunan):
2:   for i from 1 to N % N = List bangunan's size
3:     for j from 0 to N - 1
4:       if a[j] > a[j + 1]
5:         swap( a[j], a[j + 1] )
6:   end func

```

4.1.6 Penampilan gambar bangunan

Pada proses ini, Aplikasi perlu mengakses gambar yang berada pada server dengan cara mengunduh gambar melalui *URL*. Untuk melakukan proses tersebut, diperlukan sebuah implementasi pustaka yaitu *Picasso*. Pustaka tersebut bertugas menangani pemanggilan file gambar berdasarkan *URL*, mengatur ukuran gambar dan menampilkan gambar ke dalam sebuah *ImageView* pada *android*. Pseudocode proses penampilan gambar disajikan pada kode 4.4.

Kode 4.4 *Pseudocode* proses penampilan gambar

```

1: begin function PenampilGambar(koneksi , URL gambar):
2:   if koneksi == 'OK', then:
3:     getGambar(URL gambar)
4:     if gambar != null , then:
5:       ShowGambar(gambar)
6:     else:
7:       ShowGambar(null)
8:     end if
9:   else
10:    print 'tidak ada koneksi internet'
11:  end if
12: end function

```

4.1.7 Pemeriksaan Bangunan

Proses pemeriksaan bangunan memiliki dua buah keluaran. Keluaran tersebut yaitu verifikasi bangunan dan laporkan bangunan. Algoritma proses pemeriksaan bangunan disajikan pada kode 4.5.

Kode 4.5 *Pseudocode* proses pemeriksaan bangunan

```

1: begin function PeriksaBangunan(bangunan):
2:   if info bangunan == kondisi fisik bangunan, then :
3:     verifikasi(bangunan)
4:   else :
5:     laporkan(bangunan)
6:   end if
7: end function

```

4.1.8 Verifikasi Bangunan

Proses Verifikasi bangunan dilakukan dengan memanggil HTTP API verifikasi yang akan menjalankan fungsi untuk menambahkan data pada tabel verifikasi dan mengubah status IMB pada tabel bangunan. Algoritma proses verifikasi bangunan disajikan pada kode 4.6.

Kode 4.6 *Pseudocode* proses verifikasi bangunan

```
1: begin function verifikasi(bangunan):
2:   if koneksi == 'OK'
3:     HTTP POST request (API VerifikasiBangunan)
4:     if respons == 'OK', then:
5:       print 'sukses upload verifikasi'
6:     else:
7:       print 'server tidak merespons'
8:     end if
9:   else
10:    print 'tidak ada koneksi internet'
11:  end if
12: end function
```

4.1.9 Laporkan Bangunan

Proses laporkan bangunan dilakukan dengan melengkapi formulir laporan dan mengambil gambar bangunan, dilanjutkan dengan menjalankan fungsi HTTP POST Request terhadap HTTP API dengan menyisipkan variabel-variabel laporan tersebut pada *body*. API tersebut akan menjalankan fungsi untuk menambahkan data pada tabel verifikasi dan menyimpan gambar ke dalam server. Algoritma proses laporkan bangunan disajikan pada kode 4.7.

Kode 4.7 *Pseudocode* proses laporkan bangunan

```
1: begin function laporkan(formulir):
2:   if koneksi == 'OK'
3:     HTTP POST request (API LaporkanBangunan, formulir)
4:     if respons == 'OK', then:
5:       print 'sukses upload laporan'
6:     else:
7:       print 'server tidak merespons'
8:     end if
9:   else
10:    print 'tidak ada koneksi internet'
11:  end if
12: end function
```

4.2 Hasil

Pada bab ini dijelaskan mengenai hasil akhir yang didapat dalam pengerjaan tugas akhir ini.

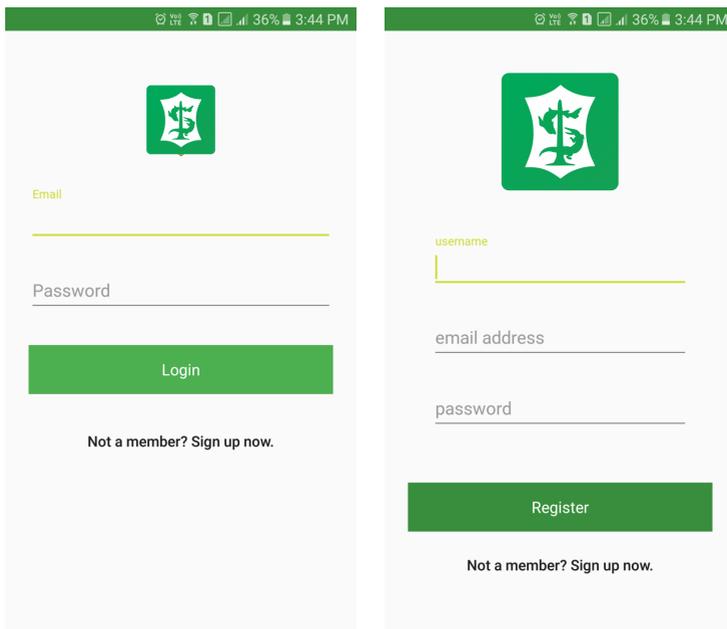
4.2.1 Hasil *User Interface*

1. Tampilan *Login*

Tampilan login berisikan logo, dua buah kolom isian untuk teks dan dua buah tombol yaitu login dan register. Tampilan login disajikan pada gambar 4.3a.

