

**PENGEMBANGAN MODUL INFORMASI OBJEK LAYANAN
KESEHATAN PADA APLIKASI MALANG SEHAT**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Muh Wildan Shalahuddin

NIM: 165150200111118



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA**

FAKULTAS ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

MALANG

2021



PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 17 Juni 2021



Muh Wildan Salahuddin

NIM: 165150200111118

PENGESAHAN

PENGEMBANGAN MODUL INFORMASI OBJEK LAYANAN KESEHATAN PADA
APLIKASI MALANG SEHAT

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer


Disusun Oleh :
Muh Wildan Shalahuddin
NIM: 165150200111118

Skripsi ini telah diuji dan dinyatakan lulus pada
5 Juli 2021

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I


Dosen Pembimbing II


Herman Tolle, Dr. Eng., S.T, M.T.
NIP: 19740823 200012 1 001


Ratih Kartika Dewi, S.T., M.Kom
NIK: 201503 890520 2 001

Mengetahui
Ketua Jurusan Teknik Informatika




Achmad Basuki, S.T., M.MG., Ph.D.
NIP: 19741118 200312 1 002

ABSTRAK

Muh Wildan Shalahuddin, Pengembangan Modul Informasi Objek Layanan Kesehatan pada Aplikasi Malang Sehat.

Pembimbing: Dr. Eng. Herman Tolle, S.T., M.T., Ratih Kartika Dewi, S.T., M.Kom

Masyarakat Indonesia memiliki hak untuk memperoleh informasi tentang kesehatan, hal ini tertulis dalam Undang-undang nomor 36 tahun 2009 bab III pasal 7. Sedangkan pemerintah memiliki kewajiban untuk memberikan ketersediaan akses terhadap informasi, edukasi, dan fasilitas pelayanan kesehatan yang tercantum dalam pasal 14 bab 4 Undang-undang nomor 36 tahun 2009. Namun, berdasarkan fakta di lapangan ketika dilakukan wawancara dengan beberapa masyarakat kota Malang, dapat ditarik kesimpulan bahwa sistem yang ada saat ini masih konvensional dan memerlukan perbaikan dalam proses penyampaian informasinya. Pemerintah kota Malang berinovasi dengan menerapkan konsep *Smart City* guna menyelesaikan permasalahan ini. Realisasi yang dilakukan adalah dengan dibangunnya aplikasi perangkat bergerak berbasis Android bernama Malang Sehat yang dipecah menjadi 3 modul, yaitu modul Pendataan dan Monitoring Kesehatan, modul Pelayanan Kesehatan dan yang terakhir sekaligus menjadi topik pada penelitian ini adalah modul Informasi Objek Layanan Kesehatan. Modul ini dibangun menggunakan metode *Prototyping* dengan 5 tahapan proses, yaitu *communication, quick plan, modelling quick design, constuction of prototype*, dan *delivery prototype and feedback*. Iterasi perulangan akan berhenti apabila pemangku kepentingan menyatakan aplikasi telah memenuhi kebutuhan pada saat memberikan *feedback*. Setelah proses iterasi selesai dan *prototype* sudah dinyatakan memenuhi kebutuhan, maka dilakukan pengujian Fungsional, *Usability*, dan *Compatibility*. Pada pengujian Fungsional menghasilkan tingkat keberhasilan 100%. Selanjutnya pengujian *Usability* yang diujikan kepada 10 responden menghasilkan *success rate* sebesar 98,57% dan skor akhir *System Usability Scale (SUS)* mencapai 84,75% dengan *adjectiva rating "Good"*. Terakhir pengujian *Compatibility* menghasilkan hanya kesimpulan bahwa sistem dapat berjalan dengan baik pada versi minimal android 5 dengan tingkatan API 21.

Kata kunci: kesehatan, aplikasi perangkat bergerak, android, *smart city*, *prototyping*

ABSTRACT

Muh Wildan Shalahuddin, Development of the Health Service Object Information Module on the Malang Sehat Application.

Supervisors: Dr. Eng. Herman Tolle, S.T., M.T., Ratih Kartika Dewi, S.T., M.Kom

Indonesian people have the right to obtain information about health, this is written in Undang-undang number 36 of 2009 chapter III article 7. Meanwhile, the government has an obligation to provide access to information, education, and health service facilities as stated in article 14 chapter 4 Undang-undang number 36 of 2009. However, based on the facts on the ground when interviews were conducted with several people in the city of Malang, it can be concluded that the current system is still conventional and requires improvements in the process of delivering information. The Malang city government innovates by applying the Smart City concept to solve this problem. The realization that was carried out was the development of an Android-based mobile application called Malang Sehat which was divided into 3 modules, namely the Health Data Collection and Monitoring module, the Health Services module and the last one that became the topic of this research was the Health Service Object Information module. This module was built using the Prototyping method with 5 process stages, namely communication, quick plan, quick design modeling, construction of prototype, and prototype delivery and feedback. The loop iteration will stop if the stakeholders state that the application has met the needs when providing feedback. After the iteration process is complete and the prototype has been declared to meet the needs, then the Functional, Usability, and Compatibility testing is carried out. On Functional testing produces a 100% success rate. Furthermore, the Usability test which was tested on 10 respondents resulted in a success rate of 98.57% and the final score of the System Usability Scale (SUS) reaching 84.75% with an adjective rating of "Good". Finally, the Compatibility test only concluded that the system can run well on a minimum version of Android 5 with API level 21.

Keyword: health, mobile apps, android, smart city, prototyping

DAFTAR ISI

PERNYATAAN ORISINALITAS	ii
PENGESAHAN	iii
PRAKATA	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL	xi
DAFTAR GAMBAR	xv
DAFTAR LAMPIRAN	xviii
DAFTAR PERSAMAAN	xix
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	3
1.4 Manfaat	3
1.5 Batasan Masalah	3
1.6 Sistematika Pembahasan	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Dasar Teori	5
2.2.1 Fasilitas Layanan Kesehatan	5
2.2.2 <i>Smart City</i>	6
2.2.3 Android	8
2.2.4 Basis Data MySQL	8
2.2.5 <i>Application Programming Interface (API)</i>	9
2.2.6 Google Maps	9
2.2.7 <i>Prototyping</i>	10
2.2.8 <i>Location Based Services (LBS)</i>	11
2.2.9 Teori Pengujian	11
BAB 3 METODE PENELITIAN	14



3.1 Studi Literatur	14
3.2 <i>Prototyping</i>	15
3.2.1 <i>Communication</i> (Komunikasi)	15
3.2.2 <i>Quick Plan</i> (Rencana Singkat)	15
3.2.3 <i>Modelling Quick Design</i> (Pemodelan Desain Cepat).....	15
3.2.4 <i>Construction of Prototype</i> (Pembangunan Prototipe).....	16
3.2.5 <i>Deployment Delivery and Feedback</i> (Penyebaran dan Umpan balik)	16
3.3 Pengujian dan Analisis	16
3.4 Kesimpulan dan Saran	16
BAB 4 ANALISIS KEBUTUHAN	17
4.1 Gambar Umum Aplikasi	17
4.2 Komunikasi.....	17
4.2.1 Iterasi 0.....	17
4.2.2 Iterasi 1.....	18
4.3 Kebutuhan Fungsional	18
4.3.1 Iterasi 0.....	18
4.3.2 Iterasi 1.....	19
4.4 Kebutuhan Non-Fungsional	19
4.4.1 Iterasi 0.....	19
4.5 <i>Use Case Diagram</i>	20
4.5.1 Iterasi 0.....	20
4.5.2 Iterasi 1.....	21
4.6 <i>Use Case Skenario</i>	21
4.6.1 Iterasi 0.....	21
4.6.2 Iterasi 1.....	23
BAB 5 PERANCANGAN.....	28
5.1 Perancangan Arsitektur Sistem.....	28
5.2 Perancangan <i>Sequence Diagram</i>	29
5.2.1 Iterasi 0.....	29
5.2.2 Iterasi 1.....	32
5.3 Perancangan <i>Class Diagram</i>	39

5.3.1 Iterasi 0.....	39
5.3.2 Iterasi 1.....	40
5.4 Perancangan Tabel Basis Data.....	41
5.4.1 Iterasi 0.....	41
5.4.2 Iterasi 1.....	43
5.5 Perancangan Antarmuka.....	44
5.5.1 Iterasi 0.....	45
5.5.2 Iterasi 1.....	61
BAB 6 IMPLEMENTASI.....	65
6.1 Spesifikasi Sistem.....	65
6.2 Batasan-batasan Implementasi.....	65
6.3 Implementasi Tabel Basis Data.....	66
6.3.1 Iterasi 0.....	66
6.3.2 Iterasi 1.....	70
6.4 Implementasi Kode Program.....	71
6.4.1 Iterasi 0.....	71
6.4.2 Iterasi 1.....	84
6.5 Implementasi Antarmuka.....	90
6.5.1 Iterasi 0.....	90
6.5.2 Iterasi 1.....	102
6.6 Umpan Balik Pemangku Kepentingan.....	107
6.6.1 Iterasi 0.....	107
6.6.2 Iterasi 1.....	107
BAB 7 PENGUJIAN DAN ANALISIS.....	108
7.1 Pengujian Fungsional (<i>Blackbox</i>).....	108
7.1.1 Analisis Hasil Pengujian Fungsional.....	115
7.2 Pengujian <i>Compatibility</i>	115
Masuk Akun.....	115
7.2.1 Analisis Pengujian <i>Compatibility</i>	116
7.3 Pengujian Usability.....	116
7.3.1 Analisis Pengujian Usability.....	119
BAB 8 PENUTUP.....	120



8.1 Kesimpulan..... 120

8.2 Saran 121

DAFTAR REFERENSI 122

LAMPIRAN A HASIL WAWANCARA..... 124

A.1 Wawancara Kelompok Pengguna..... 124

A.2 Wawancara Komunikasi Pemangku Kepentingan iterasi 0 133

A.3 Wawancara Umpan Balik Pemangku Kepentingan Iterasi 0 134

A.4 Wawancara Komunikasi Pemangku Kepentingan iterasi 1 134

A.5 Wawancara Umpan Balik Pemangku Kepentingan Iterasi 1 135

LAMPIRAN B HASIL KUESIONER SUS 136



DAFTAR TABEL

Tabel 2.1 <i>Adjectiva Rating SUS</i>	12
Tabel 4.1 Kebutuhan Fungsional Iterasi 0	18
Tabel 4.2 Kebutuhan Fungsional Iterasi 1	19
Tabel 4.3 Kebutuhan Non-Fungsional Iterasi 0	19
Tabel 4.4 <i>Use Case</i> Skenario Lihat Informasi Objek Layanan Kesehatan.....	21
Tabel 4.5 <i>Use Case</i> Skenario Lihat Informasi Dokter	22
Tabel 4.6 <i>Use Case</i> Skenario Lihat Denah Objek Layanan Kesehatan	22
Tabel 4.7 <i>Use Case</i> Skenario <i>Penanda</i>	23
Tabel 4.8 <i>Use Case</i> Skenario Masuk Akun	24
Tabel 4.9 <i>Use Case</i> Skenario Daftar Akun	24
Tabel 4.10 <i>Use Case</i> Skenario Keluar Akun.....	25
Tabel 4.11 <i>Use Case</i> Skenario Lihat Informasi Objek Layanan Kesehatan.....	25
Tabel 4.12 <i>Use Case</i> Skenario Lihat Informasi Dokter	26
Tabel 4.13 <i>Use Case</i> Skenario Lihat Denah Objek Layanan Kesehatan	26
Tabel 4.14 <i>Use Case</i> Skenario <i>Penanda</i>	27
Tabel 5.1 Rancangan Tabel Dokter	41
Tabel 5.2 Rancangan Tabel Objek_Layanan_Kesehatan.....	41
Tabel 5.3 Rancangan Tabel Spesialis.....	42
Tabel 5.4 Rancangan Tabel Kategori.....	42
Tabel 5.5 Rancangan Tabel Fasilitas.....	42
Tabel 5.6 Rancangan Tabel Dukungan_Pembayaran.....	42
Tabel 5.7 Rancangan Tabel Jenis.....	42
Tabel 5.8 Rancangan Tabel Praktik_Dokter	42
Tabel 5.9 Rancangan Tabel Dukungan Pembayaran_Objek_Layanan_Kesehatan.....	43
Tabel 5.10 Rancangan Tabel Jenis_Objek_Layanan_Kesehatan.....	43
Tabel 5.11 Rancangan Tabel Akun	44
Tabel 5.12 Rancangan Tabel_Penanda_Dokter	44
Tabel 5.13 Rancangan Tabel Penanda_Objek_Layanan_Kesehatan	44
Tabel 5.14 Informasi Element Rancangan Antarmuka Halaman <i>Splash Screen</i> ...	45
Tabel 5.15 Informasi Element Rancangan Antarmuka Halaman Beranda.....	46

Tabel 5.16 Informasi Element Rancangan Antarmuka Halaman Daftar Spesialis	47
Tabel 5.17 Informasi Element Rancangan Antarmuka Halaman Daftar Dokter ...	48
Tabel 5.18 Informasi Element Rancangan Antarmuka Saring Daftar Dokter	49
Tabel 5.19 Informasi Element Rancangan Antarmuka Halaman Detail Dokter...	50
Tabel 5.20 Informasi Element Rancangan Antarmuka Halaman Detail Dokter Tab Informasi Umum	51
Tabel 5.21 Informasi Element Rancangan Antarmuka Halaman Detail Dokter Bagian Tab Praktik.....	52
Tabel 5.22 Informasi Element Rancangan Antarmuka Halaman Daftar Objek Layanan Kesehatan	53
Tabel 5.23 Informasi Element Rancangan Antarmuka <i>Pop-up</i> Saring Daftar Objek Layanan Kesehatan	54
Tabel 5.24 Informasi Element Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan	55
Tabel 5.25 Informasi element Rancangan antarmuka halaman Detail Objek Layanan Kesehatan Kolom Informasi Umum.....	57
Tabel 5.26 Informasi Element Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Dokter Tersedia.....	58
Tabel 5.27 Informasi Element Rancangan Antarmuka Halaman Denah.....	59
Tabel 5.28 Informasi Element Rancangan Antarmuka <i>Pop-up</i> Pilih Kategori Halaman Denah.....	60
Tabel 5.29 Informasi Element Rancangan Antarmuka Halaman Masuk	61
Tabel 5.30 Informasi Element Rancangan Antarmuka Halaman Daftar	62
Tabel 5.31 Informasi Element Rancangan Antarmuka Halaman Beranda.....	63
Tabel 5.32 Informasi Element Rancangan Antarmuka Halaman Penanda	64
Tabel 6.1 Spesifikasi Perangkat Keras	65
Tabel 6.2 Spesifikasi Perangkat Lunak	65
Tabel 6.3 Kode Program Fungsi <code>getListLakes ()</code>	71
Tabel 6.4 Penjelasan Kode Program Fungsi <code>getListLakes ()</code>	72
Tabel 6.5 Kode Program Fungsi <code>getDetailLakes ()</code>	72
Tabel 6.6 Penjelasan Kode Program Fungsi <code>getDetailLakes ()</code>	73
Tabel 6.7 Kode Program Fungsi <code>getInfoUmumLakes ()</code>	73
Tabel 6.8 Penjelasan Kode Program Fungsi <code>getInfoUmumLakes ()</code>	74
Tabel 6.9 Kode Program Fungsi <code>getDokterLakes ()</code>	75

Tabel 6.10 Penjelasan Kode Program Fungsi getDokterLakes ()	76
Tabel 6.11 Kode Program Fungsi getListSpesialis ()	77
Tabel 6.12 Penjelasan Kode Program Fungsi getListSpesialis ()	77
Tabel 6.13 Kode Program Fungsi getListDokter ()	78
Tabel 6.14 Penjelasan Kode Program Fungsi getListDokter ()	78
Tabel 6.15 Kode Program Fungsi getDetailDokter ()	79
Tabel 6.16 Penjelasan Kode Program Fungsi getDetailDokter ()	79
Tabel 6.17 Kode Program Fungsi getJadwalDokter ()	80
Tabel 6.18 Penjelasan Kode Program Fungsi getJadwalDokter ()	80
Tabel 6.19 Kode Program Fungsi getInfoUmumDokter ()	81
Tabel 6.20 Penjelasan Kode Program Fungsi getInfoUmumDokter ()	81
Tabel 6.21 Kode Program Fungsi getKategori ()	82
Tabel 6.22 Penjelasan Kode Program Fungsi getKategori ()	82
Tabel 6.23 Kode Program Fungsi getListLayananKesehatan ()	83
Tabel 6.24 Penjelasan Kode Program Fungsi getListLayananKesehatan ()	84
Tabel 6.25 Kode Program Fungsi masukAkun ()	84
Tabel 6.26 Penjelasan Kode Program Fungsi masukAkun ()	85
Tabel 6.27 Kode Program Fungsi daftarAkun ()	85
Tabel 6.28 Penjelasan Kode Program Fungsi daftarAkun ()	86
Tabel 6.29 Kode Program Fungsi keluarAkun ()	87
Tabel 6.30 Penjelasan Kode Program Fungsi keluarAkun ()	87
Tabel 6.31 Kode Program Fungsi getListDokterDitandai ()	87
Tabel 6.32 Penjelasan Kode Program Fungsi getListDokterDitandai ()	88
Tabel 6.33 Kode Program Fungsi getListLakesDitandai ()	89
Tabel 6.34 Penjelasan Kode Program Fungsi getListLakesDitandai ()	89
Tabel 6.34 Penjelasan Kode Program Fungsi getListLakesDitandai () (lanjutan)	90
Tabel 7.1 Pengujian Fungsional Lihat Informasi Objek Layanan Kesehatan	108
Tabel 7.2 Pengujian Fungsional Lihat Informasi Dokter	109
Tabel 7.3 Pengujian Fungsional Lihat Denah Objek Layanan Kesehatan	110
Tabel 7.4 Pengujian Fungsional Masuk Akun	112

Tabel 7.5 Pengujian Fungsional Daftar Akun	112
Tabel 7.6 Pengujian Fungsional Keluar Akun	113
Tabel 7.7 Pengujian Fungsional Penanda	114
Tabel 7.8 Hasil Pengujian Compatibility Secara Manual	115
Tabel 7.9 Tugas Uji <i>Usability</i>	116
Tabel 7.10 Hasil Pengerjaan Tugas Oleh Responden	117
Tabel 7.11 Daftar Pernyataan Kuesioner SUS	117
Tabel 7.12 Hasil pengisian kuesioner SUS	118
Tabel 7.13 Perhitungan akumulasi skor SUS	119



DAFTAR GAMBAR

Gambar 2.1 Dimensi <i>Smart city</i>	6
Gambar 2.2 <i>The Prototyping paradigm</i>	10
Gambar 3.1 Diagram Alir Pengerjaan Skripsi	14
Gambar 4.1 <i>Use Case Diagram</i> Iterasi 0.....	20
Gambar 4.2 <i>Use Case Diagram</i> Iterasi 1.....	21
Gambar 5.1 Skema Arsitektur Aplikasi Malang Sehat	28
Gambar 5.2 Sequence Diagram Lihat Informasi Objek Layanan Kesehatan.....	29
Gambar 5.3 Sequence Diagram Lihat Informasi Dokter	30
Gambar 5.4 Sequence Diagram Lihat Denah Objek Layanan Kesehatan	31
Gambar 5.5 <i>Sequence Diagram</i> Masuk Akun.....	32
Gambar 5.6 <i>Sequence Diagram</i> Daftar Akun	33
Gambar 5.7 <i>Sequence Diagram</i> Keluar Akun.....	34
Gambar 5.8 <i>Sequence Diagram</i> Penanda	35
Gambar 5.9 <i>Sequence Diagram</i> Lihat Informasi Objek Layanan Kesehatan.....	36
Gambar 5.10 <i>Sequence Diagram</i> Lihat Informasi Dokter	37
Gambar 5.11 <i>Sequence Diagram</i> Lihat Denah Objek Layanan Kesehatan	38
Gambar 5.12 Perancangan <i>Class Diagram</i> Iterasi 0	39
Gambar 5.13 Perancangan <i>Class Diagram</i> Iterasi 1	40
Gambar 5.14 Rancangan Antarmuka Halaman <i>Splash Screen</i>	45
Gambar 5.15 Rancangan Antarmuka Halaman Beranda	46
Gambar 5.16 Rancangan Antarmuka Halaman Daftar Spesialis.....	47
Gambar 5.17 Rancangan Antarmuka Halaman.....	48
Gambar 5.18 Rancangan Antarmuka <i>Pop-up</i> Saring Daftar Dokter	49
Gambar 5.19 Rancangan Antarmuka Halaman Detail Dokter	50
Gambar 5.20 Rancangan Antarmuka Halaman Detail Dokter Tab Informasi Umum	51
Gambar 5.21 Rancangan Antarmuka Halaman Detail Dokter Tab Praktik	52
Gambar 5.22 Rancangan Antarmuka Halaman Daftar Objek Layanan Kesehatan.....	53
Gambar 5.23 Rancangan Antarmuka <i>Pop-up</i> Saring Objek Layanan Kesehatan ..	54
Gambar 5.24 Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan	55

Gambar 5.25 Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Informasi Umum	56
Gambar 5.26 Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Dokter Tersedia	57
Gambar 5.27 Rancangan Antarmuka Halaman Denah Objek Layanan Kesehatan	58
Gambar 5.28 Rancangan Antarmuka <i>Pop-up</i> Pilih Kategori Halaman Denah	60
Gambar 5.29 Rancangan Antarmuka Halaman Masuk	61
Gambar 5.30 Rancangan Antarmuka Halaman Daftar	62
Gambar 5.31 Rancangan Antarmuka Halaman Beranda	63
Gambar 5.32 Rancangan Antarmuka Halaman Penanda	64
Gambar 6.1 Implementasi Tabel Dokter	66
Gambar 6.2 Implementasi Tabel Dukungan_pembayaran	67
Gambar 6.3 Implementasi Tabel Fasilitas	67
Gambar 6.4 Implementasi Tabel Fasilitas_lakes	68
Gambar 6.5 Implementasi Tabel Jenis_rumah_sakit	68
Gambar 6.6 Implementasi Tabel Kategori	69
Gambar 6.7 Implementasi Tabel Lakes	69
Gambar 6.8 Implementasi Tabel Akun	70
Gambar 6.9 Implementasi Tabel Penanda_dokter	70
Gambar 6.10 Implementasi Tabel Penanda_lakes	70
Gambar 6.11 Antarmuka Halaman <i>Splash Screen</i>	91
Gambar 6.12 Antarmuka halaman Beranda	91
Gambar 6.13 Implementasi Antarmuka Halaman Daftar Spesialis	92
Gambar 6.14 Implementasi Antarmuka Halaman Daftar Dokter	93
Gambar 6.15 Antarmuka <i>Pop-up</i> Saring Daftar Dokter	93
Gambar 6.16 Antarmuka Halaman Detail Dokter	94
Gambar 6.17 Antarmuka Halaman Detail Dokter Tab Informasi Umum	95
Gambar 6.18 Antarmuka Halaman Detail Dokter Tab Praktik	95
Gambar 6.19 Antarmuka <i>Pop-up</i> Pilih Kecamatan	96
Gambar 6.20 Antarmuka Halaman Daftar Objek Layanan Kesehatan	97
Gambar 6.21 Antarmuka <i>Pop-up</i> Saring Daftar Objek Layanan Kesehatan	98
Gambar 6.22 Antarmuka Halaman Detail Objek Layanan Kesehatan	98

Gambar 6.23 Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Informasi Umum..... 99

Gambar 6.24 Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Dokter Tersedia..... 100

Gambar 6.25 Antarmuka *Pop-up* Pilih Kategori Halaman Denah Objek Layanan Kesehatan..... 101

Gambar 6.26 Antarmuka Halaman Denah Objek Layanan Kesehatan 101

Gambar 6.27 Antarmuka Halaman Navigasi Aplikasi Google Maps 102

Gambar 6.29 Implementasi Antarmuka Halaman Beranda Sudah Masuk Akun 103

Gambar 6.28 Implementasi Antarmuka Halaman Beranda Belum Masuk Akun 103

Gambar 6.31 Antarmuka Halaman Detail Dokter Tidak Ditantai 103

Gambar 6.30 Antarmuka Halaman Detail Dokter Ditantai 103

Gambar 6.32 Antarmuka Halaman Detail Objek Layanan Kesehatan Tidak Ditantai 104

Gambar 6.33 Antarmuka Halaman Detail Objek Layanan Kesehatan Ditantai ..104

Gambar 6.34 Implementasi Antarmuka Halaman Masuk Akun 105

Gambar 6.35 Implementasi Antarmuka Halaman Daftar Akun..... 106

Gambar 6.36 Antarmuka Halaman Penanda 106



DAFTAR LAMPIRAN

LAMPIRAN A HASIL WAWANCARA.....	124
A.1 Wawancara Kelompok Pengguna.....	124
A.2 Wawancara Komunikasi Pemangku Kepentingan iterasi 0.....	133
A.3 Wawancara Umpan Balik Pemangku Kepentingan Iterasi 0.....	134
A.4 Wawancara Komunikasi Pemangku Kepentingan iterasi 1.....	134
A.5 Wawancara Umpan Balik Pemangku Kepentingan Iterasi 1.....	135
LAMPIRAN B HASIL KUESIONER SUS.....	136





DAFTAR PERSAMAAN

Persamaan 7.1

117



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Kesehatan berasal dari kata sehat, menurut Kamus Besar Bahasa Indonesia (KBBI) sehat dapat diartikan sebagai kondisi terbebas dari penyakit. Kemudian berdasarkan pasal 1 Undang-undang nomor 36 tahun 2009 menyatakan bahwa, “kesehatan adalah keadaan sehat, baik secara fisik, mental, spritual maupun sosial yang memungkinkan setiap orang untuk hidup produktif secara sosial dan ekonomis”. Rakyat Indonesia memiliki hak untuk mendapatkan dukungan kesehatan. Salah satu hak tersebut adalah memperoleh informasi tentang kesehatan, hal ini tertulis dalam Undang-undang nomor 36 tahun 2009 bab III pasal 7 yaitu, “Setiap orang berhak untuk mendapatkan informasi dan edukasi tentang kesehatan yang seimbang dan bertanggung jawab”. Pemerintah pun memiliki kewajiban yang harus dijalankan menurut pasal 14 bab 4 Undang-undang nomor 36 tahun 2009 yaitu, “Pemerintah bertanggung jawab atas ketersediaan akses terhadap informasi, edukasi, dan fasilitas pelayanan kesehatan untuk meningkatkan dan memelihara derajat kesehatan yang setinggi-tingginya”.