2. Tampilan *Register*

Tampilan register berisikan logo, tiga buah kolom isian untuk teks dan dua buah tombol yaitu register dan login. Tampilan register disajikan pada gambar 4.3b



(a) Tampilan *login*

(b) Tampilan *register*

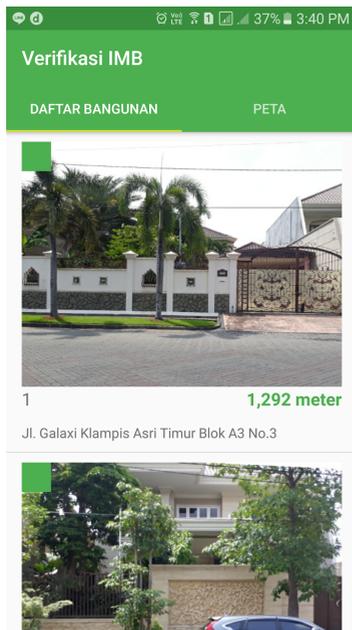
Gambar 4.3: Tampilan *login* dan *register*

3. Tampilan menu utama

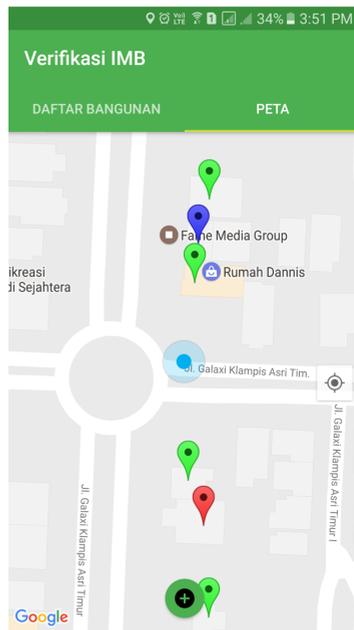
(a) Tampilan list

Pada tampilan list ditunjukkan informasi gambar bangunan, jarak bangunan berdasarkan lokasi pengguna, alamat bangunan, serta status IMB suatu bangunan. Tampilan list disajikan pada gambar 4.4a.

(b) Tampilan peta Pada tampilan peta ditunjukkan informasi letak bangunan yang diproyeksikan sebagai sebuah marker berwarna yang mana warna tersebut menandakan status bangunan serta menampilkan lokasi pengguna. Tampilan peta disajikan pada gambar 4.4b.



(a) Tampilan Daftar

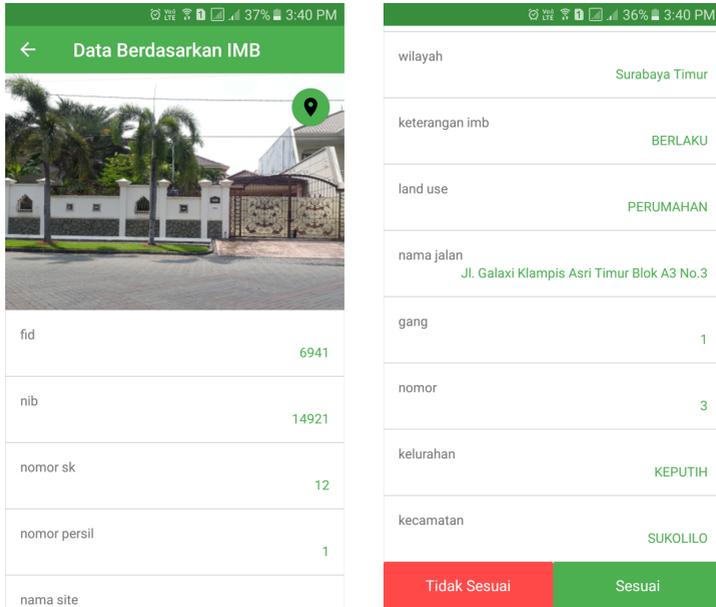


(b) Tampilan Peta

Gambar 4.4: Tampilan Menu Utama dan *register*

4. Detail bangunan

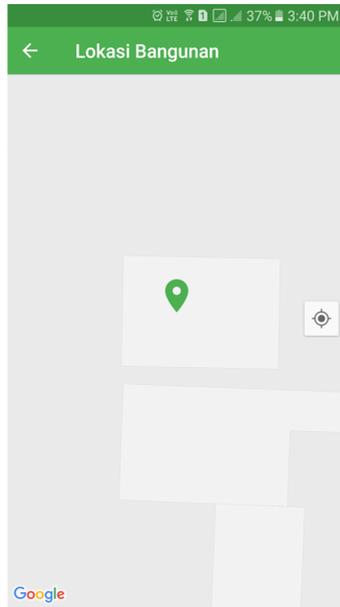
Tampilan detail bangunan memuat gambar bangunan serta informasi informasi bangunan yang ditampilkan menggunakan teks berbaris yang mana setiap baris memuat variabel yang berbeda. Tampilan detail bangunan disajikan pada gambar 4.5.