Namun, berdasarkan fakta di lapangan ketika dilakukan wawancara dengan beberapa masyarakat kota Malang, terdapat beberapa point masalah, yaitu proses pencarian informasi fasilitas layanan kesehatan dan dokter masih dilakukan secara mulut ke mulut saja, masyarakat datang secara langsung ke objek layanan kesehatan terdekat tanpa mencari tahu dulu informasi tentang ketersediaan fasilitas atau dokter spesialis yang ada pada objek layanan kesehatan tersebut, informasi yang ada sulit dipahami dengan cepat dikarenakan tampilan yang hanya berupa tulisan saja, dan terakhir ketidaksesuaian informasi objek layanan kesehatan yang ada pada internet dengan fakta di lapangan.

Pemerintah kota Malang terus berupaya melakukan inovasi guna menyelesaikan permasalahan tersebut. Salah satu upaya yang dilakukan adalah dengan menerapkan *Smart City*. *Smart City* Menurut Meijer & Bolívar (2015) adalah “kemampuan sebuah kota untuk menarik modal manusia dan memobilisasi sumber daya manusia ini dalam kolaborasi antara berbagai aktor (terorganisir dan individu) melalui penggunaan teknologi informasi dan komunikasi”. Dengan terealisasinya *Smart City* maka beberapa aspek kehidupan akan ditunjang dengan bantuan teknologi informasi, khususnya di bidang kesehatan.

Kemudian, langkah selanjutnya untuk merealisasikan *Smart City* pada aspek kesehatan ini adalah menentukan media yang dapat menjangkau masyarakat. Untuk hasil yang lebih baik, tentunya media yang digunakan harus mengikuti perkembangan teknologi yang ada, seperti pemanfaatan telepon seluler (ponsel) dan internet. namun seiring perkembangannya ponsel berkembang menjadi sebuah ponsel pintar, yang secara umum dapat diartikan sebagai ponsel yang memiliki berbagai macam fungsionalitas, termasuk dapat terhubung dengan jaringan internet. Internet memiliki keuntungan besar dalam aspek efisiensi

penyebaran informasi. Melalui internet, informasi cukup disebarakan sekali saja dan masyarakat dapat mengaksesnya secara serempak di manapun ia berada asalkan dapat terhubung dalam jaringan. Sehingga memilih media ponsel pintar yang terhubung dengan internet untuk penyebaran informasi merupakan pilihan yang tepat di era ini.

Pada saat ini, ponsel pintar dengan tingkat pengguna terbanyak adalah yang memiliki sistem operasi Android. Android sendiri telah menguasai lebih dari 90 persen pasar ponsel pintar di Indonesia dan 75 persen di dunia (Permana, 2019). Android menyediakan beberapa aplikasi bawaan untuk menunjang fitur dasar pada sistem operasinya. Salah satu aplikasi tersebut adalah sebuah peta bernama Google Maps. Google Maps memberikan layanan berbasis lokasi seperti mencari alamat suatu tempat, melihat titik lokasi pengguna, bahkan Google Maps juga dapat memberikan arahan penunjuk jalan pada pengguna yang ingin menuju ke suatu lokasi. Selain menyediakan aplikasi, Google Maps juga menyediakan *Application Programming Interface (API)* yang ditujukan kepada para pengembang agar dapat menyematkan fitur-fitur layanan lokasi dari Google Maps kedalam aplikasi / sistem buatanya. Selain memberikan dukungan *API*, Google Maps juga memberikan dukungan framework terintegrasi dengan IDE Android Studio. Dukungan ini memudahkan pengembang dalam menulis kode program ke dalam aplikasi yang menyematkan fitur Google Maps.

Berdasarkan uraian latar belakang tersebut, dibangunlah aplikasi Android bernama "Malang Sehat". Aplikasi ini terbagi menjadi 3 modul, yaitu modul Pendataan dan Monitoring Kesehatan, modul Pelayanan Kesehatan dan yang terakhir sekaligus menjadi topik pada penelitian ini adalah modul Informasi Objek Layanan Kesehatan. Informasi yang diberikan oleh modul Informasi Objek Layanan Kesehatan nantinya akan berkaitan dengan data dokter spesialis, data objek layanan kesehatan, denah lokasi penyedia layanan kesehatan yang diintegrasikan dengan aplikasi Google Maps untuk memberikan arahan rute perjalanan pada pengguna yang ingin menuju tempat penyedia layanan kesehatan berada. Dalam urusan penyimpanan data, modul aplikasi ini nantinya akan menggunakan MySQL sebagai basis datanya.

Aplikasi yang telah dibangun nantinya akan diuji secara fungsionalitas, *Usability*, dan *Compatibility* untuk mengetahui kualitasnya. Pengujian fungsionalitas berfokus dalam menguji fungsi-fungsi yang ditawarkan dalam aplikasi apakah dapat berjalan dengan baik atau tidak. Kemudian pengujian *Usability* berfokus pada umpan balik partisipan tentang kemudahan penggunaan aplikasi. Terakhir pengujian *Compatibility* berfokus pada kemampuan aplikasi berjalan pada lingkungan, sistem operasi, dan perangkat keras yang berbeda-beda.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan di atas, maka dapat diambil rumusan masalah sebagai berikut:

1. Apa saja kebutuhan fungsional dan non-fungsional aplikasi Malang Sehat?

2. Bagaimana hasil rancangan aplikasi Malang sehat?
3. Bagaimana hasil implementasi aplikasi Malang sehat?
4. Bagaimana hasil pengujian Fungsional, *Usability* dan *Compatibility* pada aplikasi Malang Sehat?

1.3 Tujuan

Tujuan dari penelitian ini adalah:

1. Mengetahui kebutuhan fungsional dan non-fungsional dalam rancangan aplikasi Malang Sehat.
2. Mengetahui hasil dari rancangan aplikasi Malang Sehat.
3. Mengetahui hasil dari implementasi aplikasi Malang Sehat.
4. Mengetahui hasil pengujian Fungsional, *Usability* dan *Compatibility* pada aplikasi Malang Sehat.

1.4 Manfaat

Manfaat dari penelitian ini adalah:

1. Memudahkan masyarakat di kota Malang dalam memperoleh informasi layanan kesehatan dengan bantuan layanan berbasis lokasi.

1.5 Batasan Masalah

Agar diperoleh hasil pembahasan sesuai dengan yang diharapkan dan tidak melebar, maka penelitian ini dibatasi oleh hal-hal sebagai berikut:

1. Aplikasi hanya memuat informasi layanan kesehatan yang ada di kota Malang.

1.6 Sistematika Pembahasan

Penyusunan dokumen skripsi ini dibagi menjadi beberapa bab, yaitu:

BAB 1 PENDAHULUAN

Bab ini menjelaskan latar belakang penelitian, rumusan masalah, tujuan, manfaat, batasan masalah, dan menjelaskan sistematika pembahasan.

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini menjelaskan landasan pustaka dan dasar teori. Landasan pustaka adalah penguat teori yang berasal dari penelitian lain yang relevan dan dasar teori adalah pembasahan tentang penggunaan teori-teori yang dipakai dalam penelitian ini.

BAB 3 METODE PENELITIAN

Bab ini menjelaskan metode dan langkah-langkah pengerjaan penelitian.

BAB 4 ANALISIS KEBUTUHAN

Bab ini menjelaskan bagaimana kebutuhan didapatkan. Kemudian, kebutuhan yang sudah didapatkan dimuat dalam bentuk *use case diagram* dan *use case skenario*

BAB 5 PERANCANGAN

Bab ini menjelaskan tentang komponen-komponen rancangan yang digunakan untuk membangun sistem pada aplikasi ini.

BAB 6 IMPLEMENTASI

Bab ini menjelaskan tentang implementasi dari komponen-komponen rancangan sistem yang telah dibuat sebelumnya.

BAB 7 PENGUJIAN DAN ANALISIS

Bab ini menjelaskan tentang pengujian yang dilakukan terhadap hasil implementasi sistem yang telah dibuat disertai dengan analisisnya.

BAB 8 PENUTUP

Bab ini menjelaskan kesimpulan dan saran dari pembuatan aplikasi Malang Sehat.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Penelitian berjudul “Sistem Informasi Geografis Letak Lokasi Rumah Sakit dan Apotek Kota Bengkulu Berbasis Android”. Penelitian yang dilakukan oleh Marlina, Deti & Aspriyono pada 2014 ini menggunakan Google Maps dan Google Maps API untuk menentukan lokasi terdekat menuju rumah sakit/apotek. Dalam penerapannya aplikasi yang dibuat oleh penelitian ini menawarkan menu untuk melihat daftar rumah sakit atau daftar apotek untuk memilihnya berdasarkan nama, kemudian dilakukan pencarian rute dari titik lokasi pengguna menuju lokasi rumah sakit atau apotek pilihan. Tersedia juga menu untuk langsung melihat peta dengan penanda disetiap lokasi yang terdapat rumah sakit atau apotek.

Penelitian berjudul “Implementasi Location Based Service Berbasis Android Untuk Mengetahui Posisi User”. Penelitian yang dilakukan Pada 2014 oleh Badrul Anwar, et al. Penelitian ini menjelaskan cara pemanfaatan LBS untuk dapat mengetahui titik lokasi dari perangkat pengguna dengan menggunakan sensor GPS yang tertanam dalam perangkat Android pengguna tersebut. Dalam pengujian aplikasi penelitian ini menggunakan metode *Black-Box Testing*.

Penelitian berjudul “Aplikasi Location-Based Service Pencarian Tempat di Kota Manado Berbasis Android”. Penelitian ini dilakukan oleh B. R. Rompas et al. Penelitian ini bertujuan untuk membuat aplikasi android yang dapat menemukan lokasi dari tempat-tempat di kota Manado seperti rumah sakit, rumah makan, sekolah, dan sebagainya. Dalam penerapannya aplikasi yang dibuat oleh penelitian ini akan menampilkan daftar tempat-tempat yang tersedia, kemudian pengguna dapat memilih tempat yang ingin dicari, setelah itu akan terbuka peta yang menunjukkan lokasi tempat tersebut dan lokasi tempat pengguna berada beserta panduan jalan untuk menuju tempat tersebut.

Dalam penelitian ini akan digunakan LBS sebagai penunjang layanan sebagaimana telah digunakan pada penelitian-penelitian yang telah disebutkan sebelumnya. Perbedaan terkait penelitian ini dengan penelitian-penelitian sebelumnya adalah pada objek yang dikerjakan dan fitur-fitur yang akan diterapkan.

2.2 Dasar Teori

2.2.1 Fasilitas Layanan Kesehatan

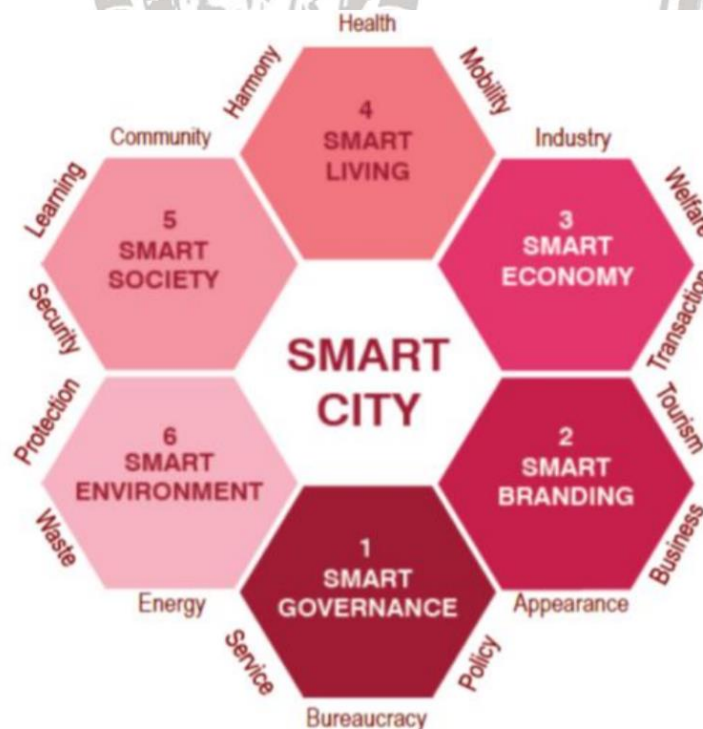
Definisi fasilitas layanan kesehatan atau fasilitas pelayanan kesehatan tertulis dalam Undang-Undang Republik Indonesia Nomor 36 Tahun 2009 Bab 1 pasal 1 ayat 7 adalah “suatu alat dan/atau tempat yang digunakan untuk menyelenggarakan upaya pelayanan kesehatan, baik promotif, preventif, kuratif maupun rehabilitatif yang dilakukan oleh Pemerintah, pemerintah daerah,

dan/atau masyarakat”. Definisi ini kemudian diperjelas dalam Peraturan Pemerintah Nomor 47 Tahun 2016 Pasal 4 ayat 1, Bahwa Jenis Fasilitas Pelayanan Kesehatan terdiri atas:

- A. Tempat Praktik Mandiri Tenaga Kesehatan;
- B. Pusat Kesehatan Masyarakat;
- C. Klinik;
- D. Rumah Sakit;
- E. Apotek;
- F. Unit Transfusi Darah;
- G. laboratorium Kesehatan;
- H. Optikal;
- I. Fasilitas Pelayanan Kedokteran Untuk Kepentingan Hukum; Dan
- J. Fasilitas Pelayanan Kesehatan Tradisional

2.2.2 Smart City

Smart City Menurut Meijer & Bolívar (2015) adalah “kemampuan sebuah kota untuk menarik modal manusia dan memobilisasi sumber daya manusia ini dalam kolaborasi antara berbagai aktor (terorganisir dan individu) melalui penggunaan teknologi informasi dan komunikasi”. Dalam *Smart City* terdapat dimensi yang memiliki beberapa fokus bidang terkait, bidang-bidang tersebut dapat dilihat dalam Gambar 2.1.



Gambar 2.1 Dimensi *Smart city*
(Susanto, 2019)

2.2.2.1 Smart Governance

Smart Governance atau tata kelola pemerintahan kota yang pintar adalah konsep sekaligus penerapan pengelolaan manajemen pemerintahan dan layanan publik secara lebih efisien. Selanjutnya dilakukan juga peningkatan kinerja birokrasi melalui inovasi yang dipadukan dengan perkembangan teknologi. *Smart Governance* direkomendasikan menjadi basis bagi keberhasilan pembangunan dimensi-dimensi Smart City lainnya. Konsep *Smart Governance* harus diterapkan sekaligus diukur dalam 3 sub-dimensi, yakni: Layanan publik (*Service*), Birokrasi (*Bureaucracy*), dan Kebijakan publik (*Policy*) (Susanto, ed., 2019).

2.2.2.2 Smart Branding

Smart Branding atau pemasaran yang pintar adalah praktik inovatif dan kreatif pemerintah daerah memanfaatkan teknologi terkini untuk membangun gambaran dan nilai jual daerah di tingkat nasional maupun internasional. Peningkatan nilai jual tersebut diharapkan mampu meningkatkan daya saing daerah dalam menarik partisipasi masyarakat dan investasi bisnis/investor dari dalam maupun luar daerah guna mendorong aktivitas perekonomian dan pengembangan kehidupan sosial dan budaya lokal yang berujung pada peningkatan kesejahteraan masyarakat. Konsep Smart Branding harus diterapkan sekaligus diukur dalam 3 sub-dimensi, yakni: Pariwisata daerah (*Tourism*), Bisnis daerah (*Business Branding*), dan Wajah kota (*City Appearance*) (Susanto, ed., 2019).

2.2.2.3 Smart Economy

Smart Economy atau tata kelola perekonomian yang pintar adalah kemampuan mewujudkan ekosistem perekonomian daerah yang mendukung sektor ekonomi unggulan daerah dan memenuhi tuntutan era informasi, yakni adaptif terhadap perubahan cepat (*agile*), bersifat global, kolaborasi ekonomi, terintegrasi, inovasi, dan kostumisasi personal. Konsep *Smart Economy* diterapkan sekaligus diukur dalam 3 sub-dimensi, yakni: ekosistem Industri (*industry*), peningkatan Kesejahteraan masyarakat (*welfare*), dan ekosistem transaksi Keuangan (*transaction*) (Susanto, ed., 2019).

2.2.2.4 Smart Living

Smart Living yakni kemampuan menciptakan lingkungan tempat tinggal yang layak, nyaman, dan efisien. Konsep *Smart Living* diterapkan dan diukur dalam 3 sub-dimensi, yakni: Harmonisasi tata ruang wilayah (*harmony*), sarana prasarana Kesehatan (*Health*), dan sarana prasarana Transportasi manusia dan barang (*Mobility*). Dalam sub-dimensi Kesehatan, pemerintah daerah dituntut untuk mampu menyediakan akses terhadap ketersediaan makanan dan minuman sehat (*food*), pelayanan kesehatan yang (*healthcare*), dan sarana dan prasarana olahraga (*sport*) (Susanto, ed., 2019).

2.2.2.5 Smart Society

Smart Society atau masyarakat yang cerdas yakni kemampuan pemerintah daerah dalam mewujudkan ekosistem sosio-teknis (*sosial-virtual*) masyarakat yang humanis, produktif, dinamis, komunikatif, dan interaktif dengan literasi digital yang tinggi. Konsep *Smart Society* diterapkan dan diukur dalam 3 sub-dimensi, yakni: Interaksi masyarakat (*Community*), Ekosistem Belajar (*Learning*), dan Keamanan masyarakat (*Security*) (Susanto, ed., 2019).

2.2.3 Android

Sekitar awal abad ke-21, Android diciptakan oleh Andy Rubin dalam perusahaannya yang bernama Android inc. sebagai sistem operasi untuk telepon seluler. Kemudian pada 2005 perusahaan Google mengakuisisi perusahaan besutan Andy Rubin ini, dan menjadikannya sebagai Direktur Platform seluler di Google (Jackson, 2011). Dengan masuknya Android ke dalam perusahaan pencarian ternama Google, membuat android cepat dikenal dunia dan mengalami perkembangan yang signifikan. Pada tahun 2020 ini android terhitung sudah memiliki 17 versi, dengan 3 versi terbarunya yaitu Oreo, Pie dan Android Q.

Android dapat deskripsikan sebagai sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware* dan aplikasi. Android bersifat *open source*, artinya siapa saja dapat menggunakan dan memodifikasi sistem operasi ini secara bebas. Pengembang aplikasi dapat membuat aplikasi untuk Android dengan menggunakan *Software Developer Kit (SDK)* secara gratis dengan *Integrated Development Environment (IDE)* yang bernama Android Studio. Program Android dapat ditulis dalam bahasa java maupun kotlin (Ekanayake, 2018). Mengingat sifatnya yang *open source*, android memberikan ruang kepada para pengembang untuk dapat berkreasi dengan sistem operasi ini. Sistem operasi ini berhasil memancing perusahaan-perusahaan ponsel terkemuka seperti Samsung, Motorola, Sony, dan Nokia untuk turut serta menggunakannya sebagai sistem operasi ponsel mereka dengan antarmuka yang mereka modifikasi sendiri. Android tidak hanya digunakan pada ponsel pintar saja, beberapa peralatan elektronik lainnya seperti televisi dan jam tangan.

2.2.4 Basis Data MySQL

Menurut Yanto (2016), Basis data dapat diibaratkan sebuah lemari arsip, yang tujuan utamanya adalah kemudahan dalam penyimpanan data dan pengambilan kembali data yang telah disimpan. Karakteristik yang ditonjolkan oleh basis data adalah kemampuan untuk pengaturan, pemilahan, pengelompokan, dan pengorganisasian data.

Menurut Komputer (2010), MySQL adalah basis data dengan sumber terbuka yang mengimplementasikan bahasa SQL (Structured Query Language). Kemudian, MySQL merupakan basis data berstruktur relasional. Prinsip basis data relasional yaitu informasi yang ada di dalamnya dibagi menjadi beberapa data yang terpisah secara logis. Data-data ini diletakkan dalam bentuk tabel. Ketika informasi sudah tersimpan pada tabel-tabel nantinya dapat dilakukan aktivitas

untuk melihat, menambah, mengedit, atau menghapusnya dengan menggunakan *query*. Sebagai contoh, untuk melihat data dapat digunakan *query* "SELECT" dan untuk menghapus data dapat digunakan *query* "DELETE".

2.2.5 Application Programming Interface (API)

Menurut Christensson, P (2016, Juni 20) API merupakan sekumpulan perintah, fungsi, protokol, dan objek yang dapat digunakan oleh programmer untuk membuat perangkat lunak atau berinteraksi sistem eksternal. API memberikan sebuah standar perintah yang dapat digunakan oleh para pengembang, sehingga pengembang tidak perlu menuliskan kode program dari awal untuk dapat berinteraksi dengan sistem. API tersedia untuk beberapa aspek perangkat lunak seperti sistem operasi dekstop, sistem operasi perangkat bergerak, dan situs internet.

Christensson, P (2016, Juni 20) juga menjelaskan beberapa contoh API, yang pertama pada Windows, API memberikan kontrol dan elemen antarmuka pengguna kepada pengembang, seperti jendela, bilah gulir, kotak dialog, perintah untuk mengakses sistem file dan melakukan operasi file. Kedua, API pada sistem operasi iOS memberikan akses kepada pengembang untuk dapat berinteraksi dengan perangkat keras seperti pengeras suara, mikrofon dan kamera. Ketiga, API pada situs internet. jika API pada sistem operasi dekstop dan perangkat bergerak memberikan serangkaian fitur yang kompleks, maka API pada situs internet dapat dikatakan memiliki tingkat kompleksitas dibawahnya, hal ini dikarenakan API situs internet bisa saja hanya berupa seperangkat element XML dengan perintah dasar untuk mengambil informasi.

2.2.6 Google Maps

Google Maps merupakan, aplikasi yang dikembangkan oleh perusahaan Google ini menyediakan fitur-fitur untuk mengolah informasi dari data lokasi. Disebutkan dalam laman support google maps, fitur-fitur tersebut terdiri dari:

1. Menemukan hasil penelusuran
2. Menjelajahi peta
3. Mengukur jarak antar titik
4. Melaporkan masalah data
5. Menemukan info lalu lintas, transportasi umum, bersepeda, dan medan
6. Menemukan dunia dalam 3D
7. Menemukan Street View
8. Menggeser peta
9. Menemukan lokasi saat ini
10. Memberi rating dan ulasan tempat

2.2.6.1 Maps SDK

Maps SDK merupakan pengembangan dari API yang disediakan oleh Google Maps untuk disematkan pada aplikasi perangkat bergerak. Dijelaskan dalam situs Developers Android (2020), Maps SDK memberikan pengembang

sebuah antarmuka untuk menambahkan peta berdasarkan data Google Maps ke dalam aplikasi dengan memanfaatkan API. API ini secara otomatis menangani akses ke server Google Maps, download data, tampilan peta, dan respons terhadap gestur peta. Pengembang juga dapat menggunakan panggilan API untuk menambahkan penanda, poligon, dan overlay ke peta dasar, serta mengubah tampilan area peta tertentu yang dilihat pengguna.

2.2.7 Prototyping

Prototyping adalah metode yang digunakan untuk menunjukkan konsep, opsi desain, dan mencari tahu lebih lanjut tentang masalah dan kemungkinan solusinya (Sommerville, 2011, p. 45). Fungsi yang dijelaskan dalam spesifikasi awal mungkin tampak bermanfaat dan didefinisikan dengan baik. Namun, ketika fungsi itu digabungkan dengan fungsi lain, pengguna sering menemukan bahwa tampilan awal mereka salah atau tidak lengkap. Spesifikasi sistem kemudian dapat dimodifikasi untuk mencerminkan pemahaman mereka yang berubah terhadap persyaratan (Sommerville, 2011, p. 45). Dengan begitu Penggunaan prototype ini dapat mengatasi masalah umum yang sering muncul, yaitu di mana pemangku kepentingan hanya mendefinisikan serangkaian tujuan umum untuk perangkat lunak, tetapi tidak mendefinisikannya secara rinci untuk fungsi dan fitur nya (Pressman, 2010, p. 43).



Gambar 2.2 The Prototyping paradigm

(Pressman, 2010, p. 43)

Paradigma prototyping Dapat dilihat dalam Gambar 2.1. Menurut Pressman (2010:43) paradigma ini dimulai dengan komunikasi. Pengembang aplikasi bertemu dengan pemangku kepentingan untuk menentukan tujuan keseluruhan perangkat lunak, mengidentifikasi persyaratan apa pun yang diketahui, dan

menjabarkan bidang-bidang di mana definisi lebih lanjut wajib. Iterasi *prototype* direncanakan dengan cepat, dan dilakukan pemodelan (dalam bentuk "desain cepat"). Desain cepat berfokus pada representasi aspek-aspek perangkat lunak yang akan terlihat oleh pengguna akhir (misal Tata letak antarmuka). Desain cepat mengarah pada pembangunan *prototype*. *Prototype* ini digunakan dan dievaluasi oleh para pemangku kepentingan, kemudian mereka memberikan umpan balik. Umpan balik ini nantinya akan menjadi pedoman dalam dilakukannya proses iterasi untuk penyempurnaan perangkat lunak.

2.2.8 Location Based Services (LBS)

Istilah *Location Based Services (LBS)* adalah konsep terbaru yang menunjukkan aplikasi yang mengintegrasikan lokasi geografis (yaitu, koordinasi spasial) dengan maksud umum untuk memberikan layanan (Schiller & Voisard, 2004). Sebagian besar penerapan LBS telah menjadi bagian dari kehidupan sehari-hari, yang mana dapat ditemukan pada beberapa perangkat seperti jam tangan pintar, ponsel pintar, dan Navigasi mobil. Pada android sendiri contoh penerapan LBS dapat dijumpai pada aplikasi peta navigasi, aplikasi layanan transportasi online, aplikasi wisata, dan aplikasi perkiraan cuaca.

2.2.9 Teori Pengujian

Pengujian adalah satu set aktifitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan. Aktifitas pengujian terdiri dari satu set atau sekumpulan langkah di mana dapat menempatkan desain kasus uji yang spesifik dan metode pengujian (Rosa & Shalahuddin, 2018).

2.2.9.1 Pengujian Fungsional (Black-Box)

Pada pengujian fungsional akan digunakan metode pengujian *Black-Box*, yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian *black-box* dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan (Rosa & Shalahuddin, 2018).

2.2.9.2 Pengujian Usability

Usability dalam Kamus Besar Bahasa Indonesia diartikan sebagai kegunaan. Kegunaan ini adalah atribut kualitas tingkat kemudahan suatu desain untuk dapat dioperasikan oleh pengguna (Nielsen, 2012). Menurut Nielsen (2012) *Usability* memiliki 5 komponen kualitas, yaitu:

1. *Learnability*: Seberapa mudahnya pengguna menyelesaikan tugas dasar pada saat ia baru pertama kali berinteraksi dengan desain.
2. *Efficiency*: Ketika pengguna sudah mempelajari desain, seberapa cepat mereka dapat menyelesaikan tugas?

3. *Memorability*: Ketika pengguna sudah tidak berinteraksi dengan desain untuk periode waktu tertentu, seberapa mudah mereka untuk dapat menguasainya kembali?
4. *Errors*: Berapa banyak kesalahan yang dilakukan pengguna? Seberapa parah kesalahan tersebut? dan seberapa mudah mereka untuk memperbaiki kesalahan tersebut?
5. *Satisfaction*: Seberapa menyenangkannya desain bagi pengguna?

Pengujian *Usability* dilakukan dengan cara peneliti (fasilitator / moderator) meminta partisipan untuk melakukan tugas, biasanya menggunakan antarmuka yang spesifik. Ketika partisipan menyelesaikan tugasnya, peneliti mengamati perilaku partisipan dan mendengarkan masukan dari partisipan terhadap desain (Moran, 2019).

Pengujian akan dilakukan menggunakan *System Usability Scale (SUS)*, yaitu 10 skala sederhana yang memberikan pandangan umum dari penilaian subyektif tentang "*Usability*" (Brooke, 1996). Penilaian SUS terdiri dari pertanyaan positif pada nomor ganjir dan pertanyaan negatif pada nomor genap yang disajikan dalam bentuk jawaban berupa skala 1-5. Penilaian dilakukan dengan memberikan nilai 1-4. Pada pertanyaan positif nilai didapatkan dari jawaban dikurangi dengan 1, sedangkan pada pertanyaan negatif nilai didapatkan dari 5 dikurangi dengan jawaban. Selanjutnya dilakukan penjumlahan nilai dari pertanyaan negatif dan positif untuk mendapatkan nilai rentang 0-40, kemudian dikalikan dengan 2,5 untuk menghasilkan nilai SUS rentang 0-100. Untuk dapat dikatakan lulus uji maka nilai akhir yang diperoleh harus diatas 68 (Sauro & Lewis, 2012). Adapun penilaian *adjectiva rating* menurut (Bangor, et al., 2009) pada nilai akhir SUS dapat dilihat pada Tabel 2.1.

Tabel 2.1 Adjectiva Rating SUS

Adjective	Mean SUS Score
Worst Imaginable	12.5
Awful	20.3
Poor	35.7
OK	50.9
Good	71.4
Excellent	85.5
Best Imaginable	90.9

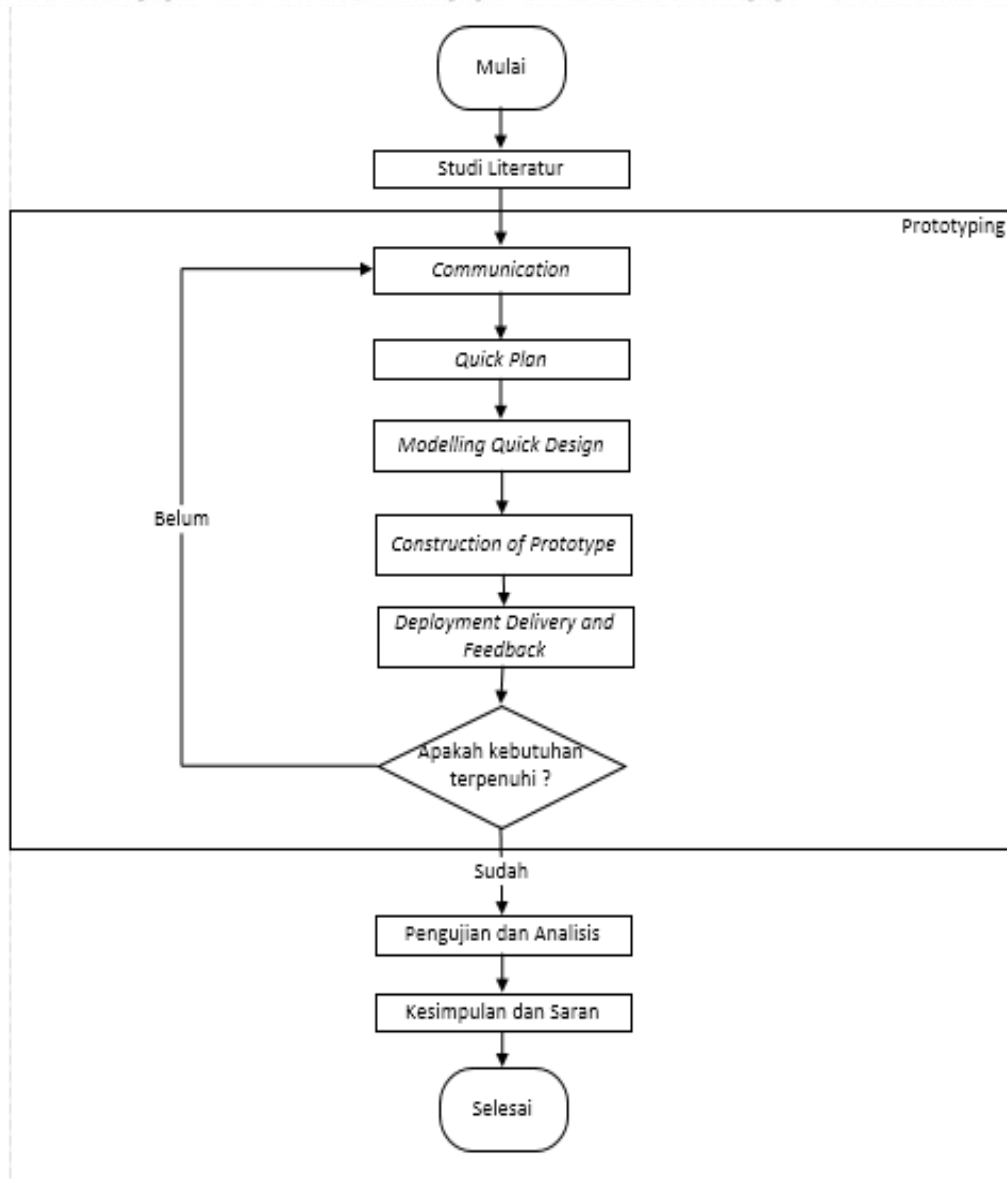
2.2.9.3 Pengujian *Compatibility*

Compatibility dalam bahasa Indoensia disebut kompatibilitas, dalam Kamus Besar Bahasa Indonesia, kompatibilitas sendiri dapat diartikan sebagai "keadaan penyesuaian diri". Untuk lebih detailnya menurut Zhang, et al. (2015), pengujian ini merupakan pengujian yang memvalidasi apakah aplikasi mobile dapat bekerja dengan baik pada perangkat mobile yang berbeda dengan berbagai platform, peralatan dan fitur, dan di lingkungan yang berbeda. Pengujian

dinyatakan berhasil apabila perangkat yang ditentukan berhasil menyelesaikan kasus uji yang diberikan.



BAB 3 METODE PENELITIAN



Gambar 3.1 Diagram Alir Pengerjaan Skripsi

Dalam mengembangkan sebuah perangkat lunak, dibutuhkan penyusunan langkah-langkah penelitian agar proses pengembangan dapat dilaksanakan dengan baik. Langkah-langkah tersebut dituangkan dalam diagram Alir yang terdapat dalam Gambar 3.1.

3.1 Studi Literatur

Studi literatur merupakan tahapan dalam pencarian teori-teori dan pustaka yang relevan dengan permasalahan yang ada untuk dijadikan bahan belajar dan pemahaman sehingga dapat menguatkan landasan pengetahuan dalam proses

penelitian. Pertama dilakukan pencarian terhadap penelitian sebelumnya yang relevan dengan penelitian ini, penelitian tersebut adalah:

1. Penelitian berjudul “Sistem Informasi Geografis Letak Lokasi Rumah Sakit Dan Apotek Kota Bengkulu Berbasis Android”. Penelitian yang dilakukan oleh Marlina, Deti & Aspriyono, Hari pada 2014.
2. Penelitian berjudul “Implementasi Location Based Service Berbasis android Untuk Mengetahui posisi User”. Penelitian yang dilakukan Pada 2014 oleh Badrul Anwar, et al.
3. Penelitian berjudul “Aplikasi Location-Based Service Pencarian Tempat Di Kota Manado Berbasis Android”. Penelitian ini dilakukan pada 2012 oleh B. R. Rompas et al.

Selanjutnya dilakukan pengkajian terhadap teori-teori yang relevan, diantaranya adalah:

1. Fasilitas Pelayanan Kesehatan
2. *Smart City*
3. Android
4. *Application Programming Interface (API)*
5. Google Maps
6. Maps SDK
7. *Prototyping*
8. *Location Based Services (LBS)*
9. Pengujian Fungsional
10. Pengujian *Usability*
11. Pengujian *Compatibility*

3.2 Prototyping

3.2.1 Communication (Komunikasi)

Pada tahap ini, akan dilakukan komunikasi dengan pemangku kepentingan untuk menentukan tujuan keseluruhan sistem yang akan dibangun. Selanjutnya akan dilakukan komunikasi juga dengan masyarakat kota kota untuk mendapatkan informasi tambahan terkait kebutuhan yang diperlukan. Tahapan ini akan dimasukkan ke dalam bab Analisis Kebutuhan.

3.2.2 Quick Plan (Rencana Singkat)

Pada tahap ini, hasil dari tahap komunikasi dituangkan dalam bentuk poin-poin kebutuhan fungsional dan non fungsional. Setelah itu kebutuhan fungsional akan dijabarkan lebih lanjut menggunakan *use case diagram* dan *use case* skenario. Tahapan ini akan dimasukkan ke dalam bab Analisis Kebutuhan.

3.2.3 Modelling Quick Design (Pemodelan Desain Cepat)

Pada tahap ini, akan dibuat desain pemodelan berupa perancangan sistem, arsitektur sistem, perancangan *sequence diagram*, perancangan *class diagram*,

perancangan tabel basis data, dan perancangan antarmuka dari hasil *use case* tahap sebelumnya. Tahapan ini akan dimasukkan ke dalam bab Perancangan.

3.2.4 Construction of Prototype (Pembangunan Prototipe)

Pada tahap ini, dilakukan implementasi prototipe dari hasil perancangan pada tahap sebelumnya yaitu berupa implementasi basis data, implementasi kode program dan implementasi antarmuka. Tahapan ini akan dimasukkan ke dalam bab Implementasi.

3.2.5 Deployment Delivery and Feedback (Penyebaran dan Umpan balik)

Pada tahap ini, dilakukan presentasi prototipe yang telah selesai diimplementasikan kepada pemangku kepentingan. Kemudian pemangku kepentingan akan memberikan umpan balik. Apabila pemangku kepentingan merasa kebutuhan belum terpenuhi, akan dilakukan iterasi kembali dimulai dari tahap *Communication*. Tahapan ini akan dimasukkan ke dalam bab Implementasi.

3.3 Pengujian dan Analisis

Pengujian dan Analisis merupakan tahapan untuk pengecekan apakah implementasi yang dibuat dapat berjalan dengan semestinya. Metode pengujian yang digunakan adalah pengujian *Black-box* untuk melakukan pengujian pada Fungsional. Pengujian *Usability* untuk mengetahui tingkat kegunaan aplikasi, dan pengujian *Compatibility* untuk melakukan pengujian aplikasi terhadap perangkat lunak yang akan digunakan oleh pengguna.

3.4 Kesimpulan dan Saran

Kesimpulan merupakan tahapan di mana dilakukan penarikan jawaban dari rumusan masalah yang ada. Selanjutnya juga akan diberikan saran yang bertujuan untuk memperbaiki kesalahan-kesalahan yang ada pada segi penulisan ataupun pada segi proses penelitian itu sendiri.

BAB 4 ANALISIS KEBUTUHAN

Bab ini berisikan bahasan tentang analisis kebutuhan pada modul Informasi Obyeaplikasi Malang Sehat, yaitu aplikasi yang menyediakan informasi layanan kesehatan berbasis lokasi di kota Malang. Analisis kebutuhan yang dijabarkan pada bab ini yang terdiri kebutuhan fungsional, kebutuhan non fungsional, *use case* diagram dan *use case* skenario.

4.1 Gambaran Umum Aplikasi

Penelitian ini mengembangkan Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat, menyajikan informasi tentang layanan kesehatan khususnya di kota Malang. Aplikasi ini nantinya akan berjalan pada sistem operasi *Android*. Informasi yang disajikan dalam aplikasi secara umum dapat dibagi menjadi dua jenis yaitu informasi penyedia layanan kesehatan dan informasi dokter. Penyedia layanan kesehatan terdiri dari praktik mandiri, Puskesmas, Klinik, Rumah sakit, Apotek, Unit Transfusi Darah (UTD), Laboratorium, dan Optik di kota Malang. Informasi mengenai penyedia layanan kesehatan tersebut didukung dengan layanan berbasis lokasi yang dapat memudahkan pengguna dalam beberapa aspek yaitu mengetahui lokasi detail dari penyedia layanan kesehatan dengan bantuan visual denah, mengetahui lokasi pengguna saat ini dan mengetahui rute menuju lokasi penyedia layanan kesehatan berada dengan terintegrasinya aplikasi ini dengan aplikasi Google Maps.

4.2 Komunikasi

4.2.1 Iterasi 0

Dalam komunikasi iterasi 0, dilakukan wawancara dengan pemangku kebutuhan dan masyarakat kota Malang untuk penggalian informasi mengenai kebutuhan. Dari hasil wawancara yang didapat, terdapat beberapa point kesimpulan yang dapat diambil, yaitu:

1. Sistem yang ada masih perlu ditingkatkan lagi untuk memudahkan masyarakat kota Malang dalam mengakses informasi kesehatan baik itu melalui website maupun aplikasi dengan informasi yang valid dan lengkap.
2. Masyarakat menggunakan ponsel cerdas untuk melakukan pencarian manual pada mesin pencari tentang informasi fasilitas pelayanan kesehatan serta informasi dokter.
3. Beberapa masyarakat mencari informasi layanan kesehatan dari mulut ke mulut kepada teman atau kerabat.
4. Beberapa masyarakat memilih layanan kesehatan yang berada disekitarnya saja.

5. Beberapa masyarakat lebih mengutamakan layanan kesehatan yang memberikan layanan pembayaran melalui BPJS
6. Beberapa masyarakat kesulitan dalam menghubungi layanan kesehatan yang ada.
7. Beberapa masyarakat tidak mengetahui apakah spesialisasi dokter yang dicari terdapat pada suatu layanan kesehatan atau tidak.

Hasil wawancara komunikasi pemangku kepentingan iterasi 0 terlampir pada Lampiran A.2 dan hasil wawancara kelompok pengguna terlampir pada Lampiran A.1.

4.2.2 Iterasi 1

Dalam komunikasi iterasi 1, dilakukan wawancara kembali dengan pemangku kepentingan. Dari proses wawancara didapatkan kesimpulan bahwa perlu ditambahkan fitur personalisasi pengguna, dimana pengguna nantinya dapat menyimpan data informasi objek layanan kesehatan. Hasil wawancara komunikasi pemangku kepentingan iterasi 1 terlampir pada Lampiran A.4.

4.3 Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang secara langsung berinteraksi dengan pengguna. Sistem harus dapat memenuhi kebutuhan fungsional untuk dapat dikatakan berhasil.

4.3.1 Iterasi 0

Didapatkan 3 Kebutuhan fungsional pada iterasi 0 yang dapat dilihat pada Tabel 4.1.

Tabel 4.1 Kebutuhan Fungsional Iterasi 0

No	Kode	Nama Kebutuhan	Deskripsi
1	MS-F01	Lihat Informasi Objek Layanan Kesehatan	Sistem dapat menampilkan informasi nama, foto, alamat, waktu operasional, nomor telepon, deskripsi, dukungan pembayaran, fasilitas medis, daftar spesialis tersedia dan daftar dokter tersedia dari fasilitas kesehatan.
2	MS-F02	Lihat Informasi Dokter	Sistem dapat menampilkan informasi nama, foto, spesialis, profil, pengalaman, pendidikan, waktu praktik, dan tempat praktik dari dokter.
3	MS-F03	Lihat Denah Objek Layanan Kesehatan	Sistem dapat menampilkan informasi seluruh objek layanan kesehatan yang tersedia dalam satu tampilan denah.

4.3.2 Iterasi 1

Kebutuhan fungsional yang telah dirumuskan pada iterasi 0 tidak mengalami perubahan, akan tetapi mendapatkan tambahan 4 fungsionalitas baru yang dapat dilihat pada Tabel 4.2.

Tabel 4.2 Kebutuhan Fungsional Iterasi 1

No	Kode	Nama Kebutuhan	Deskripsi
1	MS-F04	Masuk Akun	Sistem dapat menyediakan layanan bagi pengguna untuk masuk menggunakan username dan password.
2	MS-F05	Daftar Akun	ketika pengguna belum memiliki akun, maka sistem dapat menyediakan layanan untuk daftar menggunakan username dan password.
3	MS-F06	Keluar Akun	Sistem dapat menyediakan layanan bagi pengguna untuk keluar dari akun yang sedang masuk ke dalam sistem.
4	MS-F07	Penanda	Sistem dapat menyediakan menu untuk menandai objek layanan kesehatan atau dokter yang dapat dilihat ulang oleh pengguna

4.4 Kebutuhan Non-Fungsional

4.4.1 Iterasi 0

Kebutuhan non-fungsional merupakan kebutuhan yang secara tidak langsung memberikan kepuasan kepada pengguna saat menggunakan aplikasi. Kebutuhan ini diajukan kepada pemangku kepentingan dan telah memenuhi kebutuhan pada iterasi 0. Kebutuhan non-fungsional dapat dilihat pada Tabel 4.3

Tabel 4.3 Kebutuhan Non-Fungsional Iterasi 0

No	Kode	Nama Kebutuhan	Deskripsi
1	MS-NF01	<i>Compatibility</i>	Aplikasi dapat berjalan pada perangkat bergerak berbasis Android dengan berbagai versi sesuai dengan batasan.
2	MS-NF02	<i>Usability</i>	Aplikasi memiliki kegunaan bagi pengguna dan dapat digunakan dengan mudah.

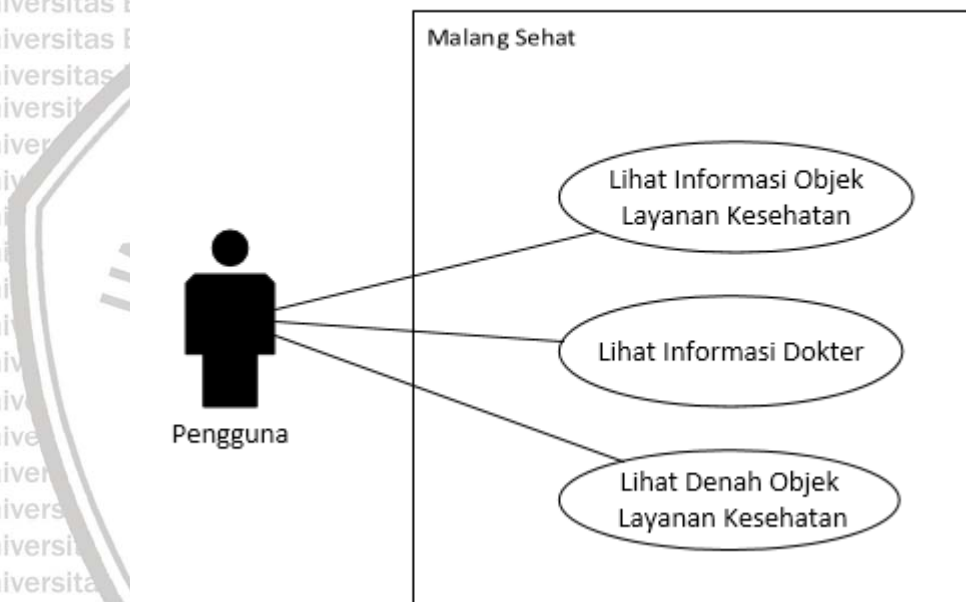
Tabel 4.3 Kebutuhan Non-Fungsional Iterasi 0 (lanjutan)

No	Kode	Nama Kebutuhan	Deskripsi
3	MS-NF03	Reliability	Aplikasi dapat memberikan hasil yang konsisten apabila dijalankan berulang.

4.5 Use Case Diagram

Use case diagram menggambarkan diagram dari interaksi yang terjadi antara aktor dan use case.