Gambar 4.5: Detail bangunan

5. Lokasi bangunan

Tampilan lokasi bangunan merupakan sebuah tampilan yang memberikan informasi lokasi dari suatu bangunan tertentu dan informasi lokasi pengguna. Tampilan lokasi bangunan disajikan pada gambar 4.6.



Gambar 4.6: Lokasi bangunan

6. Pelaporan bangunan

Tampilan pelaporan bangunan berisikan formulir pelaporan yang diisi oleh petugas. Isi formulir tersebut adalah gambar, status bangunan, nomor rumah, jalan, dan berita acara. Pengguna dapat mengambil gambar bangunan dengan menekan tombol dengan simbol kamera. Selain itu, pengguna dapat memilih status bangunan dengan memilih pilihan yang ditawarkan oleh spinner status bangunan. Pengguna juga dapat memberikan informasi nomor rumah, alamat, serta berita

acara dengan mengisi kolom isian teks yang tersedia di dalam formulir. Tombol upload data tersedia di akhir formulir yang berfungsi untuk memanggil *API* laporkan bangunan yang tersedia di dalam server yang berfungsi untuk menyimpan isian formulir yang telah diisi oleh petugas.

The image shows a mobile application interface for building verification. At the top, there is a green header with a back arrow and the title "Verifikasi Bangunan". Below the header, the form is organized into several sections:

- Tambahkan Foto Bangunan**: A section with three camera icons, indicating where to upload photos of the building.
- Status Bangunan**: A dropdown menu currently showing "Belum berdiri".
- Jalan**: A text input field with the placeholder "tuliskan nama jalan".
- Nomor**: A text input field with the placeholder "tuliskan nomor rumah".
- Berita Acara**: A text input field with the placeholder "tuliskan apa yang ingin anda laporkan".

At the bottom of the form is a prominent green button labeled "Upload Data Verifikasi". The top status bar of the phone shows the time as 3:41 PM and a battery level of 36%.

Gambar 4.7: Formulir pelaporan bangunan

Halaman ini sengaja dikosongkan

BAB 5

PENGUJIAN

Pada tugas akhir ini, dilakukan pengujian terhadap sistem yang sudah dibuat. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Terdapat empat buah pengujian yang dilakukan pada tugas akhir ini yaitu pengujian Sistem yang berjalan, pengujian fungsional aplikasi, pengujian pemanggilan *API* menggunakan beberapa tipe jaringan internet yang berbeda dan pengujian akurasi sensor yang digunakan yaitu sensor *GPS*.

5.1 Pengujian Sistem

Pengujian sistem yang dilakukan adalah pengujian *Betha* yang dilakukan secara objektif. Pengujian dilakukan dengan kuesioner yang disusun berdasarkan faktor-faktor jaminan kualitas perangkat lunak. Pengujian dilakukan kepada 29 pengguna yang terbagi ke dalam 2 tipe pengguna yaitu 17 mahasiswa dan 12 petugas. Dari hasil pengujian kuesioner tersebut, dilakukan perhitungan untuk dapat menentukan kesimpulan terhadap penilaian aplikasi yang dibangun.

Kuesioner untuk pengguna terdiri atas 10 pertanyaan yang disusun berdasarkan faktor-faktor yang berkaitan dengan sifat-sifat operasional perangkat lunak, yaitu faktor-faktor yang menunjukkan atribut kualitas produk dilihat dari sudut pandang pengguna (Ketergunaan, Kebenaran dan Keandalan), dengan 5 pilihan jawaban, yaitu Sangat Tidak Setuju (STS), Tidak Setuju (TS), Kurang Setuju (KS), Setuju (S), dan Sangat Setuju (SS). Terdapat perbedaan isi kuesioner yang diberikan kepada mahasiswa, pada mahasiswa pertanyaan yang diberikan hanya pertanyaan seputar ketergunaan. Berikut adalah daftar pertanyaan pada kuesioner:

1. *Usability*

- (a) Warna *background* dan tulisan yang ditampilkan sudah sesuai.
- (b) Ukuran gambar sudah sesuai.

- (c) Ukuran tulisan sudah sesuai.
- (d) Tata letak tombol serta menu menu yang disediakan sudah sesuai.
- (e) Data yang ditampilkan sudah jelas dan informatif.
- (f) Seberapa mudah menggunakan aplikasi ini.

2. *Correctness*

- (a) Data-data yang ditampilkan sudah sesuai dengan kebutuhan petugas pengendalian.
- (b) Parameter Laporan yang diisikan sudah sesuai.
- (c) Seberapa sering anda akan menggunakan aplikasi ini.

3. *Reliability*

- (a) Data-data yang ditampilkan di dalam aplikasi sudah sesuai dengan data IMB yang terdaftar.

Hasil pengujian *betha* kepada pengguna disajikan pada tabel 5.1 dan tabel 5.2.