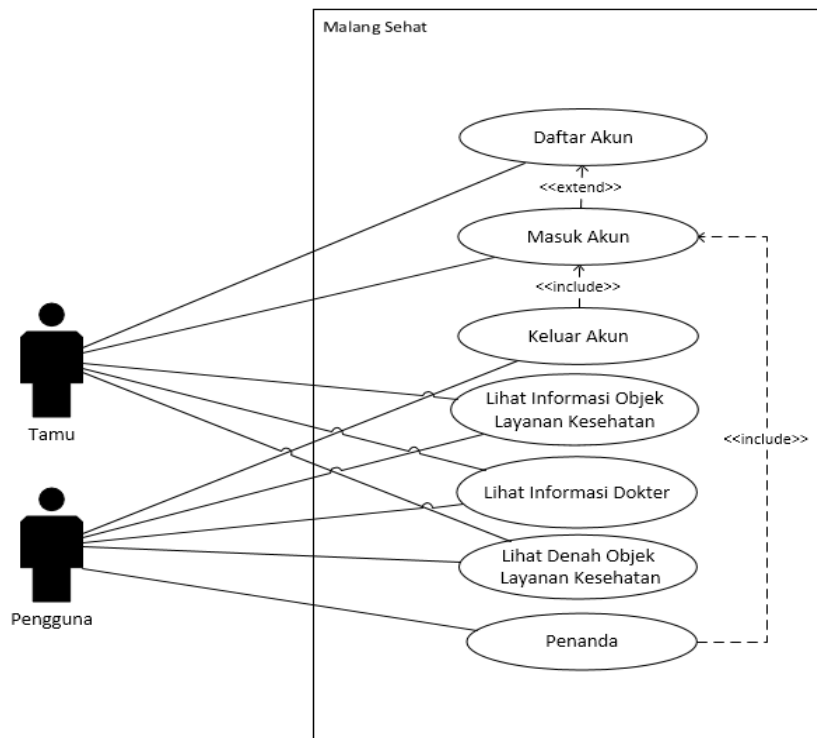
4.5.1 Iterasi 0



Gambar 4.1 Use Case Diagram Iterasi 0

Dalam use case diagram iterasi 0 terdapat satu aktor yaitu Pengguna, dan 3 buah use case yaitu lihat informasi objek layanan kesehatan, lihat informasi dokter dan lihat denah objek layanan kesehatan. Use case diagram iterasi 0 dapat dilihat dalam Gambar 4.1.

4.5.2 Iterasi 1



Gambar 4.2 Use Case Diagram Iterasi 1

Dalam use case diagram iterasi 1 terdapat dua aktor yaitu Tamu dan Pengguna, serta 7 buah use case yang beberapa diantaranya berelasi. Pertama use case penanda dan Keluar Akun include pada use case Masuk Akun, berarti kedua use case ini memerlukan use case Masuk Akun untuk menjalankan aksinya. Selanjutnya use case Masuk Akun adalah extend dari use case Daftar Akun, berarti use case Masuk Akun adalah use case yang ditambahkan dari use case Daftar Akun, tetapi use case Masuk Akun tetap dapat berdiri sendiri. Use Case Diagram iterasi 1 dapat dilihat dalam Gambar 4.2.

4.6 Use Case Skenario

Use case skenario menjelaskan rincian point-point skenario terjadinya sebuah use case yang terdapat pada use case diagram.

4.6.1 Iterasi 0

Use case skenario iterasi 0 akan dijabarkan pada Tabel 4.4 – Tabel 4.7.

Tabel 4.4 Use Case Skenario Lihat Informasi Objek Layanan Kesehatan

Aktor	Tamu, Pengguna
Tujuan	Aktor mendapatkan informasi dari objek layanan kesehatan
Pra-kondisi	Aktor berada di halaman beranda

Tabel 4.4 Use Case Skenario Lihat Informasi Objek Layanan Kesehatan (lanjutan)

Alur Utama	<ol style="list-style-type: none"> 1. Aktor memilih kategori objek layanan kesehatan 2. Sistem memberikan pilihan kecamatan dari objek layanan kesehatan yang ingin ditampilkan 3. Aktor memilih kecamatan 4. Sistem menampilkan daftar objek layanan kesehatan beserta opsi untuk penyaringan dan pencarian berdasarkan nama 5. Aktor memilih objek layanan kesehatan 6. Sistem menampilkan informasi objek layanan kesehatan beserta tombol Panggil Nomor, Navigasi, dan Tandai.
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Pasca-kondisi	Informasi objek layanan kesehatan berhasil ditampilkan

Tabel 4.5 Use Case Skenario Lihat Informasi Dokter

Aktor	Tamu, Pengguna
Tujuan	Aktor mendapatkan detail informasi dokter
Pra-kondisi	Aktor berada di halaman beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol spesialis 2. Sistem menampilkan daftar spesialis 3. Aktor memilih spesialisasi dokter 4. Sistem menampilkan daftar dokter yang dapat disaring dan dilakukan pencarian menggunakan kata kunci 5. Aktor memilih dokter 6. Sistem menampilkan informasi dokter beserta tombol untuk menandai dokter.
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Pasca-kondisi	Informasi dokter berhasil ditampilkan

Tabel 4.6 Use Case Skenario Lihat Denah Objek Layanan Kesehatan

Aktor	Tamu, Pengguna
Tujuan	Aktor dapat mengetahui informasi layanan kesehatan yang disajikan dalam denah

Tabel 4.6 Use Case Skenario Lihat Denah Objek Layanan Kesehatan (lanjutan)

Pra-kondisi	Aktor berada pada halaman beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Denah Lokasi” 2. Sistem menampilkan pilihan <i>checkbox</i> kategori objek layanan kesehatan 3. Aktor memilih kategori apa saja yang ingin ditampilkan 4. Sistem menampilkan marker layanan kesehatan 5. Aktor memilih marker 6. Sistem memberikan tombol untuk menuju halaman informasi dan tombol navigasi 7. Sistem menampilkan halaman detail informasi bila aktor memilih detail informasi atau menampilkan halaman Navigasi pada Google Maps apabila memilih Navigasi
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Pasca-kondisi	Denah lokasi objek layanan kesehatan berhasil ditampilkan

Tabel 4.7 Use Case Skenario Penanda

Aktor	Pengguna
Tujuan	Aktor dapat melihat objek layanan kesehatan atau dokter yang telah ditandai
Pra-kondisi	Aktor telah masuk ke dalam sistem dan berada di halaman beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Penanda” 2. Sistem akan menampilkan tombol pilihan “Dokter” dan “Layanan Kesehatan” yang berisi dokter atau layanan kesehatan yang telah ditandai 3. Sistem menampilkan daftar objek layanan kesehatan atau dokter yang telah ditandai
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Pasca-kondisi	Aktor berhasil mengakses penanda

4.6.2 Iterasi 1

Pada iterasi 1 ini terdapat 4 use case skenario baru, yaitu Masuk akun, Daftar akun, Keluar akun, dan Penanda. Kemudian terdapat perubahan pada use

case skenario iterasi 0, yaitu penambahan aktor Tamu pada use case skenario lihat informasi objek layanan kesehatan, lihat informasi dokter dan lihat denah objek layanan kesehatan. Use case skenario iterasi 1 dapat dilihat pada Tabel 4.8 – 4.14.

Tabel 4.8 Use Case Skenario Masuk Akun

Aktor	Tamu
Tujuan	Aktor dapat masuk ke dalam akun yang telah terdaftar
Pra-kondisi	Aktor berada pada halaman beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Penanda” 2. Sistem menampilkan form dengan kolom “Nama”, “Kata Sandi”, dan tombol “Masuk” 3. Aktor mengisi nama dan kata sandi 4. Aktor menekan tombol “Masuk”
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan “Nama / Kata Sandi salah” saat nama/kata sandi yang dimasukkan salah 2. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Pasca-kondisi	Aktor berhasil masuk ke dalam akun yang telah terdaftar dan halaman berpindah ke Beranda

Tabel 4.9 Use Case Skenario Daftar Akun

Aktor	Tamu
Tujuan	Aktor dapat daftar akun baru
Pra-kondisi	Aktor berada pada halaman beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Penanda” 2. Sistem menyediakan tombol “Daftar” 3. Aktor menekan tombol “Daftar” 4. Sistem menampilkan form yang terdiri dari kolom “nama”, kolom “Kata Sandi”, kolom “Ulangi Kata Sandi”, dan tombol “Daftar” 5. Aktor mengisi form 6. Aktor menekan tombol “Daftar”
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan “Nama Sudah Terpakai” apabila nama yang dimasukkan sudah ada dalam basis data 2. Sistem menampilkan “Nama / Kata Sandi tidak boleh kosong” apabila nama / kata sandi kosong. 3. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data

Tabel 4.9 Use Case Skenario Daftar Akun (lanjutan)

Pasca-kondisi	Aktor berhasil mendaftar akun baru dan halaman kembali ke Masuk
---------------	---

Tabel 4.10 Use Case Skenario Keluar Akun

Aktor	Pengguna
Tujuan	Aktor dapat mengeluarkan akun yang sedang masuk ke sistem.
Pra-kondisi	Aktor telah masuk ke dalam sistem dan berada pada halaman Beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Penanda” 2. Sistem menampilkan tombol “Keluar” 3. Aktor menekan tombol “Keluar”
Alur Alternatif	-
Pasca-kondisi	Aktor berhasil mengeluarkan akun yang sedang masuk ke sistem.

Tabel 4.11 Use Case Skenario Lihat Informasi Objek Layanan Kesehatan

Aktor	Tamu, Pengguna
Tujuan	Aktor mendapatkan informasi dari objek layanan kesehatan
Pra-kondisi	Aktor berada di halaman beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor memilih kategori objek layanan kesehatan 2. Sistem memberikan pilihan kecamatan dari objek layanan kesehatan yang ingin ditampilkan 3. Aktor memilih kecamatan 4. Sistem menampilkan daftar objek layanan kesehatan beserta opsi untuk penyaringan dan pencarian berdasarkan nama 5. Aktor memilih objek layanan kesehatan 6. Sistem menampilkan informasi objek layanan kesehatan beserta tombol Panggil Nomor, Navigasi, dan Tandai.
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Pasca-kondisi	Informasi objek layanan kesehatan berhasil ditampilkan

Tabel 4.12 Use Case Skenario Lihat Informasi Dokter

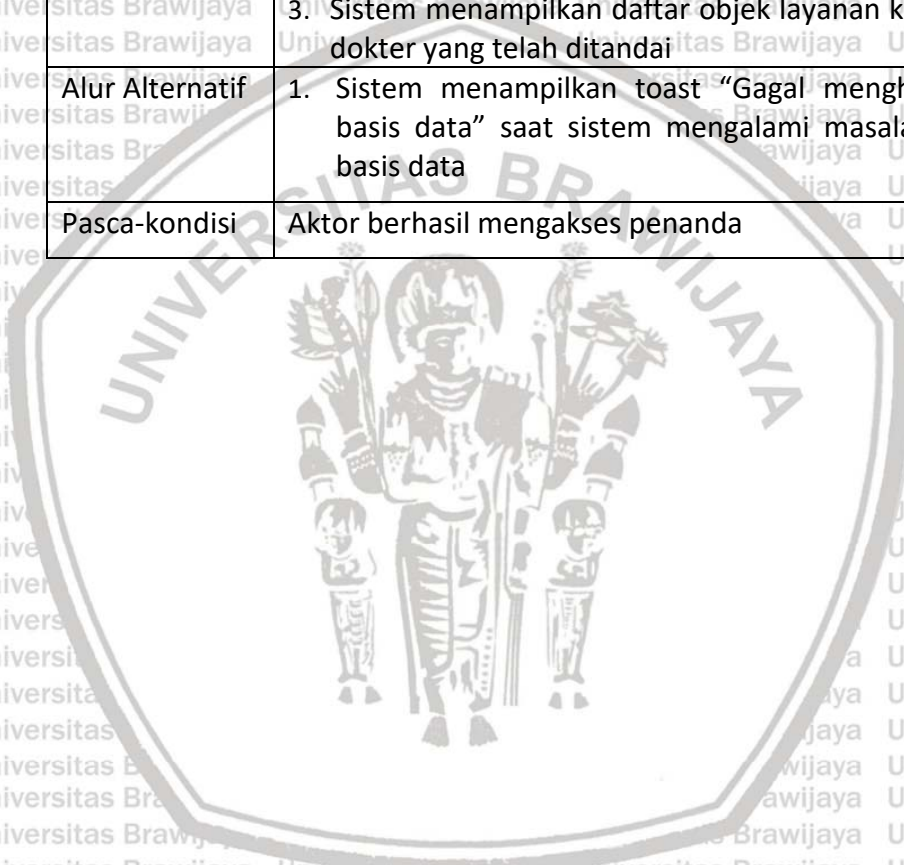
Aktor	Tamu, Pengguna
Tujuan	Aktor mendapatkan detail informasi dokter
Pra-kondisi	Aktor berada di halaman beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol spesialis 2. Sistem menampilkan daftar spesialis 3. Aktor memilih spesialisasi dokter 4. Sistem menampilkan daftar dokter yang dapat disaring dan dilakukan pencarian menggunakan kata kunci 5. Aktor memilih dokter 6. Sistem menampilkan informasi dokter beserta tombol untuk menandai dokter.
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Pasca-kondisi	Informasi dokter berhasil ditampilkan

Tabel 4.13 Use Case Skenario Lihat Denah Objek Layanan Kesehatan

Aktor	Tamu, Pengguna
Tujuan	Aktor dapat mengetahui informasi layanan kesehatan yang disajikan dalam denah
Pra-kondisi	Aktor berada pada halaman beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Denah Lokasi” 2. Sistem menampilkan pilihan <i>checkbox</i> kategori objek layanan kesehatan 3. Aktor memilih kategori apa saja yang ingin ditampilkan 4. Sistem menampilkan marker layanan kesehatan 5. Aktor memilih marker 6. Sistem memberikan tombol untuk menuju halaman informasi dan tombol navigasi 7. Sistem menampilkan halaman detail informasi bila aktor memilih detail informasi atau menampilkan halaman Navigasi pada Google Maps apabila memilih Navigasi
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Pasca-kondisi	Denah lokasi objek layanan kesehatan berhasil ditampilkan

Tabel 4.14 Use Case Skenario Penanda

Aktor	Pengguna
Tujuan	Aktor dapat melihat objek layanan kesehatan atau dokter yang telah ditandai
Pra-kondisi	Aktor telah masuk ke dalam sistem dan berada di halaman beranda
Alur Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Penanda” 2. Sistem akan menampilkan tombol pilihan “Dokter” dan “Layanan Kesehatan” yang berisi dokter atau layanan kesehatan yang telah ditandai 3. Sistem menampilkan daftar objek layanan kesehatan atau dokter yang telah ditandai
Alur Alternatif	<ol style="list-style-type: none"> 1. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Pasca-kondisi	Aktor berhasil mengakses penanda



BAB 5 PERANCANGAN

Bab ini berisikan bahasan tentang rancangan dari Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat. Perancangan akan terdiri dari perancangan arsitektur sistem, perancangan *sequence* diagram, perancangan *class* diagram, perancangan basis data, serta perancangan antar muka.

5.1 Perancangan Arsitektur Sistem



Gambar 5.1 Skema Arsitektur Aplikasi Malang Sehat

Kode program aplikasi Malang Sehat secara umum dipecah menjadi dua bagian, yaitu *back-end* dan *front-end*. Pada bagian *back-end* dirancang dengan framework *Codeigniter3* menggunakan desain arsitektur yang berbasis *Model View Controller (MVC)*. Seperti namanya, *MVC* dalam penerapannya membagi kode program menjadi 3 bagian yaitu *Model*, *View* dan *Controller*. *Model* merupakan bagian kode program yang merepresentasikan data dalam basis data, *View* merupakan bagian kode program yang memuat tampilan halaman sebagai antarmuka pengguna dan *Controller* merupakan bagian kode program yang mengatur data apa saja yang masuk ke dalam *View*.

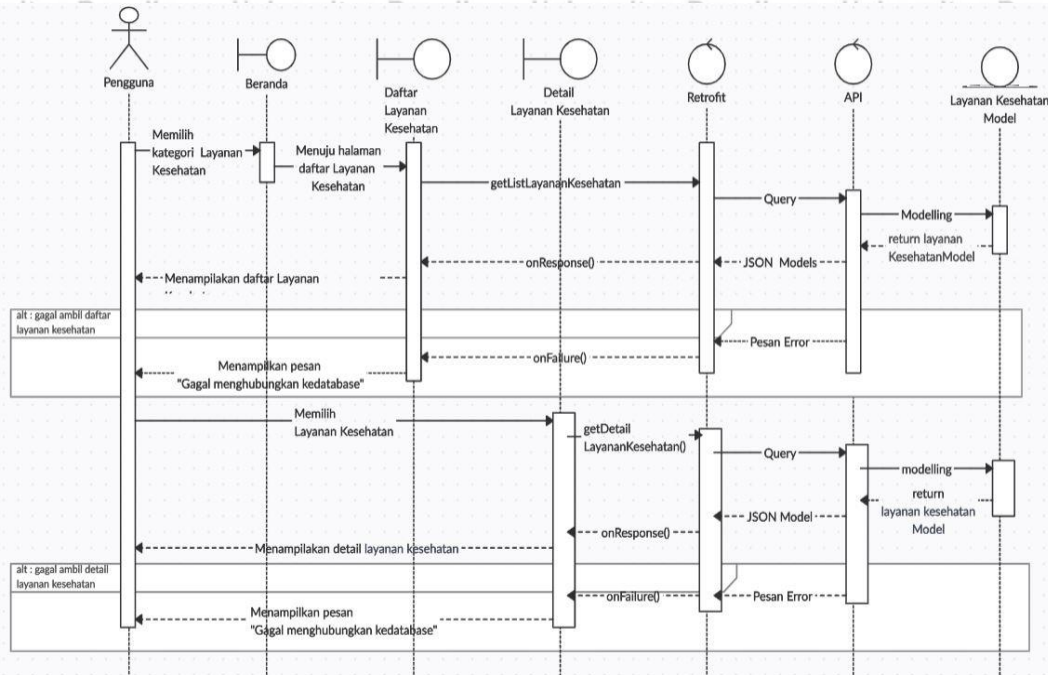
Namun pada aplikasi ini dilakukan penyesuaian fungsi dengan hanya menggunakan *Model* dan *Controller* saja sebagai *API* yang menerima request dan memberikan nilai kembalian berupa data *JSON*. Data *JSON* inilah yang nantinya akan dirubah menjadi *Model* dalam bahasa java menggunakan *framework* *Retrofit* dan diolah dalam bagian *frontend* *Android*. Skema Arsitektur aplikasi Malang Sehat dapat dilihat dalam Gambar 5.1. Dalam gambar tersebut diilustrasikan bagaimana aplikasi ini berinteraksi dengan objek luar untuk dapat berjalan.

5.2 Perancangan Sequence Diagram

Perancangan *Sequence Diagram* berisikan diagram alur interaksi yang terjadi di dalam proses berjalanya suatu fungsionalitas.

5.2.1 Iterasi 0

5.2.1.1 Lihat Informasi Objek Layanan Kesehatan



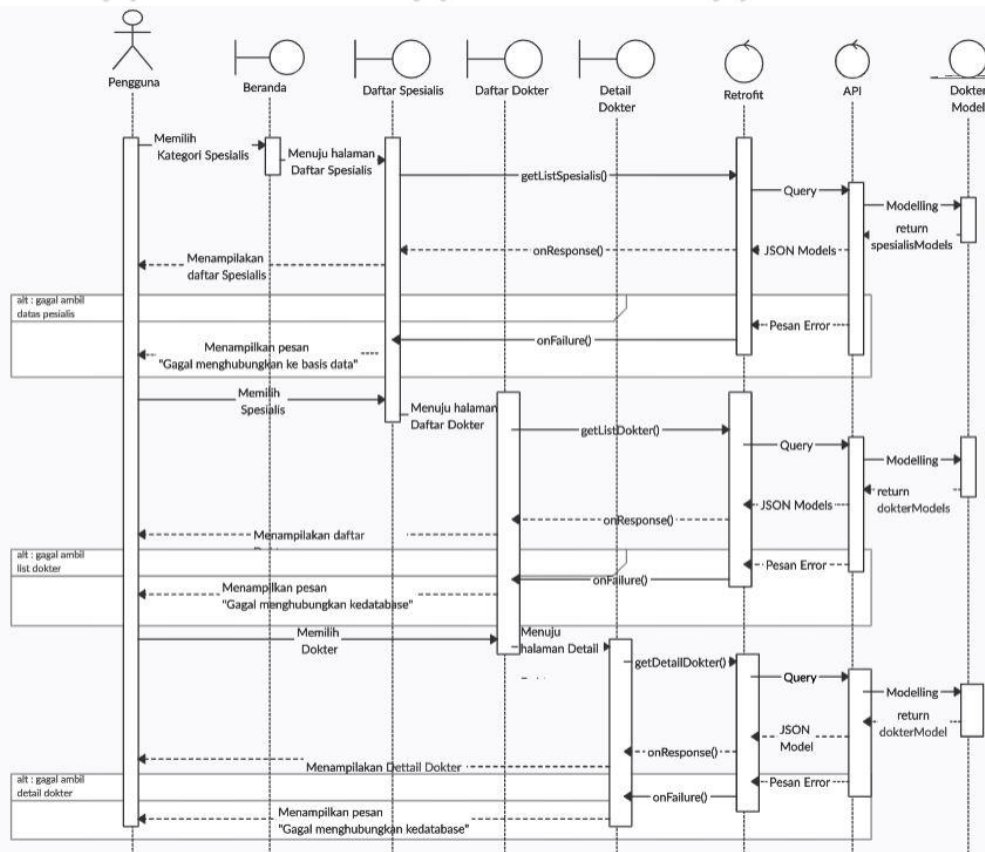
Gambar 5.2 Sequence Diagram Lihat Informasi Objek Layanan Kesehatan

Sequence diagram Lihat Informasi Objek Layanan Kesehatan yang dapat dilihat dalam Gambar 5.2 memiliki 1 aktor yaitu Pengguna. Pada *sequence diagram* ini terjadi interaksi aktor memilih kategori layanan kesehatan pada halaman beranda, selanjutnya halaman berpindah dari beranda menuju halaman Daftar Layanan Kesehatan sesuai dengan kategori yang dipilih, kemudian dipanggil fungsi `getListLayananKesehatan()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit akan mengirim objek yang ditangkap halaman Daftar Layanan Kesehatan kesehatan pada fungsi `onResponse()` dan kemudian daftar layanan kesehatan akan ditampilkan. Apabila daftar objek layanan kesehatan gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar layanan kesehatan pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

Setelah daftar ditampilkan, aktor memilih objek layanan kesehatan dan halaman akan berpindah ke halaman Detail Layanan Kesehatan. Pada halaman Detail Layanan Kesehatan dipanggil fungsi `getDetailLayananKesehatan()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit

akan mengirim objek yang ditangkap halaman Daftar Layanan Kesehatan kesehatan pada fungsi `onResponse()` dan kemudian detail layanan kesehatan akan ditampilkan. Apabila detail objek layanan kesehatan gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar layanan kesehatan pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

5.2.1.2 Lihat Informasi Dokter



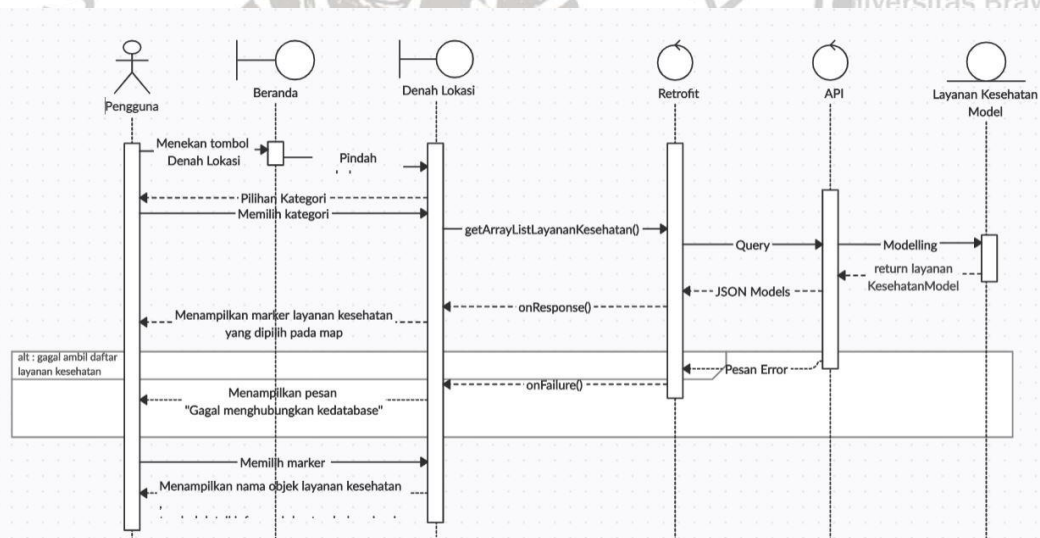
Gambar 5.3 Sequence Diagram Lihat Informasi Dokter

Sequence diagram Lihat Informasi Dokter yang dapat dilihat Dalam Gambar 5.3 memiliki 1 aktor yaitu Pengguna. Pada *sequence* diagram ini terjadi interaksi aktor memilih kategori Spesialis pada halaman beranda, selanjutnya halaman berpindah dari beranda menuju halaman Daftar Spesialis, kemudian dipanggil fungsi `getListSpesialis()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit akan mengirim objek yang ditangkap halaman Daftar Spesialis pada fungsi `onResponse()` dan kemudian daftar layanan kesehatan akan ditampilkan. Apabila daftar objek layanan kesehatan gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar spesialis pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

Selanjutnya aktor memilih spesialisasi pada halaman Daftar Spesialis , kemudian halaman berpindah dari Daftar Spesialis menuju halaman Daftar Dokter, selanjutnya dipanggil fungsi `getListDokter()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, kemudian retrofit akan mengirim objek yang ditangkap halaman Daftar Dokter pada fungsi `onResponse()` lalu daftar layanan kesehatan akan ditampilkan. Apabila daftar dokter gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar layanan kesehatan pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

Setelah daftar dokter ditampilkan, aktor memilih objek dokter dan halaman akan berpindah ke halaman Detail Dokter. Pada halaman Detail Dokter dipanggil fungsi `getDetailDokter()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit akan mengirim objek yang ditangkap halaman Daftar Layanan Kesehatan kesehatan pada fungsi `onResponse()` dan kemudian detail dokter akan ditampilkan. Apabila detail dokter gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar layanan kesehatan pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

5.2.1.3 Lihat Denah Objek Layanan Kesehatan



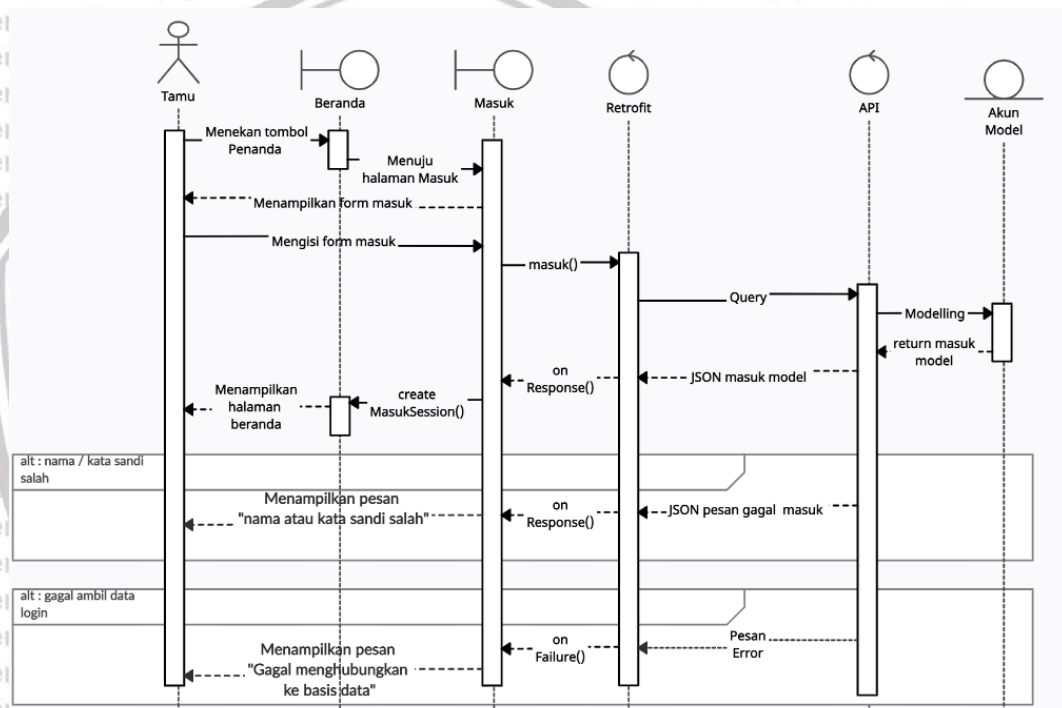
Gambar 5.4 Sequence Diagram Lihat Denah Objek Layanan Kesehatan

Sequence diagram Lihat Denah Objek Layanan Kesehatan yang dapat dilihat Dalam Gambar 5.4 memiliki 1 aktor yaitu Pengguna. Pada *sequence* diagram ini terjadi interaksi aktor menekan tombol Denah Lokasi pada halaman beranda, selanjutnya halaman berpindah dari beranda menuju halaman Denah Lokasi, kemudian sistem menyediakan pilihan kategori objek layanan kesehatan apa saja yang ingin ditampilkan, setelah aktor memilih kategori, dipanggil fungsi `getListLayananKesehatan()` untuk meminta *Controller* Retrofit membuat

Query yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit akan mengirim objek yang ditangkap halaman Daftar Spesialis pada fungsi `onResponse()` dan kemudian daftar layanan kesehatan akan ditampilkan pada denah berupa marker yang titiknya sesuai dengan lokasi tempat layanan kesehatan tersebut berada. Apabila daftar objek layanan kesehatan gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar spesialis pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor. Selanjutnya aktor memilih marker lalu sistem akan menampilkan nama objek layanan kesehatan, tombol Detail Informasi dan tombol Navigasi.

5.2.2 Iterasi 1

5.2.2.1 Masuk Akun

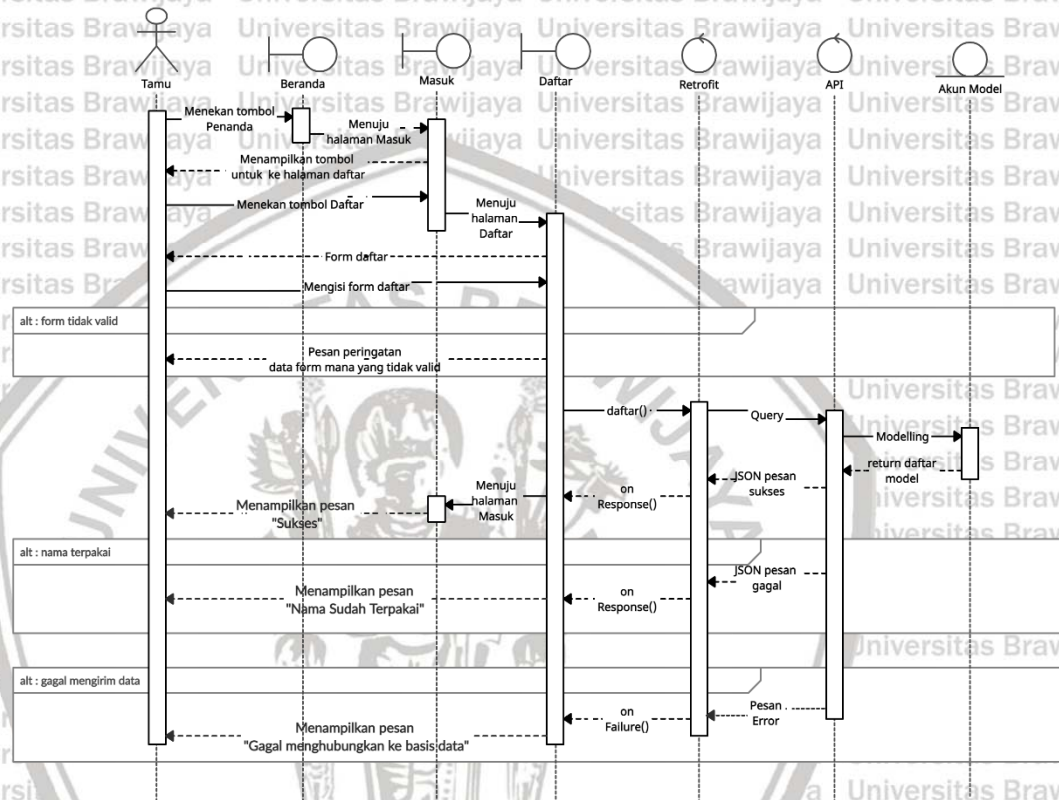


Gambar 5.5 Sequence Diagram Masuk Akun

Sequence diagram Masuk Akun yang dapat dilihat Dalam Gambar 5.5 memiliki 1 aktor yaitu Tamu. Pada sequence diagram ini terjadi interaksi Tamu menekan tombol Penanda pada halaman Beranda, selanjutnya halaman berpindah dari beranda menuju halaman Masuk. kemudian sistem menampilkan form untuk masuk lalu Tamu mengisi form tersebut. Selanjutnya dipanggil fungsi `masuk()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk dilakukan pengecekan akun dan mengembalikan data berupa model JSON kepada Retrofit. selanjutnya retrofit akan mengirim objek yang ditangkap halaman Masuk pada fungsi `onResponse()` dan konfirmasi status masuk akan ditampilkan. Apabila data form masuk salah maka akan memasuki alur alternatif dimana API akan mengirimkan pesan gagal masuk yang akan ditangkap halaman

Masuk pada fungsi `onResponse()` dan pesan akan ditampilkan kepada Tamu. Apabila daftar objek layanan kesehatan gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar spesialis pada fungsi `onFailure()` dan pesan akan ditampilkan kepada Tamu.

5.2.2.2 Daftar Akun

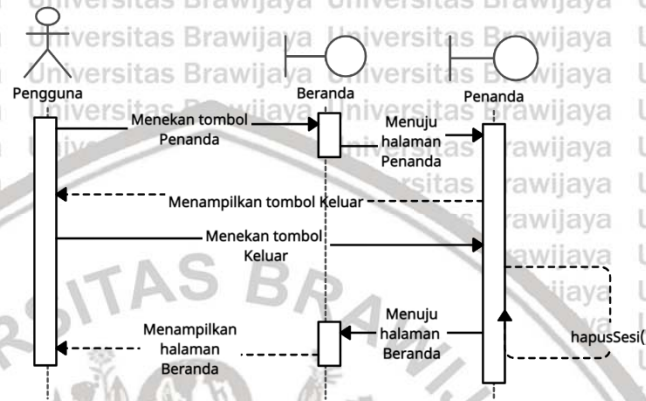


Gambar 5.6 Sequence Diagram Daftar Akun

Sequence diagram Daftar Akun yang dapat dilihat Dalam Gambar 5.6 memiliki 1 aktor yaitu Tamu. Pada *sequence* diagram ini terjadi interaksi Tamu menekan tombol Penanda pada halaman Beranda, selanjutnya halaman berpindah dari beranda menuju halaman Masuk. kemudian sistem menampilkan tombol Daftar lalu Tamu menekan tombol tersebut. Selanjutnya halaman berpindah ke halaman Daftar, pada halaman daftar ditampilkan form untuk mendaftarkan, Tamu mengisi form tersebut. Apabila karakter yang diisikan pada form tidak valid, maka sistem memasuki alur alternatif dimana dilakukan validasi pada perangkat dan akan ditampilkan pesan peringatan tentang data form yang tidak valid. Selanjutnya apabila data sudah valid akan dipanggil fungsi `daftar()` untuk meminta *Controller Retrofit* membuat *Query* yang dikirimkan ke API untuk dilakukan input data ke dalam basis data. Kemudian API akan mengembalikan pesan sukses yang akan ditangkap pada fungsi `onResponse()`.

Apabila nama yang diisikan pada form Daftar sudah terdaftar pada basis data, maka sistem akan memasuki alur alternatif dimana API akan mengirimkan pesan “Nama Sudah Terpakai” yang akan ditangkap halaman Daftar pada fungsi `onResponse()` dan pesan akan ditampilkan kepada Tamu. Apabila sistem gagal menghubungkan ke jaringan, maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar spesialis pada fungsi `onFailure()` dan pesan akan ditampilkan kepada Tamu.

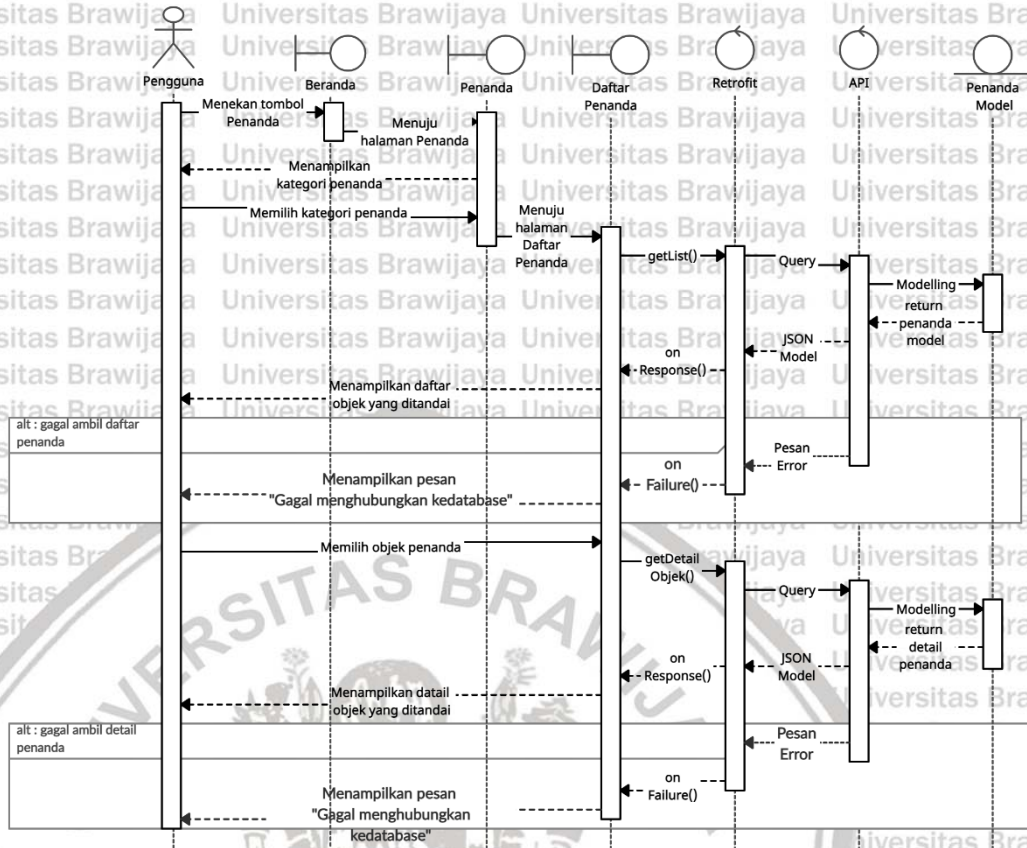
5.2.2.3 Keluar Akun



Gambar 5.7 Sequence Diagram Keluar Akun

Sequence diagram Keluar Akun yang dapat dilihat Dalam Gambar 5.7 memiliki 1 aktor yaitu Pengguna. Pada *sequence* diagram ini terjadi interaksi pengguna menekan tombol Penanda pada halaman Beranda, selanjutnya halaman berpindah dari beranda menuju halaman Penanda. kemudian sistem menampilkan tombol Keluar, lalu pengguna menekan tombol tersebut. Selanjutnya dipanggil fungsi `hapusSesi()` untuk menghapus sesi yang tersimpan pada perangkat dan halaman akan berpindah ke Beranda.

5.2.2.4 Penanda



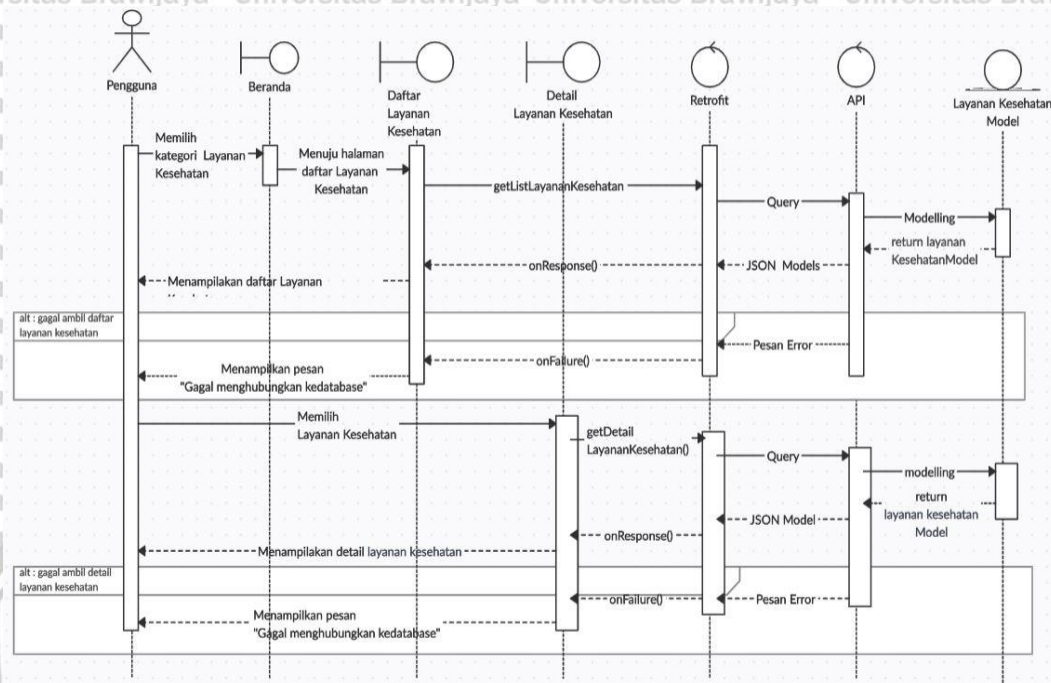
Gambar 5.8 Sequence Diagram Penanda

Sequence diagram Penanda yang dapat dilihat Dalam Gambar 5.8 memiliki 1 aktor yaitu Pengguna. Pada *sequence* diagram ini terjadi interaksi pengguna menekan tombol Penanda pada halaman Beranda. Selanjutnya halaman berpindah dari beranda menuju halaman Penanda, kemudian sistem menampilkan kategori penanda dan Pengguna memilih kategori. Setelah memilih kategori halaman akan berpindah menuju halaman Daftar Penanda yang berisikan daftar objek yang ditandai. Pada halaman Daftar Penanda dipanggil fungsi `getList()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya Retrofit akan mengirim objek yang ditangkap pada fungsi `onResponse()`, kemudian daftar objek yang ditandai akan ditampilkan. Apabila daftar gagal diambil maka sistem akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap pada fungsi `onFailure()` dan pesan akan ditampilkan kepada pengguna.

Setelah Daftar objek yang ditandai berhasil ditampilkan, Pengguna memilih salah satu objek dan fungsi `getDetailObjek()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya Retrofit akan mengirim objek yang ditangkap pada fungsi `onResponse()`, kemudian detail objek yang ditandai akan

ditampilkan. Apabila detail objek gagal diambil maka sistem akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap pada fungsi `onFailure()` dan pesan akan ditampilkan kepada pengguna.

5.2.2.5 Lihat Informasi Objek Layanan Kesehatan



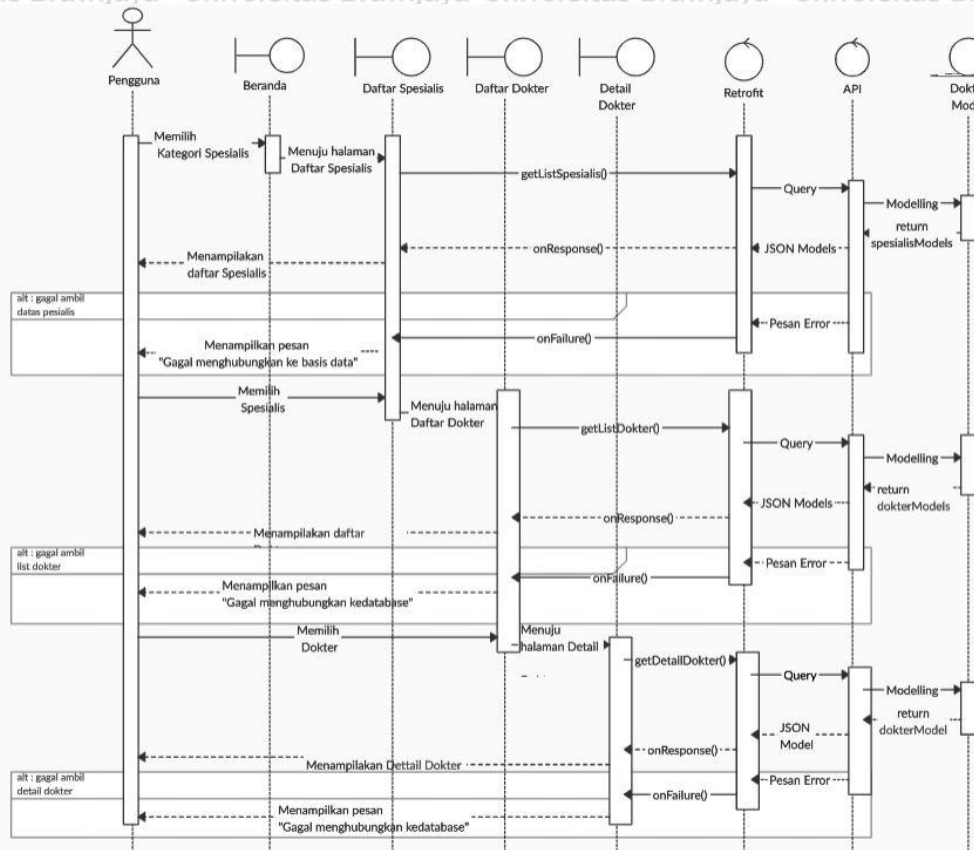
Gambar 5.9 Sequence Diagram Lihat Informasi Objek Layanan Kesehatan

Sequence diagram Lihat Informasi Objek Layanan Kesehatan yang dapat dilihat Dalam Gambar 5.9 memiliki 2 aktor yaitu Tamu dan Pengguna. Pada *sequence* diagram ini terjadi interaksi aktor memilih kategori layanan kesehatan pada halaman beranda, selanjutnya halaman berpindah dari beranda menuju halaman Daftar Layanan Kesehatan sesuai dengan kategori yang dipilih, kemudian dipanggil fungsi `getListLayananKesehatan()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit akan mengirim objek yang ditangkap halaman Daftar Layanan Kesehatan kesehatan pada fungsi `onResponse()` dan kemudian daftar layanan kesehatan akan ditampilkan. Apabila daftar objek layanan kesehatan gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar layanan kesehatan pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

Setelah daftar ditampilkan, aktor memilih objek layanan kesehatan dan halaman akan berpindah ke halaman Detail Layanan Kesehatan. Pada halaman Detail Layanan Kesehatan dipanggil fungsi `getDetailLayananKesehatan()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit akan mengirim objek yang ditangkap halaman Daftar Layanan Kesehatan kesehatan pada fungsi `onResponse()` dan kemudian detail layanan kesehatan akan ditampilkan. Apabila detail objek layanan kesehatan gagal diambil maka akan

memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar layanan kesehatan pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

5.2.2.6 Lihat Informasi Dokter



Gambar 5.10 Sequence Diagram Lihat Informasi Dokter

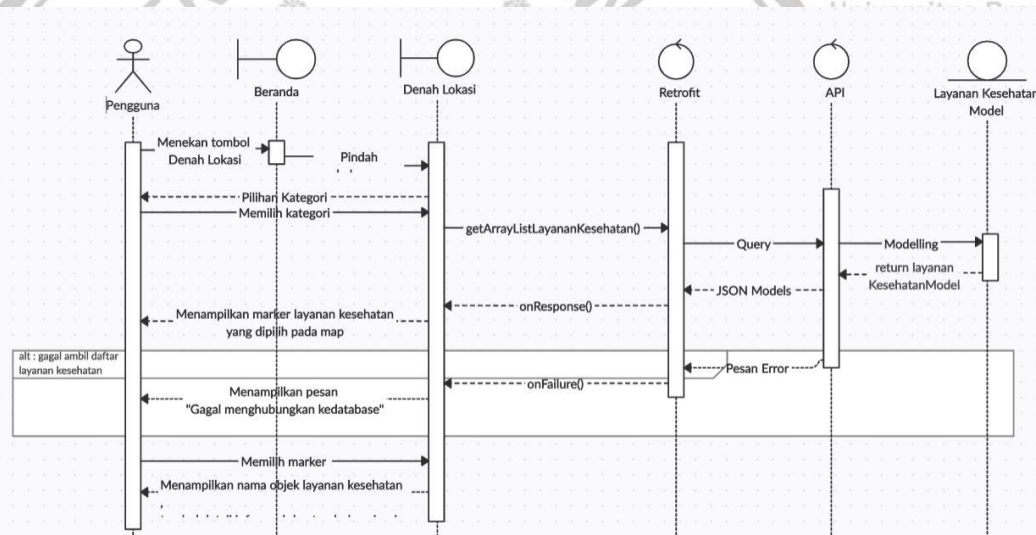
Sequence diagram Lihat Informasi Dokter yang dapat dilihat Dalam Gambar 5.10 memiliki 2 aktor yaitu Tamu dan Pengguna. Pada *sequence* diagram ini terjadi interaksi aktor memilih kategori Spesialis pada halaman beranda , selanjutnya halaman berpindah dari beranda menuju halaman Daftar Spesialis, kemudian dipanggil fungsi `getListSpesialis()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit akan mengirim objek yang ditangkap halaman Daftar Spesialis pada fungsi `onResponse()` dan kemudian daftar layanan kesehatan akan ditampilkan. Apabila daftar objek layanan kesehatan gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar spesialis pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

Selanjutnya aktor memilih spesialisasi pada halaman Daftar Spesialis , kemudian halaman berpindah dari Daftar Spesialis menuju halaman Daftar Dokter, selanjutnya dipanggil fungsi `getListDokter()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model

JSON kepada Retrofit, kemudian retrofit akan mengirim objek yang ditangkap halaman Daftar Dokter pada fungsi `onResponse()` lalu daftar layanan kesehatan akan ditampilkan. Apabila daftar dokter gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar layanan kesehatan pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

Setelah daftar dokter ditampilkan, aktor memilih objek dokter dan halaman akan berpindah ke halaman Detail Dokter. Pada halaman Detail Dokter dipanggil fungsi `getDetailDokter()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit akan mengirim objek yang ditangkap halaman Daftar Layanan Kesehatan kesehatan pada fungsi `onResponse()` dan kemudian detail dokter akan ditampilkan. Apabila detail dokter gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar layanan kesehatan pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor.