Tabel 5.1: Hasil Pengujian *Betha* Kepada Mahasiswa

No	Pertanyaan	STS	TS	KS	S	SS
Ketergunaan						
1	Warna <i>background</i> dan tulisan yang ditampilkan sudah sesuai				12	5
2	Ukuran gambar sudah sesuai		1	1	9	6
3	Ukuran tulisan sudah sesuai			1	10	5
4	Tata letak tombol serta menu menu yang disediakan sudah sesuai			2	11	4
5	Data yang ditampilkan sudah jelas dan informatif			1	13	3
6	Seberapa mudah menggunakan aplikasi ini			3	12	2

Tabel 5.2: Hasil Pengujian *Betha* Kepada Petugas

No	Pertanyaan	STS	TS	KS	S	SS
Ketergunaan						
1	Warna <i>background</i> dan tulisan yang ditampilkan sudah sesuai	4			8	
2	Ukuran gambar sudah sesuai			2	10	
3	Ukuran tulisan sudah sesuai			2	10	
3	Tata letak tombol serta menu menu yang disediakan sudah sesuai	4		1	6	1
4	Data yang ditampilkan sudah jelas dan informatif	2		3	5	2
5	Seberapa mudah menggunakan aplikasi ini			4	3	5
Kebenaran						
7	Data-data yang ditampilkan sudah sesuai dengan kebutuhan petugas pengendalian	1		3	6	2
8	Parameter Laporan yang diisikan sudah sesuai	1		2	5	4
9	Seberapa sering anda akan menggunakan aplikasi ini		1	2	6	3
Keandalan						
10	Data-data yang ditampilkan di dalam aplikasi sudah sesuai dengan data IMB yang terdaftar		1	3	6	2

Respons yang didapat dari kuesioner yang diberikan kepada mahasiswa dan petugas dihitung menggunakan skala Linkert dengan bobot penilaian yang berbeda untuk masing-masing pilihan jawaban. Bobot untuk masing-masing pilihan jawaban ditunjukkan

pada tabel 5.3.

Data survey yang didapat kemudian dicari bobot masing-masing pertanyaan. Untuk mendapatkan hasil interpretasi, data diolah menggunakan rumus yang ditunjukkan pada persamaan 5.1 dan 5.2. Hasil interpretasi nilai berdasarkan metode linkert pada data perolehan survey disajikan pada tabel 5.4 dan 5.5.

$$Nilai = TotalSkor/Nilaimaks * 100\% \quad (5.1)$$

$$Nilaimaks = SkorTertinggiLinkert * jumlahresponden \quad (5.2)$$

Tabel 5.3: Bobot penilaian berdasarkan pilihan jawaban

Jawaban	Bobot
SS	5
S	4
KS	3
ST	2
SST	1

Tabel 5.4: Hasil penghitungan nilai dengan metode Linkert

	STS	TS	KS	S	SS	Total Skor	Nilai Max	Hasil
Pertanyaan 1	0	0	0	48	25	73	85	85,9%
Pertanyaan 2	0	2	3	36	30	71	85	83,5%
Pertanyaan 3	0	0	3	40	25	68	85	80,0%
Pertanyaan 4	0	0	6	44	20	70	85	82,4%
Pertanyaan 5	0	0	3	52	15	70	85	82,4%
Pertanyaan 6	0	0	9	48	10	67	85	78,8%
Rata-rata skor Ketergunaan								82,2%

Hasil yang disajikan pada tabel 5.4 menunjukkan 82% dari total responden mahasiswa setuju apabila aplikasi telah memenuhi unsur ketergunaan.

Tabel 5.5: Hasil penghitungan nilai dengan metode Linkert

	STS	TS	KS	S	SS	Total Skor	Nilai Max	Hasil
Pertanyaan 1	4	0	0	32	0	36	60	60%
Pertanyaan 2	0	0	6	40	0	46	60	76.7%
Pertanyaan 3	0	0	6	40	0	46	60	76.7%
Pertanyaan 4	4	0	3	24	5	36	60	60%
Pertanyaan 5	2	0	9	20	10	41	60	68.3%
Pertanyaan 6	0	0	12	12	25	49	60	81.7%
Rata-rata skor Ketergunaan								70.6%
Pertanyaan 7	1	0	9	24	10	44	60	73.3%
Pertanyaan 8	1	0	6	20	20	47	60	78.3%
Pertanyaan 9	0	2	6	24	15	47	60	78.3%
Rata-rata skor Kebenaran								76.7%
Pertanyaan 10	0	2	9	24	10	45	60	75%
Rata-rata skor keandalan								75%

Hasil kuesioner yang disajikan pada tabel 5.5 menunjukkan 70.6% dari total responden setuju apabila aplikasi telah memenuhi unsur ketergunaan. Selain itu, ada 76.7% responden setuju apabila aplikasi telah memenuhi unsur kebenaran. Sedangkan pada Keandalan setidaknya 75% responden setuju bahwa aplikasi telah memenuhi unsur keandalan.

5.2 Pengujian Fungsional Aplikasi

Pengujian perangkat lunak ini menggunakan metode pengujian *Black Box*. Pengujian *Black Box* digunakan untuk menguji fungsi-fungsi khusus dari aplikasi perangkat lunak yang dirancang. Adapun hasil pengujian sistem menggunakan teknik pengujian *Black Box* disajikan pada tabel 5.6.

Dari tabel 5.6 terlihat bahwa keseluruhan skenario yang ada pada aplikasi mulai dari awal membuka aplikasi hingga mengirim laporan menghasilkan keluaran yang diharapkan. Dengan demikian, dapat dipastikan bahwa fungsi-fungsi yang terdapat pada aplikasi juga dapat berjalan dengan semestinya.