5.2.2.7 Lihat Denah Objek Layanan Kesehatan



Gambar 5.11 Sequence Diagram Lihat Denah Objek Layanan Kesehatan

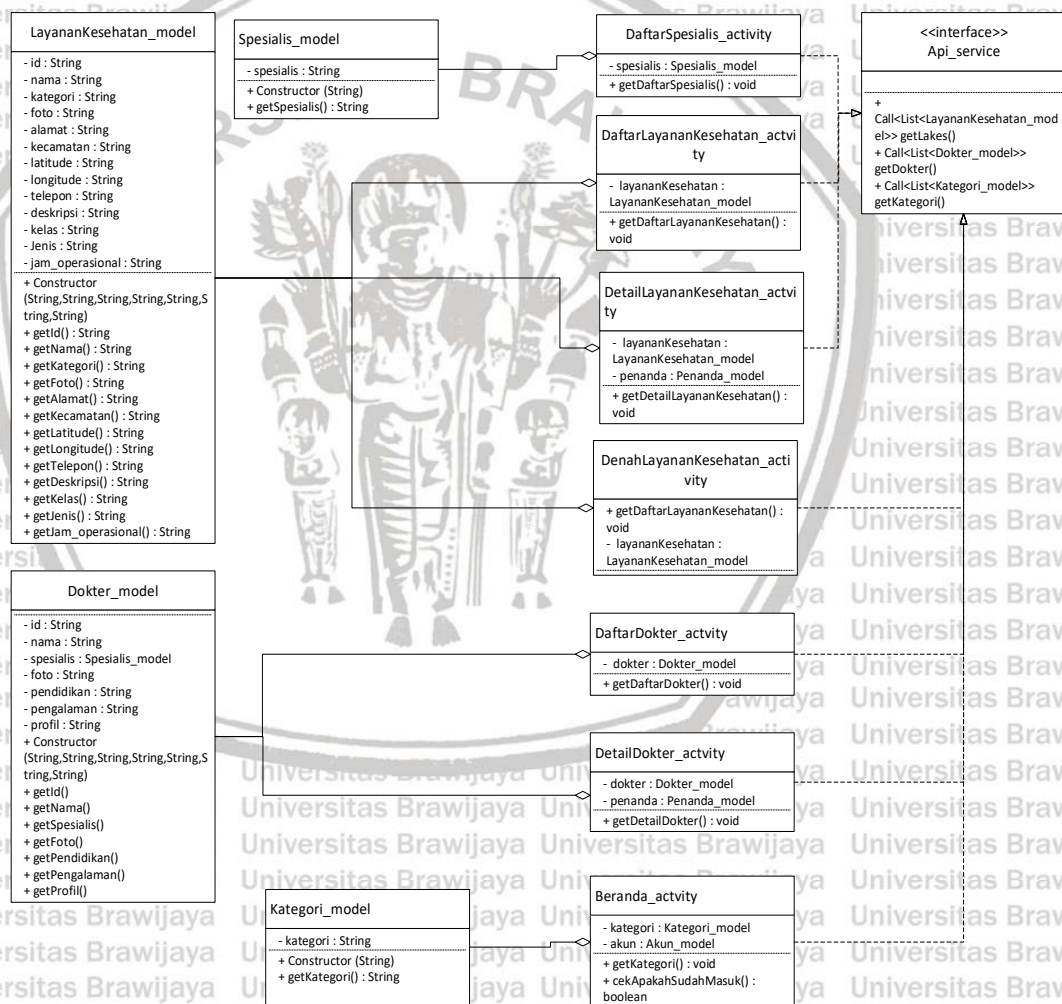
Sequence diagram Lihat Denah Objek Layanan Kesehatan yang dapat dilihat Dalam Gambar 5.11 memiliki 2 aktor yaitu Tamu dan Pengguna. Pada *sequence* diagram ini terjadi interaksi aktor menekan tombol Denah Lokasi pada halaman beranda, selanjutnya halaman berpindah dari beranda menuju halaman Denah Lokasi, kemudian sistem menyediakan pilihan kategori objek layanan kesehatan apa saja yang ingin ditampilkan, setelah aktor memilih kategori, dipanggil fungsi `getListLayananKesehatan()` untuk meminta *Controller* Retrofit membuat *Query* yang dikirimkan ke API untuk mengembalikan data berupa model JSON kepada Retrofit, selanjutnya retrofit akan mengirim objek yang ditangkap halaman Daftar Spesialis pada fungsi `onResponse()` dan kemudian daftar layanan kesehatan akan ditampilkan pada denah berupa marker yang titiknya

sesuai dengan lokasi tempat layanan kesehatan tersebut berada. Apabila daftar objek layanan kesehatan gagal diambil maka akan memasuki alur alternatif dimana API akan mengirimkan pesan *error* yang akan ditangkap halaman daftar spesialis pada fungsi `onFailure()` dan pesan akan ditampilkan kepada aktor. Selanjutnya aktor memilih marker lalu sistem akan menampilkan nama objek layanan kesehatan, tombol Detail Informasi dan tombol Navigasi.

5.3 Perancangan Class Diagram

Class diagram berisikan hubungan antar *class* yang ada. *Class* diagram ini dijadikan pedoman dalam mengimplementasikan *class* dalam bahasa pemrograman pada tahap implementasi.

5.3.1 Iterasi 0

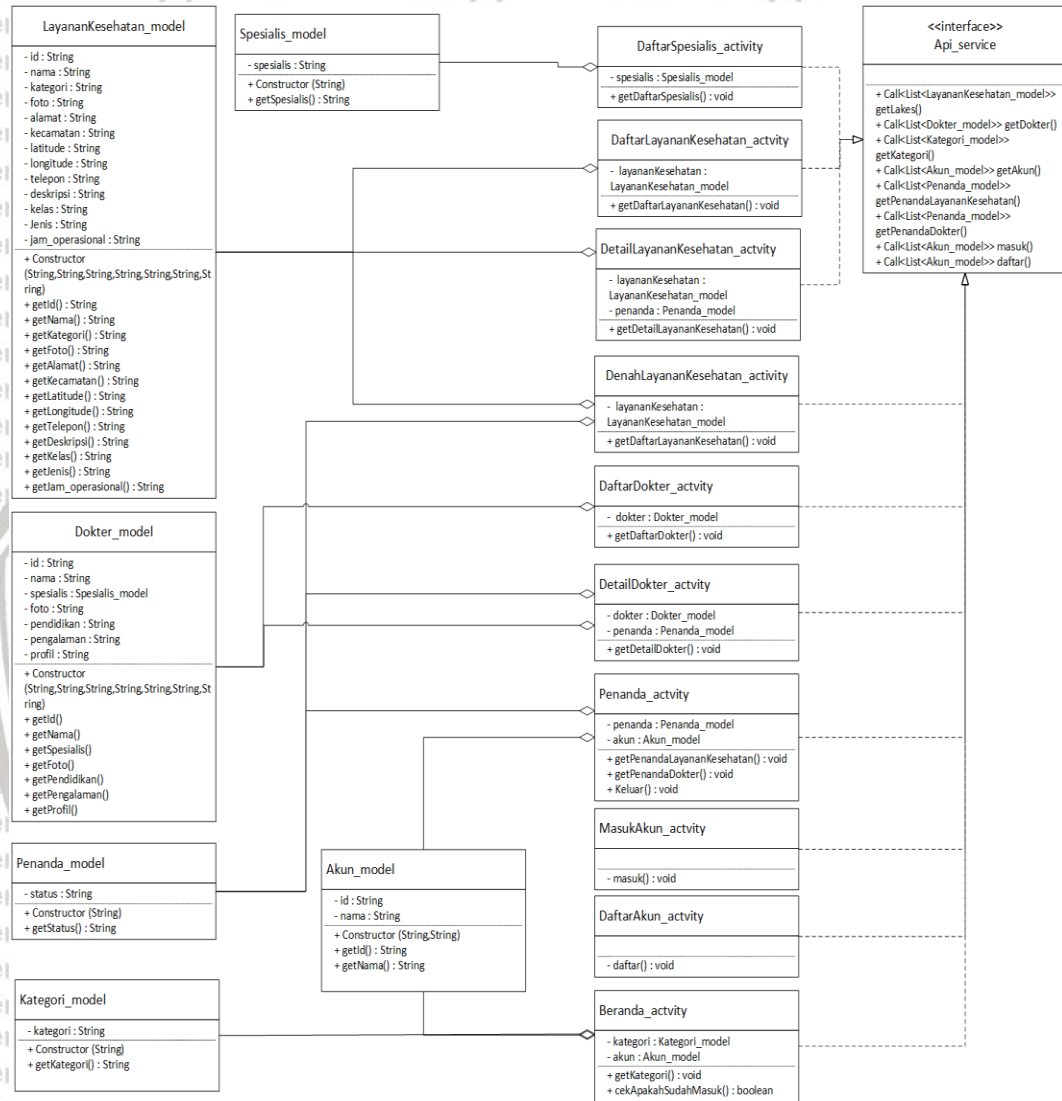


Gambar 5.12 Perancangan Class Diagram Iterasi 0

Class diagram iterasi 0 Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat dapat dilihat dalam Gambar 5.12. Pada *class* diagram ini terdapat 4 *class model*, 7 *class activity* dan 1 *interface*. *Class model* nantinya akan yang

memiliki relasi *Aggregation* dengan *class activity*, hal ini berarti bahwa *class model* merupakan bagian dari *class activity*. Selanjutnya *class activity* memiliki relasi *Realization* dengan *class interface* yang berarti *class activity* mengimplementasikan fungsi yang ada pada *class interface*.

5.3.2 Iterasi 1



Gambar 5.13 Perancangan Class Diagram Iterasi 1

Class diagram iterasi 1 Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat dapat dilihat dalam Gambar 5.13. Pada *class* diagram iterasi 1 ini terdapat penambahan 2 *class model* yaitu *Akun_model* dan *Penanda_model*. Pada diagram ini *class Model* memiliki relasi *Aggregation* dengan *class activity*, hal ini berarti bahwa *class model* merupakan bagian dari *class activity*. Selanjutnya *class activity* memiliki relasi *Realization* dengan *class interface* yang berarti *class activity* mengimplementasikan fungsi yang ada pada *class interface*.

5.4 Perancangan Tabel Basis Data

Perancangan tabel basis data berisikan rancangan dari struktur tabel pada basis data yang nantinya akan diimplementasikan pada sistem. Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat nantinya akan menggunakan basis data berbasis SQL. Dalam perancangan basis data ini dijabarkan tentang nama tabel, nama atribut dan deskripsi dari atribut.

5.4.1 Iterasi 0

Pada perancangan basis data iterasi 0, terdapat 10 buah rancangan, yaitu rancangan tabel dokter, rancangan tabel objek layanan kesehatan, rancangan tabel spesialis, rancangan tabel kategori, rancangan tabel fasilitas, rancangan tabel dukungan_pembayaran, rancangan tabel jenis, rancangan tabel praktik dokter, rancangan tabel dukungan_pembayaran_objek_layanan_kesehatan, dan rancangan tabel jenis_objek_layanan_kesehatan. Rancangan-rancangan tersebut dapat dilihat pada Tabel 5.1 – Tabel 5.10.

Tabel 5.1 Rancangan Tabel Dokter

No	Atribut	Deskripsi
1	Id	Angka unik yang menjadi <i>primary key</i> dari dokter
2	nama	Nama lengkap dokter beserta gelar
3	spesialis	<i>Foreign key</i> dari dari spesialis
4	pendidikan	Riwayat pendidikan yang pernah ditempuh oleh dokter
5	pengalaman	Riwayat pengalaman bekerja dokter
6	profil	Deskripsi personal dokter

Tabel 5.2 Rancangan Tabel Objek_Layanan_Kesehatan

No	Atribut	Deskripsi
1	Id	Angka unik yang menjadi <i>primary key</i> dari Objek Layanan Kesehatan
2	nama	Nama objek layanan lesehatan
3	foto	Foto dari objek layanan kesehatan
4	kategori	<i>Foreign key</i> dari dari kategori
5	Jenis	<i>Foreign key</i> dari dari jenis
6	alamat	Alamat objek layanan kesehatan
7	telepon	Telepon objek layanan kesehatan

Tabel 5.2 Rancangan Tabel Objek_Layanan_Kesehatan (lanjutan)

No	Atribut	Deskripsi
8	latitude	Garis lintang posisi objek layanan kesehatan
9	longitude	Garis bujur posisi objek layanan kesehatan
10	deskripsi	Nararsi yang mendeskripsikan objek layanan kesehatan
11	jam_operasional	Informasi jam operasional dari objek layanan kesehatan

Tabel 5.3 Rancangan Tabel Spesialis

No	Atribut	Deskripsi
1	spesialis	Nama dari spesialisasi dokter

Tabel 5.4 Rancangan Tabel Kategori

No	Atribut	Deskripsi
1	kategori	Nama dari kategori objek layanan kesehatan

Tabel 5.5 Rancangan Tabel Fasilitas

No	Atribut	Deskripsi
1	fasilitas	Nama dari fasilitas pada objek layanan kesehatan

Tabel 5.6 Rancangan Tabel Dukungan_Pembayaran

No	Atribut	Deskripsi
1	dukungan_pembayaran	Nama dari dukungan pembayaran

Tabel 5.7 Rancangan Tabel Jenis

No	Atribut	Deskripsi
1	jenis	Nama dari tingkatan kelas pada objek layanan kesehatan

Tabel 5.8 Rancangan Tabel Praktik_Dokter

No	Atribut	Deskripsi
1	Id	Angka unik yang menjadi <i>primary key</i> praktek dokter

Tabel 5.8 Rancangan Tabel Praktik_Dokter (lanjutan)

No	Atribut	Deskripsi
2	id_dokter	Foreign key dari dari dokter
3	id_lakes	Foreign key dari dari objek layanan kesehatan
4	senin	Jadwal praktek hari senin
5	selasa	Jadwal praktek hari selasa
6	rabu	Jadwal praktek hari rabu
7	kamis	Jadwal praktek hari kamis
8	jumat	Jadwal praktek hari jumat
9	sabtu	Jadwal praktek hari sabtu
10	minggu	Jadwal praktek hari minggu

Tabel 5.9 Rancangan Tabel Dukungan Pembayaran_Objek_Layanan_Kesehatan

No	Atribut	Deskripsi
1	Id	Angka unik yang menjadi <i>primary key</i> Dukungan Pembayaran objek layanan kesehatan
2	id_lakes	Foreign key dari objek layanan kesehatan
3	dukungan_pembayaran	Foreign key dari dukungan pembayaran

Tabel 5.10 Rancangan Tabel Jenis_Objek_Layanan_Kesehatan

No	Atribut	Deskripsi
1	Id	Angka unik yang menjadi <i>primary key</i> Jenis Objek Layanan Kesehatan
2	id_lakes	Foreign key dari objek layanan kesehatan
3	jenis	Foreign key dari jenis

5.4.2 Iterasi 1

Pada perancangan tabel basis data iterasi 1, tidak terjadi perubahan pada perancangan iterasi sebelumnya, akan tetapi terdapat penambahan 3 rancangan, yaitu rancangan tabel akun, rancangan tabel penanda_dokter dan rancangan tabel penanda_objek_layanan_kesehatan. Rancangan-rancangan ini dapat dilihat pada Tabel 5.11 – Tabel 5.13.

Tabel 5.11 Rancangan Tabel Akun

No	Atribut	Deskripsi
1	Id	Angka unik yang menjadi <i>primary key</i> akun
2	nama	Nama pemilik akun
3	kata_sandi	Kata sandi untuk masuk ke dalam akun

Tabel 5.12 Rancangan Tabel_Penanda_Dokter

No	Atribut	Deskripsi
1	Id	Angka unik yang menjadi <i>primary key</i> akun
2	id_akun	<i>Foreign key</i> dari akun
3	id_dokter	<i>Foreign key</i> dari dokter

Tabel 5.13 Rancangan Tabel Penanda_Objek_Layanan_Kesehatan

No	Atribut	Deskripsi
1	Id	Angka unik yang menjadi <i>primary key</i> akun
2	id_akun	<i>Foreign key</i> dari akun
3	id_lakes	<i>Foreign key</i> dari Objek Layanan Kesehatan

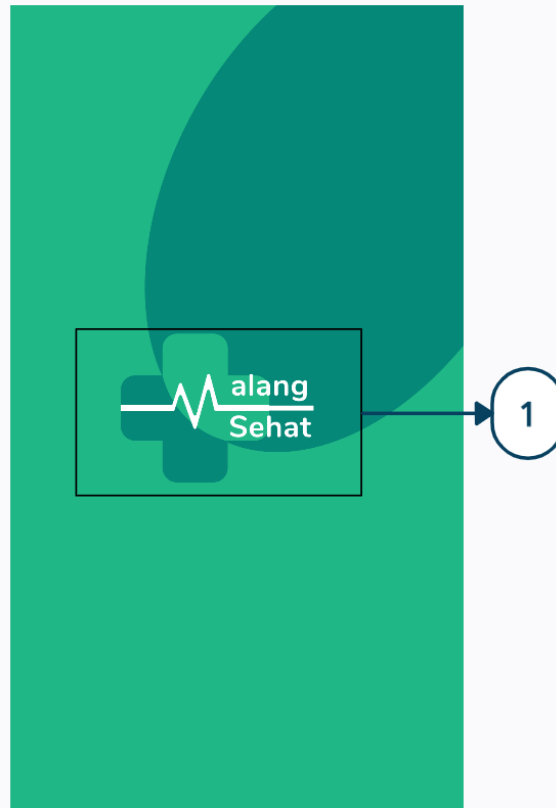
5.5 Perancangan Antarmuka

Perancangan antarmuka adalah sketsa dari tiap-tiap halaman yang nantinya akan diterapkan pada sistem. Penggambaran perancangan antarmuka pada aplikasi Malang Sehat dibuat dengan menggunakan aplikasi web bernama Figma. Setelah perancangan antarmuka selesai dibuat, semua aset yang ada pada aplikasi Figma akan di ekspor dan kemudian akan di impor ke dalam aplikasi Android Studio.

5.5.1 Iterasi 0

5.5.1.1 Rancangan Antarmuka Halaman *Splash Screen*

Splash screen merupakan halaman pertaman yang muncul apabila aplikasi dijalankan. Penjelasan setiap element halaman *Splash Screen* yang ada dalam Gambar 5.14 terletak pada Tabel 5.14.



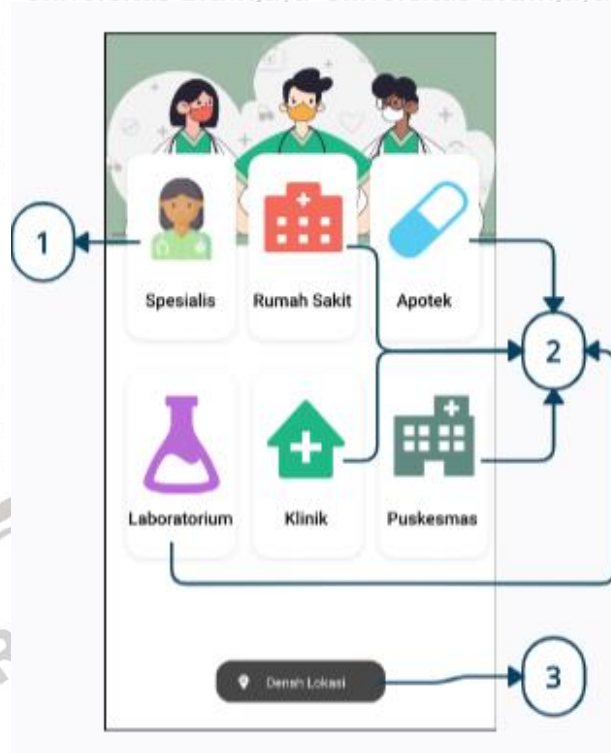
Gambar 5.14 Rancangan Antarmuka Halaman *Splash Screen*

Tabel 5.14 Informasi Element Rancangan Antarmuka Halaman *Splash Screen*

No	Penjelasan	Detail Fungsi
1	Lambang aplikasi Malang Sehat	Untuk memperjelas aplikasi yang sedang dibuka pengguna

5.5.1.2 Rancangan Antarmuka Halaman Beranda

Beranda merupakan halaman yang muncul setelah halaman *Splash Screen*. Pada halaman ini terdapat beberapa menu berupa tombol. Penjelasan Setiap element antarmuka halaman Beranda yang ada dalam Gambar 5.16 terletak pada Tabel 5.15 berikut :



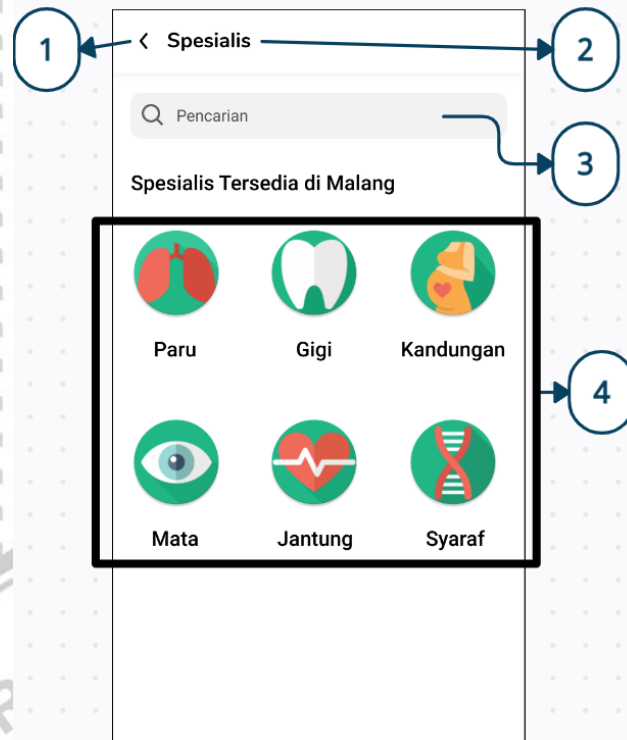
Gambar 5.15 Rancangan Antarmuka Halaman Beranda

Tabel 5.15 Informasi Element Rancangan Antarmuka Halaman Beranda

No	Penjelasan	Detail Fungsi
1	Tombol Spesialis	Sebagai pemicu untuk dilakukanya perpindahan ke halaman Daftar Spesialis
2	Tombol kategori objek layanan kesehatan	Sebagai pemicu untuk dilakukanya perpindahan ke halaman Daftar Layanan Kesehatan
3	Tombol Denah Lokasi	Sebagai pemicu untuk dilakukanya perpindahan ke halaman Denah Lokasi

5.5.1.3 Rancangan Antarmuka Halaman Daftar Spesialis

Halaman Daftar Spesialis berisikan pilihan menu spesialisasi yang tersedia. Penjelasan Setiap element antarmuka halaman Daftar Spesialis yang ada dalam Gambar 5.16 terletak pada Tabel 5.16 berikut :