Tabel 5.6: Hasil pengujian *Black Box*

No	Komponen Uji	Skenario Butir Uji	Hasil yang Diharapkan	Hasil Pengujian
1	Tampilan awal	Memilih <i>icon launcher</i> aplikasi	Ketika icon disentuh maka pengguna akan masuk ke tampilan awal/splash	Berhasil
2	Login	Mengisi formulir dan memilih tombol <i>login</i>	Ketika tombol disentuh dan data yang diisikan pengguna sesuai maka pengguna akan diarahkan ke menu utama	Berhasil
3	Register	Mengisi formulir dan memilih tombol <i>register</i>	Ketika tombol disentuh maka data pengguna baru akan tersimpan di basis data	Berhasil
4	Memilih tampilan daftar	Memilih menu daftar pada <i>toolbar</i>	Ketika menu daftar ditekan maka pengguna akan diarahkan ke tampilan daftar	Berhasil
5	Memilih tampilan peta	Memilih menu peta pada <i>toolbar</i>	Ketika menu peta ditekan maka pengguna akan diarahkan ke tampilan peta	Berhasil
6	Akses detail bangunan	Memilih salah satu item yang ada pada tampilan daftar	Ketika salah satu item ditekan maka pengguna akan diarahkan ke tampilan detail	Berhasil
7	Akses detail lokasi bangunan	Memilih detail lokasi bangunan	Ketika tombol detail lokasi ditekan maka pengguna akan diarahkan ke tampilan peta	Berhasil
8	Akses detail bangunan	Memilih salah satu marker yang ada pada tampilan peta	Ketika salah satu marker ditekan maka pengguna akan diarahkan ke tampilan detail	Berhasil
9	Laporkan bangunan baru	Menekan lama tampilan peta	Ketika peta ditekan lama maka akan muncul <i>marker</i> untuk melaporkan bangunan baru	Berhasil
10	Verifikasi bangunan	Menekan tombol Sesuai	Ketika tombol sesuai ditekan maka data verifikasi pada basis data akan diperbarui	Berhasil
11	Laporkan bangunan	Menekan tombol Tidak sesuai	Ketika tombol tidak sesuai ditekan maka pengguna akan diarahkan ke tampilan formulir laporan	Berhasil
12	Mengambil gambar	Menekan tombol Kamera	Ketika tombol kamera ditekan maka pengguna akan diarahkan untuk mengambil gambar	Berhasil
13	<i>Upload</i> laporan	Menekan tombol <i>Upload</i> Laporan	Ketika tombol <i>upload</i> laporan ditekan aplikasi akan mengirimkan dan menyimpan data laporan ke basis data	Berhasil

5.3 Pengujian API Berdasarkan Tipe Jaringan

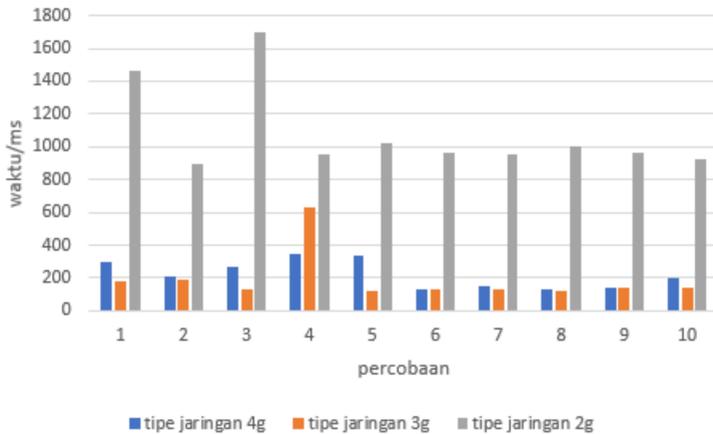
Pada bagian ini, akan dilakukan pengujian untuk memastikan apakah sistem yang dibuat berjalan dengan sesuai. Pengujian ini dilakukan terhadap *API* yang terdapat pada *web server*.

5.3.1 API *Login*

Pengujian ini dilakukan untuk memastikan tidak ada pengguna ilegal yang dapat mengakses fungsi yang disediakan aplikasi. Dalam pengujian ini dilakukan 10 kali percobaan pemanggilan *API* dengan menggunakan tiga buah tipe jaringan internet yang berbeda yaitu jaringan 4g, 3g, dan 2g. Hasil pengujian ini disajikan pada gambar 5.1.

Tabel 5.7: Hasil pengujian *API Login*

Percobaan	Jaringan		
	4g	3g	2g
1	296	183	1466
2	208	190	899
3	263	135	1702
4	350	631	957
5	340	117	1023
6	128	130	968
7	154	135	952
8	129	123	1000
9	142	138	961
10	201	139	929



Gambar 5.1: Hasil pengujian *API Login*

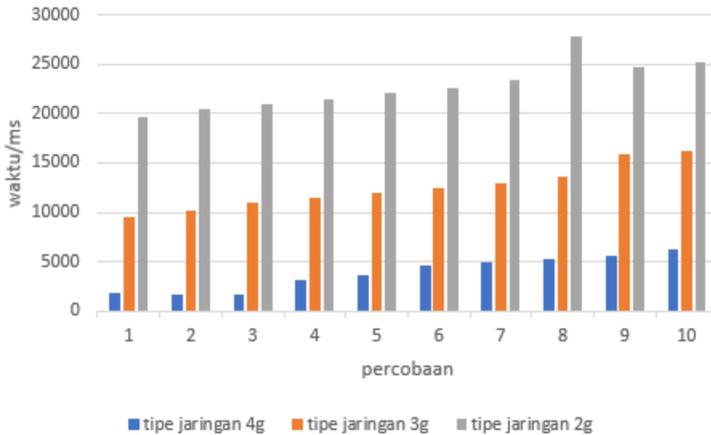
Pengujian login yang dilakukan menggunakan ketiga buah jaringan berjalan dengan baik. Percobaan ini memberikan keluaran berupa waktu yang dibutuhkan untuk menjalankan fungsi login dalam mili detik. Jaringan 4g menghasilkan rata-rata waktu proses selama 221,1 ms, jaringan 3g menghasilkan rata-rata waktu proses selama 192,1 ms dan jaringan 2g menghasilkan rata-rata waktu proses selama 1.085,5 ms. Lama waktu proses yang dilakukan pada jaringan 4g dan 3g relatif sama, sedangkan proses yang dilakukan menggunakan jaringan 2g membutuhkan waktu yang relatif lebih lama.

5.3.2 *API GetData*

Pengujian ini dilakukan untuk memastikan pengguna mendapatkan data bangunan yang sesuai ketika mengakses halaman menu utama pada aplikasi. Dalam pengujian ini dilakukan 10 kali percobaan pemanggilan *API* dengan menggunakan tiga buah tipe jaringan internet yang berbeda yaitu jaringan 4g, 3g, dan 2g. Dari percobaan tersebut didapatkan hasil yang disajikan pada gambar 5.2.

Tabel 5.8: Hasil pengujian *API GetData*

Percobaan	Jaringan		
	4g	3g	2g
1	1788	9605	19729
2	1750	10230	20429
3	1762	11028	20900
4	3148	11430	21420
5	3693	12017	22166
6	4609	12541	22583
7	4946	13013	23336
8	5242	13672	27890
9	5679	15942	24646
10	6254	16265	25152



Gambar 5.2: Hasil pengujian *API GetData*

Pengujian penerimaan data yang dilakukan menggunakan ketiga buah jaringan berjalan dengan baik. Pengujian ini memberikan keluaran berupa waktu yang dibutuhkan untuk menjalankan fungsi *getData* dalam mili detik. Dari 10 kali percobaan, jaringan 4g menghasilkan rata-rata waktu proses selama 3.887,1 ms, jaringan

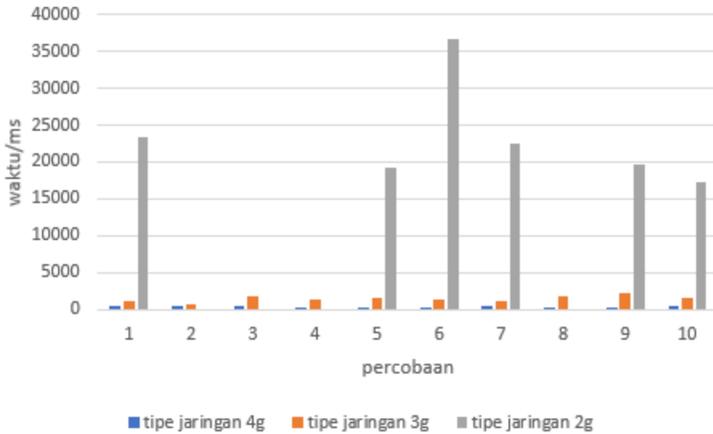
3g menghasilkan rata-rata waktu proses selama 12.574,3 ms, dan jaringan 2g menghasilkan rata-rata waktu proses selama 22.825,1 ms. Proses yang dilakukan dengan 4g menghasilkan proses tercepat, berikutnya adalah proses yang dilakukan dengan 3g dan proses terlambat adalah dengan jaringan 2g.

5.3.3 *API PostLaporan*

Pengujian ini dilakukan untuk memastikan *API PostLaporan* dapat berjalan sehingga pengguna berhasil menyimpan laporan hasil pemeriksaan ke dalam basis data yang berada di dalam server. Dalam pengujian ini dilakukan 10 kali percobaan pemanggilan *API* dengan menggunakan tiga buah tipe jaringan internet yang berbeda yaitu jaringan 4g, 3g, dan 2g. Data laporan yang disimpan berupa teks dan gambar. Dari percobaan tersebut didapatkan hasil yang disajikan pada gambar 5.3.