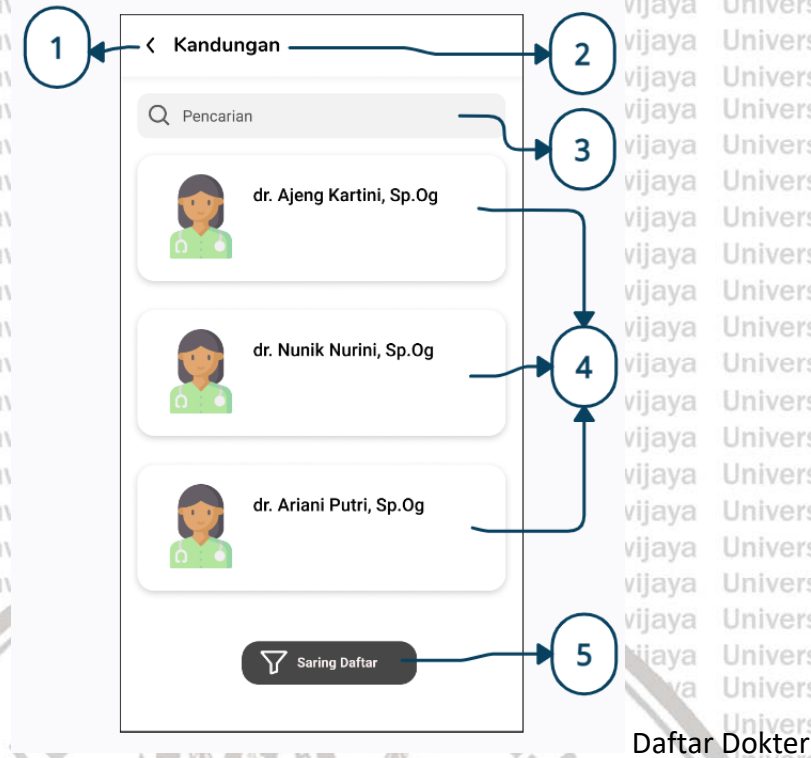
Gambar 5.16 Rancangan Antarmuka Halaman Daftar Spesialis

Tabel 5.16 Informasi Element Rancangan Antarmuka Halaman Daftar Spesialis

No	Penjelasan	Detail Fungsi
1	Tombol kembali	Sebagai pemicu untuk dilakukanya perpindahan ke halaman sebelumnya
2	Nama Kategori	Untuk memperjelas kategori menu yang sedang dibuka pengguna
3	Kolom pencarian	Untuk melakukan pencarian
4	Tombol pilihan spesialis yang tersedia	Sebagai pemicu untuk dilakukanya perpindahan ke halaman daftar dokter dari spesialisasi yang dipilih

5.5.1.4 Rancangan Antarmuka Halaman Daftar Dokter

Halaman Daftar Dokter berisikan pilihan daftar dokter yang ada. Daftar dokter ini direpresentasikan dngn tampilan kartu. Penjelasan Setiap element antarmuka halaman Daftar dokter yang ada dalam Gambar 5.17 terletak pada Tabel 5.17 berikut :



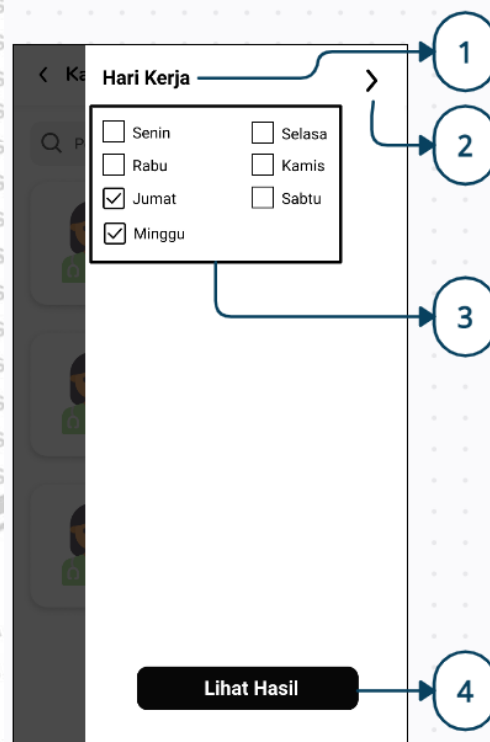
Gambar 5.17 Rancangan Antarmuka Halaman

Tabel 5.17 Informasi Element Rancangan Antarmuka Halaman Daftar Dokter

No	Penjelasan	Detail Fungsi
1	Tombol kembali	Sebagai pemicu untuk dilakukanya perpindahan ke halaman sebelumnya
2	Nama Spesialisasi	Untuk memperjelas spesialisasi dari daftar dokter yang sedang dibuka pengguna
3	Kolom pencarian	Untuk melakukan pencarian
4	Objek kartu dokter	Sebagai representasi dari tiap-tiap dokter yang diwakili dengan foto dan nama dokter tersebut, apabila objek di tekan maka halaman akan berpindah ke halaman informasi detail dokter
5	Tombol Saring Daftar	Sebagai pemicu untuk ditampilkanya <i>pop-up</i> penyaringan

5.5.1.5 Rancangan Antarmuka *Pop-up Saring Daftar Dokter*

Pop-up Saring Daftar Dokter berisikan pilihan menu untuk melakukan penyaringan pada daftar dokter yang ada. Penjelasan Setiap element antarmuka *Saring Daftar Dokter* yang ada dalam Gambar 5.18 terletak pada Tabel 5.18.



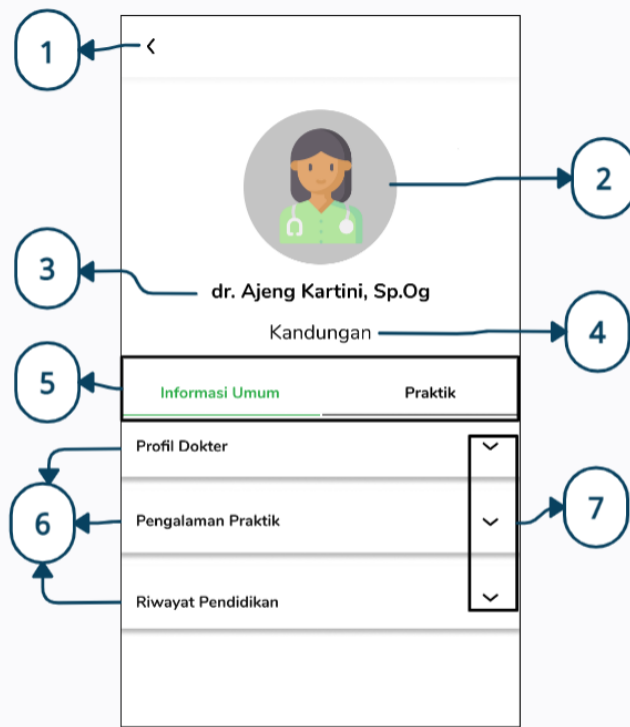
Gambar 5.18 Rancangan Antarmuka *Pop-up* Saring Daftar Dokter

Tabel 5.18 Informasi Element Rancangan Antarmuka Saring Daftar Dokter

No	Penjelasan	Detail Fungsi
1	Nama Kategori Penyaringan	Untuk memperjelas kategori yang menjadi dasar dari penyaringan
2	Tombol kembali	Sebagai pemicu untuk ditutupnya antarmuka saring
3	Checkbox	Sebagai antarmuka pengguna dalam memilih, ketika <i>checkbox</i> ditekan maka tanda centang akan ditambahkan
4	Tombol Lihat Hasil	Sebagai pemicu untuk menyaring daftar berdasarkan masukan pengguna

5.5.1.6 Rancangan Antarmuka Halaman Detail Dokter

Halaman Detail Dokter berisikan informasi detail dari dokter. Penjelasan setiap element antarmuka halaman Detail Dokter yang ada dalam Gambar 5.19 terletak pada Tabel 5.19.



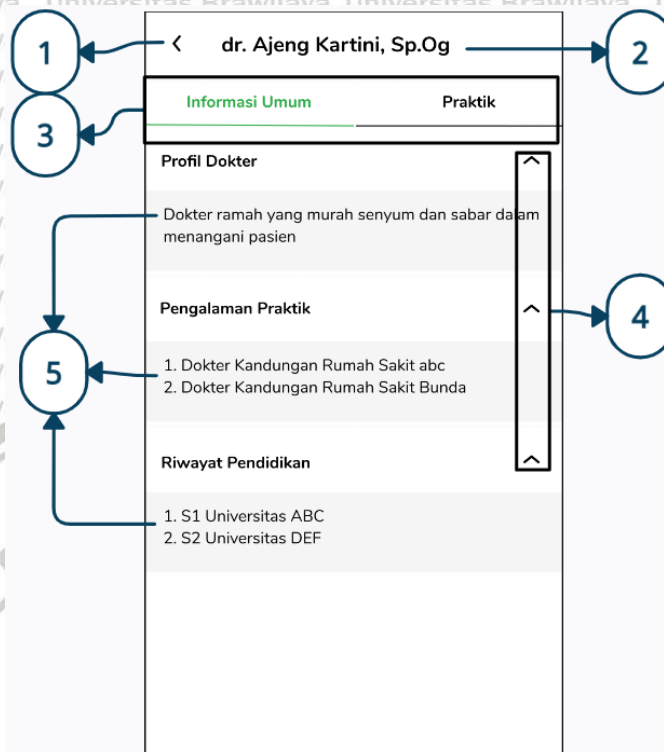
Gambar 5.19 Rancangan Antarmuka Halaman Detail Dokter

Tabel 5.19 Informasi Element Rancangan Antarmuka Halaman Detail Dokter

No	Penjelasan	Detail Fungsi
1	Tombol Kembali	Sebagai pemicu untuk dilakukanya perpindahan ke halaman sebelumnya
2	Foto	Sebagai media untuk memperjelas informasi tentang dokter
3	Nama	Untuk memberikan informasi nama dokter
4	Spesialisasi	Untuk memberikan informasi spesialisasi dokter
5	Tab layout	Sebagai pilihan menu kategori untuk informasi rinci dari dokter
6	Informasi umum dokter	Untuk memberikan informasi umum berupa profil dokter, pengalaman praktik dan riwayat pendidikan
7	Tombol <i>Dropdown</i>	Untuk membuka atau menutup kolom isi pada informasi rinci

5.5.1.7 Rancangan Antarmuka Halaman Detail Dokter Tab Informasi Umum

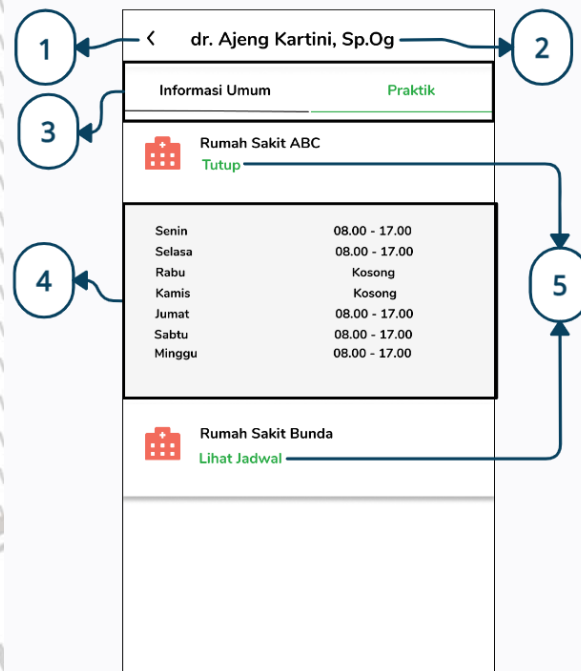
Penjelasan setiap element antarmuka halaman Detail Dokter Tab Informasi Umum yang ada dalam Gambar 5.20 terletak pada Tabel 5.20 berikut



Gambar 5.20 Rancangan Antarmuka Halaman Detail Dokter Tab Informasi Umum

Tabel 5.20 Informasi Element Rancangan Antarmuka Halaman Detail Dokter Tab Informasi Umum

No	Penjelasan	Detail Fungsi
1	Tombol Kembali	Sebagai pemicu untuk dilakukanya perpindahan ke halaman sebelumnya
2	Nama	Untuk memberikan informasi nama dokter
3	Tab layout	Sebagai pilihan menu kategori untuk informasi rinci dari dokter
4	Tombol <i>Dropdown</i>	Untuk membuka atau menutup kolom isi pada informasi rinci
5	Informasi rinci	Untuk memberikan informasi rinci berupa profil dokter, pengalaman praktik dan riwayat pendidikan

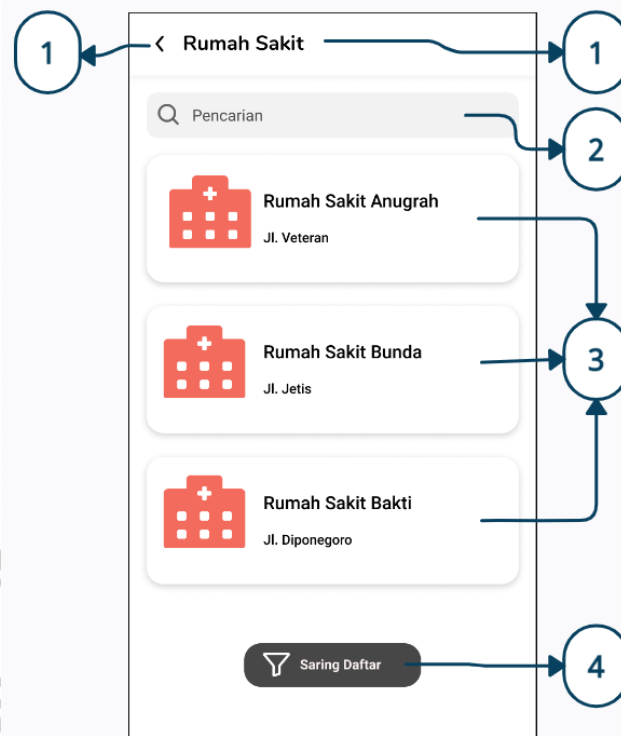


Gambar 5.21 Rancangan Antarmuka Halaman Detail Dokter Tab Praktik

Penjelasan setiap element antarmuka halaman Detail Dokter Tab Praktik yang ada dalam Gambar 5.21 terletak pada Tabel 5.21 berikut :

Tabel 5.21 Informasi Element Rancangan Antarmuka Halaman Detail Dokter Bagian Tab Praktik

No	Penjelasan	Detail Fungsi
1	Tombol Kembali	Sebagai pemicu untuk dilakukanya perpindahan ke halaman sebelumnya
2	Nama	Untuk memberikan informasi nama dokter
3	Tab layout	Sebagai pilihan menu kategori untuk informasi rinci dari dokter
4	Kolom isi informasi praktik	Sebagai penampil informasi jadwal praktik dokter
5	Tombol Lihat Jadwal	Untuk membuka atau menutup kolom isi informasi praktik

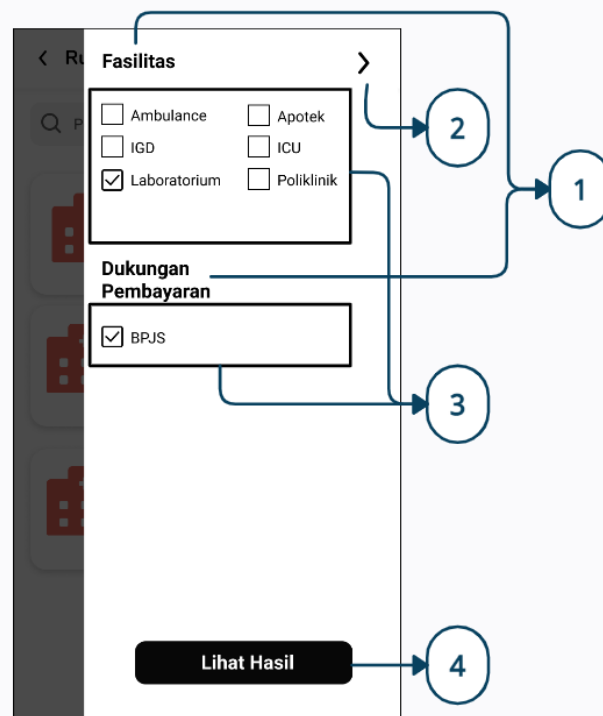


Gambar 5.22 Rancangan Antarmuka Halaman Daftar Objek Layanan Kesehatan

Penjelasan setiap element antarmuka halaman Daftar Objek Layanan Kesehatan yang ada dalam Gambar 5.22 terletak pada Tabel 5.22 berikut :

Tabel 5.22 Informasi Element Rancangan Antarmuka Halaman Daftar Objek Layanan Kesehatan

No	Penjelasan	Detail Fungsi
1	Tombol kembali	Sebagai pemicu untuk dilakukanya perpindahan ke halaman sebelumnya
2	Nama Kategori	Untuk memperjelas kategori dari daftar dokter yang sedang dibuka pengguna
3	Kolom pencarian	Untuk melakukan pencarian
4	Objek kartu layanan kesehatan	Sebagai representasi dari tiap-tiap objek layanan kesehatan yang diwakili dengan foto dan nama layanan kesehatan tersebut, apabila objek di tekan maka halaman akan berpindah ke halaman Detail Objek Layanan Kesehatan
5	Tombol Saring Daftar	Sebagai pemicu untuk ditampilkannya <i>pop-up</i> penyaringan

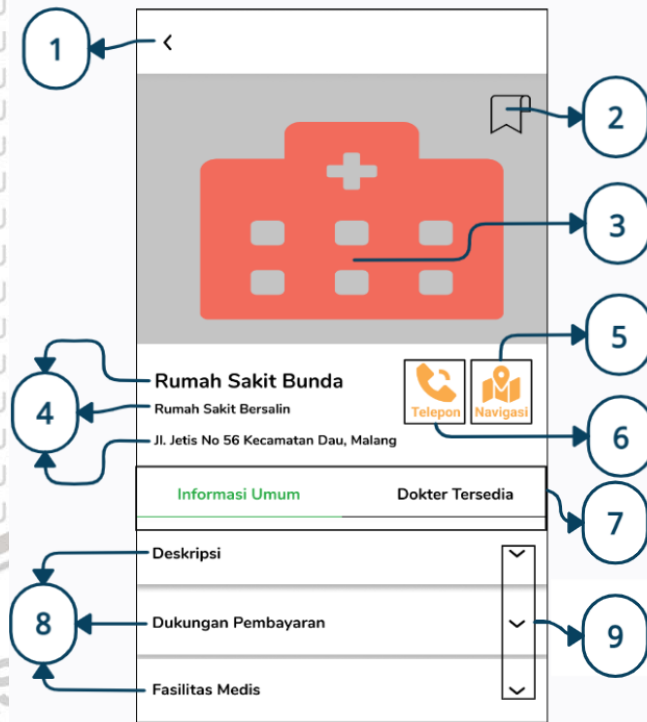


Gambar 5.23 Rancangan Antarmuka *Pop-up* Saring Objek Layanan Kesehatan

Penjelasan setiap element antarmuka Saring Daftar Objek Layanan Kesehatan yang ada dalam Gambar 5.23 terletak pada Tabel 5.23 berikut :

Tabel 5.23 Informasi Element Rancangan Antarmuka *Pop-up* Saring Daftar Objek Layanan Kesehatan

No	Penjelasan	Detail Fungsi
1	Nama Kategori Penyaringan	Untuk memperjelas kategori yang menjadi dasar dari penyaringan
2	Tombol kembali	Sebagai pemicu untuk ditutupnya antarmuka saring
3	<i>Checkbox</i>	Sebagai antarmuka pengguna dalam memilih, ketika <i>checkbox</i> ditekan maka tanda centang akan ditambahkan
4	Tombol Lihat Hasil	Sebagai pemicu untuk menyaring daftar berdasarkan masukan pengguna



Gambar 5.24 Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan

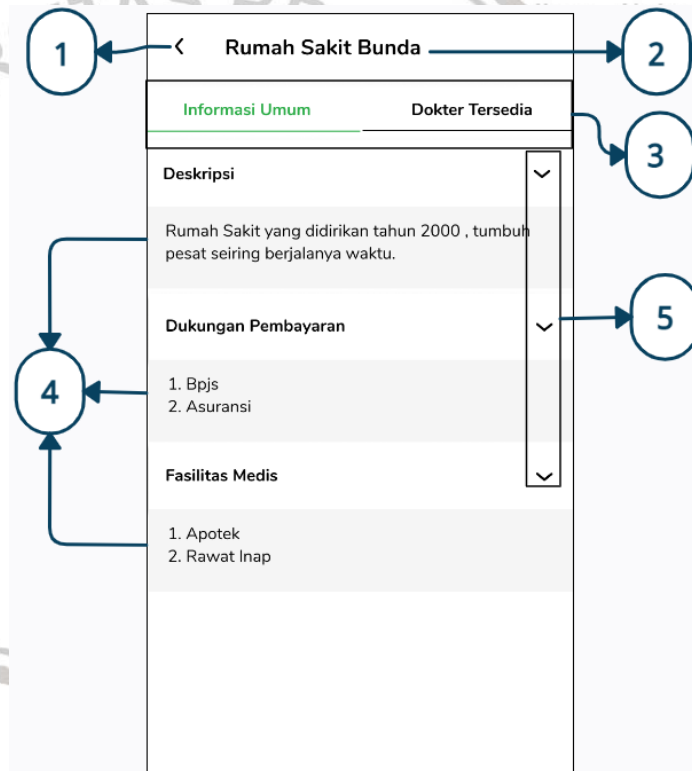
Penjelasan setiap element antarmuka halaman Detail Objek Layanan Kesehatan yang ada dalam Gambar 5.24 terletak pada Tabel 5.26 berikut :

Tabel 5.24 Informasi Element Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan

No	Penjelasan	Detail Fungsi
1	Tombol Kembali	Sebagai pemicu untuk dilakukanya perpindahan ke halaman sebelumnya
2	Tombol Tandai	Untuk menandai objek layanan kesetan pada halaman ini agar dapat dilihat lagi nanti pada halaman Penanda
3	Foto	Sebagai media untuk memperjelas informasi tentang objek layanan kesehatan
4	Informasi dasar	Untuk memberikan informasi dasar berupa nama, jenis dan alamat dari objek layanan kesehatan
5	Tombol Telepon	Untuk melakukan panggilan telepon ke nomor objek layanan kesehatan

Tabel 5.24 Informasi Element Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan (lanjutan)

6	Tombol Navigasi	Untuk berpindah aplikasi ke Google Maps dan melakukan navigasi menuju alamat dari objek layanan kesehatan
7	Tab layout	Sebagai pilihan menu kategori untuk informasi rinci dari objek layanan kesehatan
8	Informasi rinci	Untuk memberikan informasi rinci berupa nama, jenis dan alamat dari objek layanan kesehatan
9	Tombol <i>Dropdown</i>	Untuk membuka atau menutup kolom isi pada informasi rinci

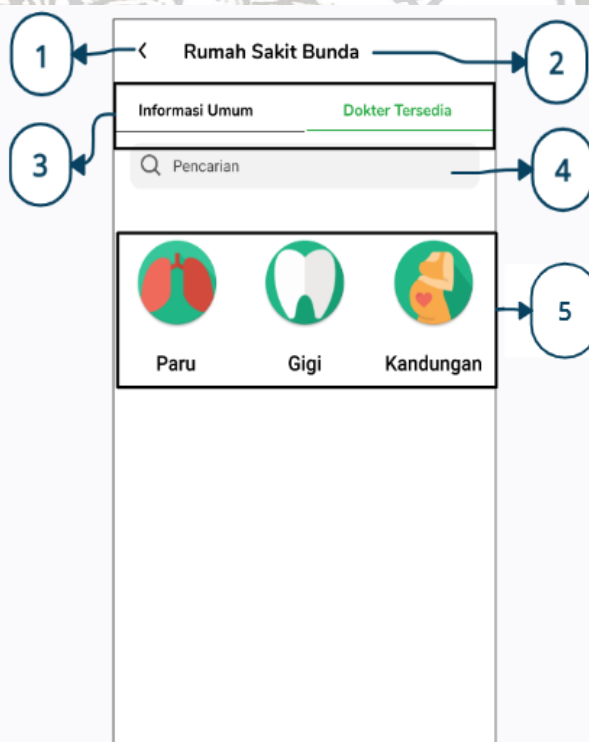


Gambar 5.25 Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Informasi Umum

Penjelasan element antarmuka halaman Detail Objek Layanan Kesehatan Tab Informasi Umum yang ada dalam Gambar 5.25 terletak pada Tabel 5.25 berikut :

Tabel 5.25 Informasi element Rancangan antarmuka halaman Detail Objek Layanan Kesehatan Kolom Informasi Umum

No	Penjelasan	Detail Fungsi
1	Tombol Kembali	Sebagai pemicu untuk dilakukanya perpindahan ke halaman sebelumnya
2	Nama objek layanan kesehatan	Untuk memperjelas objek layanan kesehatan yang sedang dibuka pengguna
3	Tab Layout	Sebagai pilihan menu kategori untuk informasi rinci dari objek layanan kesehatan
4	Informasi rinci	Untuk memberikan informasi rinci berupa nama, jenis dan alamat dari objek layanan kesehatan
5	Tombol <i>Dropdown</i>	Untuk membuka atau menutup kolom isi pada informasi rinci