Tabel 5.9: Hasil pengujian *API PostLaporan*

Percobaan	Jaringan		
	4g	3g	2g
1	578	1111	23408
2	521	641	0
3	443	1739	0
4	388	1294	0
5	348	1552	19333
6	366	1328	36725
7	497	1240	22506
8	353	1894	0
9	321	2168	19634
10	417	1495	17206



Gambar 5.3: Hasil pengujian *API PostLaporan*

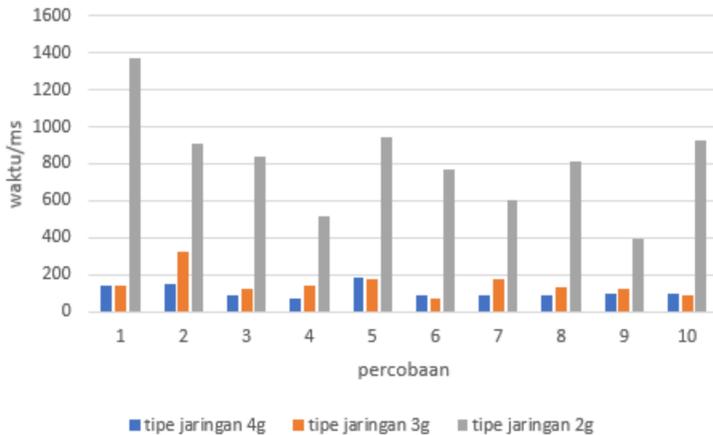
Pengujian ”*upload* data laporan hasil pemeriksaan” yang dilakukan menggunakan tiga jaringan yang berberda memberikan keluaran yang baik pada tipe jaringan 4g dan 3g, namun memberikan hasil yang kurang baik pada jaringan 2g. Dari 10 kali percobaan, jaringan 4g menghasilkan rata-rata waktu proses selama 423,2 ms, jaringan 3g menghasilkan rata-rata waktu proses selama 1.446,2 ms, sedangkan pada jaringan 2g dari 10 kali percobaan menghasilkan 6 percobaan berhasil dengan rata-rata waktu proses selama 138.812 ms dan 4 percobaan gagal. Proses upload data menggunakan jaringan 4g menghasilkan waktu tercepat, dibandingkan jaringan 3g dan 2g. Sedangkan pada jaringan 2g tingkat kesuksesan dari 10 kali percobaan hanya 60%.

5.3.4 *API PostVerifikasi*

Pengujian upload laporan verifikasi bangunan dilakukan untuk memastikan pengguna berhasil menyimpan laporan verifikasi bangunan ke dalam basis data. Dalam pengujian ini dilakukan 10 kali percobaan dengan menggunakan tiga buah tipe jaringan internet yang berbeda yaitu jaringan 4g, 3g, dan 2g. Dari percobaan tersebut didapatkan hasil yang disajikan pada gambar 5.4.

Tabel 5.10: Hasil pengujian *API PostVerifikasi*

Percobaan	Jaringan		
	4g	3g	2g
1	140	142	1370
2	154	327	909
3	92	124	841
4	75	142	518
5	184	174	943
6	88	75	765
7	90	177	607
8	89	130	813
9	95	127	398
10	101	90	926



Gambar 5.4: Hasil pengujian *API PostVerifikasi*

Pengujian upload laporan verifikasi bangunan yang dilakukan menggunakan tiga jaringan yang berbeda memberikan keluaran yang baik pada tipe jaringan 4g, 3g dan 2g. Dari 10 kali percobaan, jaringan 4g menghasilkan rata-rata waktu proses selama 110,8 ms, jaringan 3g menghasilkan rata-rata waktu proses selama 150,8 ms

dan jaringan 3g menghasilkan rata-rata waktu proses selama 809 ms. Ketiganya menghasilkan waktu proses yang tidak jauh berbeda dan relatif cepat.

5.4 Pengujian Sensor

Pada bagian ini, akan dilakukan pengujian untuk memastikan apakah sensor yang digunakan oleh perangkat berjalan dengan baik. Oleh karena itu dilakukan pengujian terhadap titik koordinat pengguna yang didapat oleh aplikasi dengan titik koordinat pengguna yang didapatkan oleh *Google Maps* dan menghitung jarak kedua titik tersebut menggunakan metode haversine untuk mengetahui seberapa besar kesalahan perolehan titik koordinat. Hasil pengujian sensor *GPS* dapat dilihat di tabel 5.11.

Tabel 5.11: Hasil pengujian akurasi sensor *GPS*

No	Lat 1	Lon 1	Lat 2	Lon 2	Jarak (meter)
1	-7,286715	112,799980	-7,286742	112,800030	0,2
2	-7,286724	112,799973	-7,286510	112,799937	1,4
3	-7,286667	112,799938	-7,286524	112,799957	0,9
4	-7,286630	112,799920	-7,286551	112,799985	0,5
5	-7,286619	112,799882	-7,286545	112,799926	0,5
6	-7,286630	112,799892	-7,286530	112,799943	0,7
7	-7,286744	112,800033	-7,286543	112,799954	1,3
8	-7,286759	112,800017	-7,286536	112,799989	1,4
9	-7,286060	112,799741	-7,286536	112,799983	3,1
10	-7,286770	112,799902	-7,286585	112,799987	1,2
Rata-rata jarak					1,1

Tabel 5.11 menunjukkan hasil pengambilan lokasi dari 10 kali percobaan pengambilan data menggunakan aplikasi dan menggunakan *Google Maps*. Lat 1 dan Lon 1 merupakan latitude dan longitude yang diperoleh dari aplikasi, sedangkan Lat 2 dan Lon 2 merupakan latitude dan longitude yang didapatkan dari Peta *Google Maps*. Jarak merupakan hasil perhitungan keempat parameter tersebut menggunakan rumus haversine. Dari pengujian yang dilakukan didapatkan rata-rata kesalahan yang didapatkan dari pengambilan lokasi pengguna adalah 1,1 meter.

Halaman ini sengaja dikosongkan

BAB 6

PENUTUP

6.1 Kesimpulan

Pada tugas akhir ini telah diterapkan serangkaian prosedur untuk membangun sebuah aplikasi mobile berbasis *android* sebagai alat bantu untuk menyediakan informasi bangunan serta menyediakan wadah untuk menampung laporan pemeriksaan IMB pada suatu kota.

Dari survey yang dilakukan kepada petugas, 70,6% petugas menyetujui bahwa aplikasi ini mampu memenuhi ketergunaan, 76,7% petugas menyetujui bahwa aplikasi ini mampu memenuhi unsur kebenaran, dan 75% petugas menyetujui bahwa aplikasi ini mampu memenuhi unsur keandalan. Sedangkan survey yang dilakukan kepada mahasiswa, 82,2% mahasiswa menyetujui bahwa aplikasi ini telah memenuhi unsur ketergunaan. Dari hasil survey tersebut maka dapat disimpulkan bahwa aplikasi ini bisa diterima oleh pengguna baik dari segi ketergunaan, kebenaran maupun keandalan.

Hasil pengujian Fungsional dan *API* menunjukkan bahwa aplikasi mampu menjalankan semua fungsinya sesuai dengan harapan apabila perangkat yang digunakan menggunakan jaringan 4g dan 3g. Namun, pada jaringan 2g ditemukan kesalahan pada proses pengiriman data. Kesalahan tersebut diakibatkan pada jaringan 2g aliran data tidak stabil sehingga perangkat tidak mendapatkan balasan dari server.

Pada pengujian sensor lokasi/*GPS* ditunjukkan bahwa rata-rata kesalahan koordinat yang didapat oleh aplikasi adalah 1,1 meter dari jarak aslinya. Dari data tersebut dapat disimpulkan bahwa lokasi yang didapatkan dari aplikasi masih dapat diandalkan.

6.2 Penelitian Selanjutnya

Pada penelitian selanjutnya akan dilakukan pengembangan tampilan antar muka yang lebih nyaman digunakan oleh pengguna, menambahkan metode pengambilan posisi yang lebih baik agar aplikasi dapat memperoleh data posisi yang lebih akurat, mengembangkan metode pengiriman data ke server agar data laporan dapat tersimpan ketika menggunakan segala tipe jaringan, serta menambahkan

fitur riwayat pemeriksaan agar petugas dapat mengetahui riwayat pemeriksaan suatu bangunan.

Dengan kompleksitas data yang dimiliki, penelitian ini dapat dikembangkan menjadi sebuah aplikasi pemeriksaan IMB yang dapat digunakan oleh masyarakat. Dengan demikian, masyarakat dapat ikut serta dalam pengawasan pembangunan yang terjadi pada lingkungannya sehingga proses pemeriksaan IMB dapat berjalan dengan cepat dan diharapkan dapat memberikan dampak pada pembangunan daerah yang tertib.

Pengembangan lain yang mungkin dilakukan dengan memanfaatkan kompleksitas data adalah dengan memanfaatkan data-data tersebut untuk menentukan secara otomatis biaya retribusi yang ditanggungkan kepada pemilik bangunan, sehingga selain proses pemeriksaan IMB, keluaran dari pemeriksaan juga bisa didapatkan dengan cepat.

DAFTAR PUSTAKA

- [1] M. P. Siahaan, Hukum Pajak Material. 2008. (Dikutip pada halaman 7).
- [2] M. P. Siahaan, Pajak dan Retribusi Daerah. 2005. (Dikutip pada halaman 7).
- [3] “Dokumentasi imb kota surabaya.” http://ssw.surabaya.go.id/index.php?hal=subijin&kd_ijin=010100&kd_paket=22&kd_skpd=&kd_ijin_ext. Diakses pada 21 November 2016, pukul 12:00. (Dikutip pada halaman 7, 8).
- [4] R. Indonesia, Undang-undang No.28 tahun 2002 tentang Bangunan Gedung. 2002. (Dikutip pada halaman 7).
- [5] R. Indonesia, Undang-undang No.26 tahun 2007 tentang Penataan Ruang. 2007. (Dikutip pada halaman 8).
- [6] d. Cecep Nurul Alam, “Implementation of haversine formula for counting event visitor in the radius based on android application,” IEEE Conference, 2016. (Dikutip pada halaman 9).
- [7] I. C. A. Organization, World Geodetic System 1984 (WGS-84) Manual. 2002. (Dikutip pada halaman 10).

Halaman ini sengaja dikosongkan

BIOGRAFI PENULIS



Nurfajariyawan Juniantera, lahir pada 30 Juni 1995 di Kediri, Jawa Timur. Penulis lulus dari MTsN 2 Kediri pada tahun 2010 kemudian melanjutkan pendidikan ke SMA Negeri 2 Kediri hingga akhirnya lulus pada tahun 2013. Penulis kemudian melanjutkan pendidikan Strata satu ke Departemen Teknik Komputer ITS Surabaya bidang studi Telematika. Saat di kuliah penulis aktif menjadi Asisten laboratorium B201 (Telematika) hingga saat ini dan pernah menjabat sebagai koordinator asisten Lab B201 periode 2013/2014. Selama masa kuliah penulis aktif dalam mengikuti ajang perlombaan seperti *Hackaton* dan Karya Tulis Ilmiah. Penulis sangat tertarik dengan segala hal yang berhubungan dengan komputer, dan berencana mendalami cabang ilmu komputer lain selain telematika.

Halaman ini sengaja dikosongkan