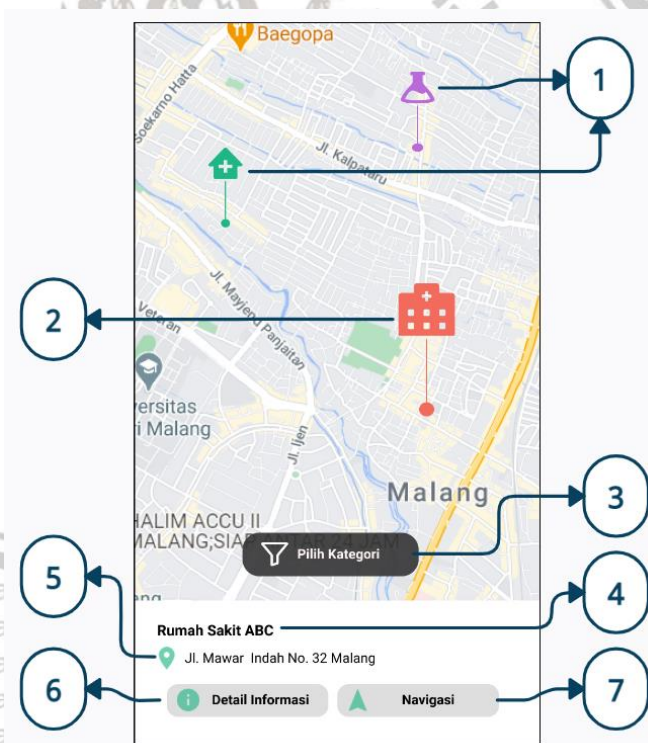


Gambar 5.26 Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Dokter Tersedia

Penjelasan setiap element antarmuka halaman Detail Objek Layanan Kesehatan Tab Dokter Tersedia yang ada dalam Gambar 5.26 terletak pada Tabel 5.26 berikut

Tabel 5.26 Informasi Element Rancangan Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Dokter Tersedia

No	Penjelasan	Detail Fungsi
1	Tombol Kembali	Sebagai pemicu untuk dilakukanya perpindahan ke halaman sebelumnya
2	Nama objek layanan kesehatan	Untuk memperjelas objek layanan kesehatan yang sedang dibuka pengguna
3	Tab Layout	Sebagai pilihan menu kategori untuk informasi rinci dari objek layanan kesehatan
4	Kolom pencarian	Untuk melakukan pencarian
5	Tombol Spesialis	Sebagai pemicu untuk dilakukanya perpindahan ke halaman daftar dokter berdasarkan spesialisasi yang dipilih

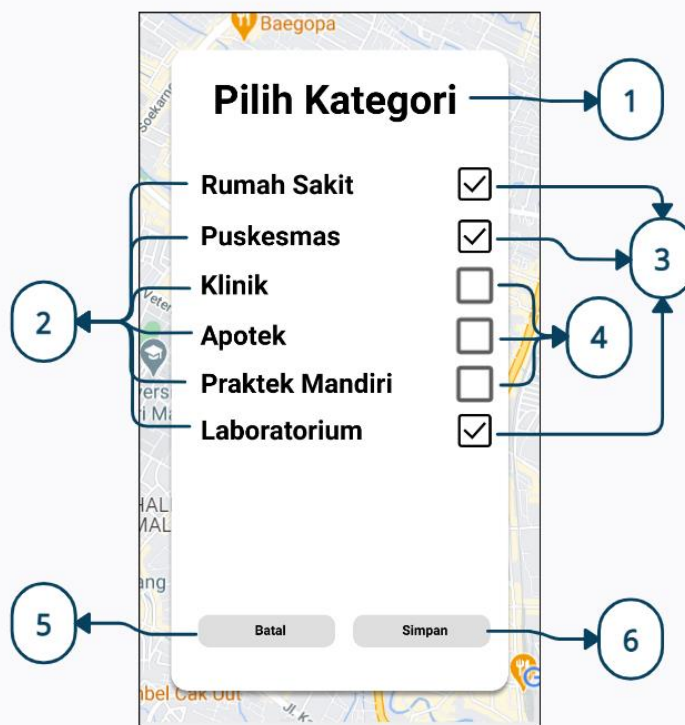


Gambar 5.27 Rancangan Antarmuka Halaman Denah Objek Layanan Kesehatan

Penjelasan setiap element antarmuka halaman Denah Objek Layanan Kesehatan yang ada dalam Gambar 5.27 terletak pada Tabel 5.27 berikut :

Tabel 5.27 Informasi Element Rancangan Antarmuka Halaman Denah

No	Penjelasan	Detail Fungsi
1	Marker pasif	Sebagai representasi dari objek layanan kesehatan yang tidak diklik
2	Marker aktif	Sebagai representasi dari objek layanan kesehatan yang sedang diklik
3	Tombol Pilih Kategori	Untuk membuka atau menutup menu ceklist dari pilihan kategori yang akan ditampilkan
4	Nama objek layanan kesehatan	Untuk memberikan informasi nama objek layanan kesehatan yang sedang diklik
5	Alamat objek layanan kesehatan	Untuk memberikan informasi alamat objek layanan kesehatan yang sedang diklik
6	Tombol Detail Informasi	Sebagai pemicu untuk dilakukanya perpindahan ke halaman Detail Informasi dari objek layanan kesehatan yang sedang diklik
7	Tombol Navigasi	Sebagai pemicu untuk dilakukanya perpindahan ke aplikasi Google Maps dan dilakukan perintah untuk navigasi menuju objek layanan kesehatan yang sedang diklik



Gambar 5.28 Rancangan Antarmuka *Pop-up* Pilih Kategori Halaman Denah

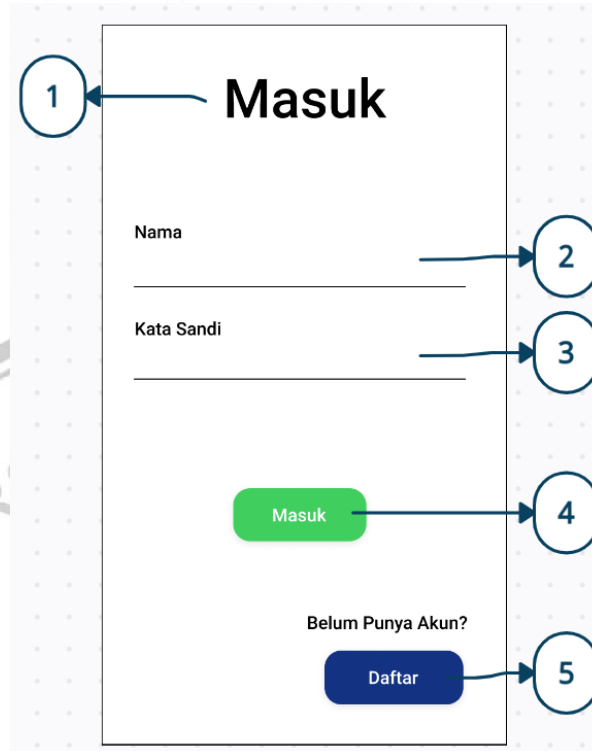
Penjelasan setiap element antarmuka pilih kategori halaman Denah yang ada dalam Gambar 5.28 terletak pada Tabel 5.28 berikut :

Tabel 5.28 Informasi Element Rancangan Antarmuka *Pop-up* Pilih Kategori Halaman Denah

No	Penjelasan	Detail Fungsi
1	Judul Menu	Untuk memperjelas isi dari <i>pop-up</i>
2	Nama pilihan kategori	Sebagai judul dari tampilan <i>pop-up</i>
3	Tombol ceklist (diceklist)	Sebagai penanda bahwa kategori dipilih
4	Tombol ceklist (kosong)	Sebagai penanda bahwa kategori tidak dipilih
5	Tombol Batal	Untuk menutup <i>pop-up</i> pilih kategori tanpa melakukan perubahan pada data yang ditampilkan
6	Tombol Simpan	Untuk menutup <i>pop-up</i> sekaligus melakukan perubahan terbaru pada data yang ditampilkan

5.5.2 Iterasi 1

Pada perancangan antarmuka iterasi 1, terdapat penambahan 3 rancangan antarmuka baru dan perubahan pada rancangan antarmuka halaman beranda.

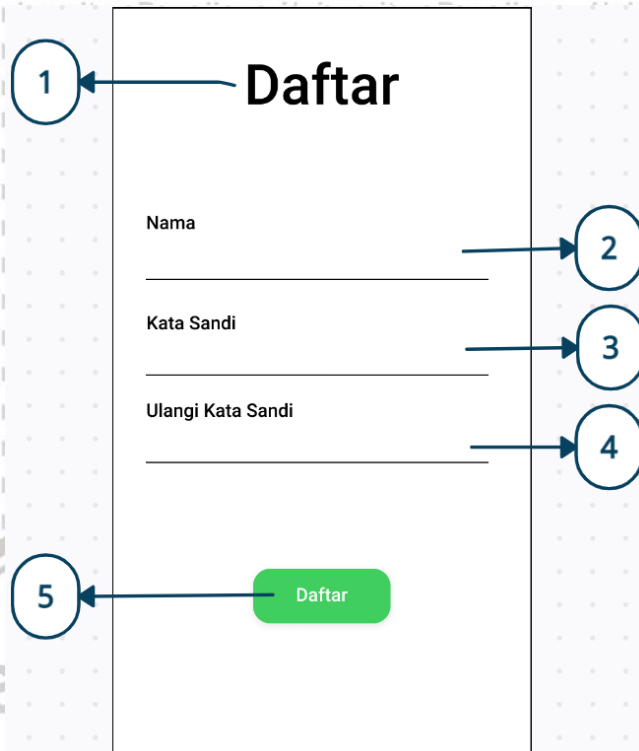


Gambar 5.29 Rancangan Antarmuka Halaman Masuk

Penjelasan Setiap element antarmuka halaman Masuk yang ada dalam Gambar 5.29 terletak pada Tabel 5.29 berikut :

Tabel 5.29 Informasi Element Rancangan Antarmuka Halaman Masuk

No	Penjelasan	Detail Fungsi
1	Nama Halaman	Untuk memperjelas halaman yang sedang dibuka pengguna
2	Kolom Nama	Sebagai tempat Pengguna menuliskan data form Nama
3	Kolom Kata Sandi	Sebagai tempat Pengguna menuliskan data form Kata Sandi
4	Tombol Masuk	Sebagai pemicu dilakukanya proses untuk masuk
5	Tombol Daftar	Sebagai pemicu untuk dilakukanya perpindahan ke halaman Daftar

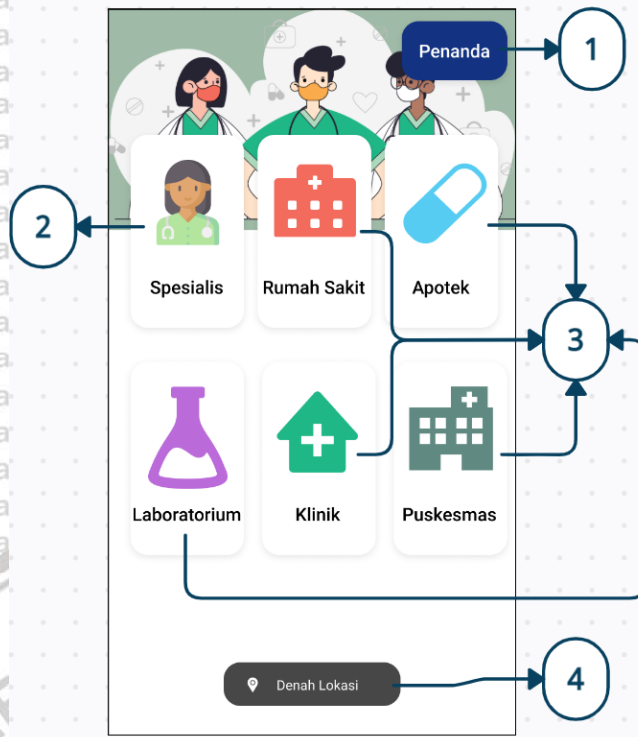


Gambar 5.30 Rancangan Antarmuka Halaman Daftar

Penjelasan Setiap element antarmuka halaman Daftar yang ada dalam Gambar 5.30 terletak pada Tabel 5.30 berikut :

Tabel 5.30 Informasi Element Rancangan Antarmuka Halaman Daftar

No	Penjelasan	Detail Fungsi
1	Nama Halaman	Untuk memperjelas halaman yang sedang dibuka pengguna
2	Kolom Nama	Sebagai tempat Pengguna menuliskan data form Nama
3	Kolom Kata Sandi	Sebagai tempat Pengguna menuliskan data form Kata Sandi
4	Kolom Ulangi Kata Sandi	Sebagai tempat Pengguna menuliskan data form Kata Sandi untuk kedua kalinya dengan tujuan validasi
5	Tombol Daftar	Sebagai pemicu dilakukannya proses untuk Daftar

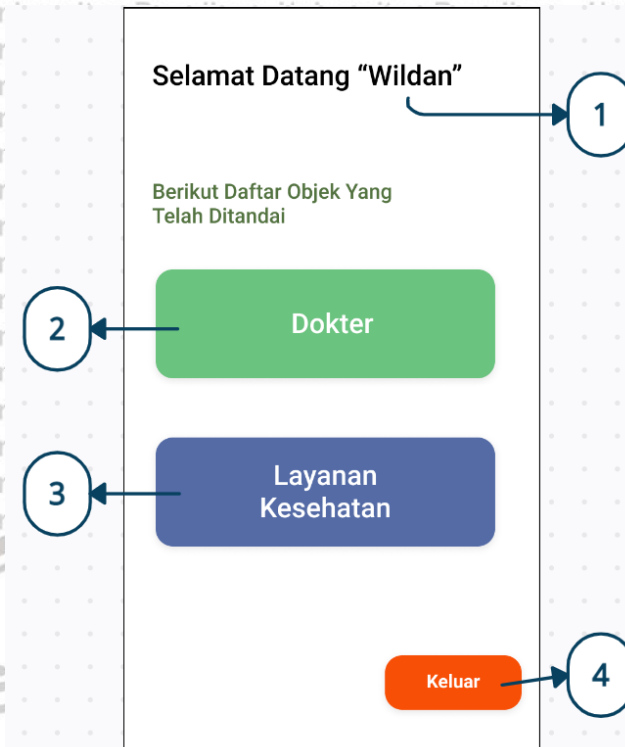


Gambar 5.31 Rancangan Antarmuka Halaman Beranda

Penjelasan Setiap element antarmuka halaman Beranda yang ada dalam Gambar 5.31 terletak pada Tabel 5.31 berikut :

Tabel 5.31 Informasi Element Rancangan Antarmuka Halaman Beranda

No	Penjelasan	Detail Fungsi
1	Tombol Penanda	Sebagai pemicu untuk dilakukanya perpindahan ke halaman Penanda
2	Tombol kategori : Dokter	Sebagai pemicu untuk dilakukanya perpindahan ke halaman Daftar Spesialis
3	Tombol kategori : Layanan Kesehatan	Sebagai pemicu untuk dilakukanya perpindahan ke halaman Daftar Layanan Kesehatan
4	Tombol Denah Lokasi	Sebagai pemicu untuk dilakukanya perpindahan ke halaman Denah Lokasi



Gambar 5.32 Rancangan Antarmuka Halaman Penanda

Penjelasan setiap element antarmuka halaman Penanda yang ada dalam Gambar 5.32 terletak pada Tabel 5.32 berikut :

Tabel 5.32 Informasi Element Rancangan Antarmuka Halaman Penanda

No	Penjelasan	Detail Fungsi
1	Nama Pengguna	Untuk memperjelas informasi tentang akun yang dimasuki
2	Tombol Dokter	Sebagai pemicu untuk berpindah halaman menuju halaman Daftar Dokter yang telah ditandai
3	Tombol Layanan Kesehatan	Sebagai pemicu untuk berpindah halaman menuju halaman Daftar Layanan Kesehatan yang telah ditandai
4	Tombol Keluar	Sebagai pemicu untuk dilakukanya proses keluar dari akun yang sedang dimasuki

BAB 6 IMPLEMENTASI

Pada bab ini dibahas mengenai implementasi dari Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat. Terdapat beberapa aspek bahasan yaitu spesifikasi perangkat keras dan perangkat lunak, batasan-batasan implementasi, implementasi data, dan implementasi kode program.

6.1 Spesifikasi Sistem

Pada bagian spesifikasi sistem ini akan dibahas mengenai detail informasi perangkat keras dan perangkat lunak dari perangkat yang digunakan dalam proses pengembangan Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat.

Tabel 6.1 Spesifikasi Perangkat Keras

Komponen	Spesifikasi
ACER NITRO 5	<ol style="list-style-type: none"> 1. <i>Processor</i> Intel Core i7 8750H 2. <i>Memory</i> 12GB DDR4 3. <i>Graphics</i> Nvidia 1050Ti with 4GB GDDR5 VRAM 4. <i>STORAGE</i> 1TB

Tabel 6.2 Spesifikasi Perangkat Lunak

Komponen	Spesifikasi
Sistem Operasi	1. Windows 10
<i>Integrated Development Environment</i> (IDE)	1. Android Studio
Emulator Android	<ol style="list-style-type: none"> 1. Android 5 (API 21) 2. Android 11 (API 30)

6.2 Batasan-batasan Implementasi

Aplikasi Malang Sehat memiliki beberapa batasan implementasi dalam pengembangannya. Batasan-batasan tersebut adalah sebagai berikut :

1. Aplikasi Malang Sehat dibangun dengan *framework* yang mendukung minimal Android 5 (API 21).
2. Aplikasi Malang Sehat membutuhkan koneksi internet untuk terhubung dengan basis data

6.3 Implementasi Tabel Basis Data

Tabel basis data diimplementasikan menggunakan MySQL. Implementasi data dibuat sesuai dengan perancangan yang telah dilakukan pada bab sebelumnya yaitu pada bagian Perancangan Tabel Basis Data. Berikut adalah hasil implementasi tabel basis data pada Modul Informasi Layanan Kesehatan Aplikasi Malang Sehat.

6.3.1 Iterasi 0

Pada iterasi 0, terdapat 7 buah implementasi tabel yang dihasilkan, yaitu tabel dokter, tabel dukungan_penmbayaran, tabel fasilitas, tabel fasilitas_lakes, tabel jenis_rumah_sakit, tabel kategori, tabel lakes,

id	nama	foto	pendidikan	pengalaman	profil	spesialis
6	dr. Tatit Nurseta, Sp. OG	https://	Sp. OG - Spesialis Obstetri dan Ginekologi - Univer...	Dokter Obygn - Rumah Sakit Lavalette	Dr. dr. Tatit Nurseta Sp. OG(I Onk adalah Dokter	Kandungan
26	Dr. dr. Krisni Subandiya Sp.A (K)	https://	Sp.A - Spesialis Anak - Universitas Airlangga	Dokter Anak di Rumah Sakit Lavalette, RSUD Saiful	Dr. dr. Krisni Subandi Sp.A (K) adalah Dokter	Anak
28	dr. Ariani Setyanings Sp. A	https://	Sp.A - Spesialis Anak - Universitas Brawijaya	Dokter Anak - Rumah Sakit Lavalette	dr. Ariani Setyanir Sp.A adalah Dokter Anak	Anak

Gambar 6.1 Implementasi Tabel Dokter

Implementasi basis data tabel Dokter tertera dalam Gambar 6.1. Pada tabel tersebut tiap data memiliki id yang berbeda satu dengan lainnya, foto yang berisikan URL tempatnya berada, narasi pendidikan, narasi pengalaman, narasi profil, dan *foreign key* spesialis yang berasal dari tabel Spesialis.

dukunganPembayaran
BPJS
Jaminan Kesehatan Tenaga Kerja
Jasa Raharja
Surat Keterangan Tidak Mampu
Umum

Gambar 6.2 Implementasi Tabel Dukungan_pembayaran

Implementasi basis data tabel Dukungan_pembayaran tertera dalam Gambar 6.2. Pada tabel tersebut hanya berisikan satu kolom data yaitu dukunganPembayaran. Nantinya data-data pada kolom ini akan dijadikan acuan foreign key untuk tabel lainnya.

fasilitas
Ambulance
Apotek
Hemodialisa
ICU
IGD
Instalasi Bersalin
Instalasi Fisioterapi
Instalasi Gizi

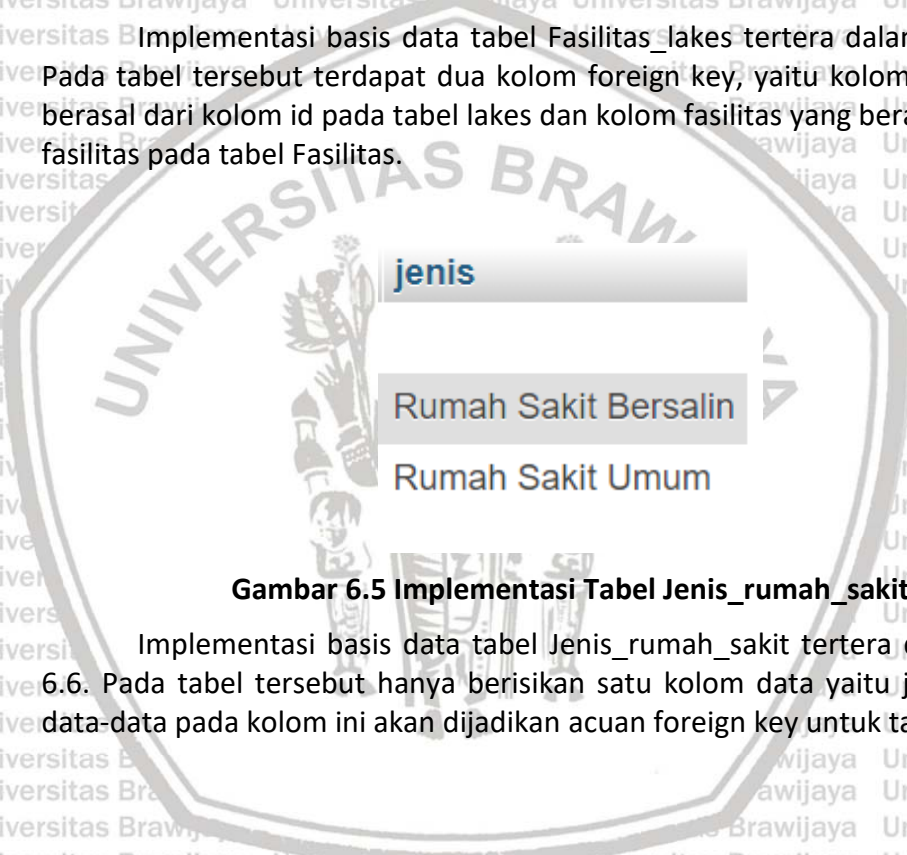
Gambar 6.3 Implementasi Tabel Fasilitas

Implementasi basis data tabel Fasilitas tertera dalam Gambar 6.3. Pada tabel tersebut hanya berisikan satu kolom data yaitu fasilitas. Nantinya data-data pada kolom ini akan dijadikan acuan foreign key untuk tabel lainnya.

id	id_lakes	fasilitas
1	2	Ambulance
2	2	Apotek
3	2	ICU
4	2	Laboratorium
5	2	Hemodialisa
6	2	Radiologi
7	2	Rawat Inap

Gambar 6.4 Implementasi Tabel Fasilitas_lakes

Implementasi basis data tabel Fasilitas_lakes tertera dalam Gambar 6.5. Pada tabel tersebut terdapat dua kolom foreign key, yaitu kolom id_lakes yang berasal dari kolom id pada tabel lakes dan kolom fasilitas yang berasal dari kolom fasilitas pada tabel Fasilitas.



Gambar 6.5 Implementasi Tabel Jenis_rumah_sakit

Implementasi basis data tabel Jenis_rumah_sakit tertera dalam Gambar 6.6. Pada tabel tersebut hanya berisikan satu kolom data yaitu jenis. Nantinya data-data pada kolom ini akan dijadikan acuan foreign key untuk tabel lainnya.

kategori
Apotek
Klinik
Laboratorium
Optik
Praktek Mandiri
Puskesmas
Rumah Sakit
Unit Transfusi Darah

Gambar 6.6 Implementasi Tabel Kategori

Implementasi basis data tabel Kategori tertera dalam Gambar 6.7. Pada tabel tersebut hanya berisikan satu kolom data yaitu kategori. Nantinya data-data pada kolom ini akan dijadikan acuan foreign key untuk tabel lainnya.

id	nama	kategori	alamat	kecamatan	telepon	latitude	longitru	deskripsi	foto	kelas	jenis	senin	selasa	rabu	kamis	Jumat	sabtu	minggu
1	RSUD Dr. Saiful Anwar Malang	Rumah Sakit	Jl. Jaksu Agung Suprapt No 2, Klojen, Kec. Klojen...	Klojen	+620341	-7.97235	112.631	RSUD Dr. Saiful Anwar merupakan rumah sakit umum d...	https	A	Rumah Sakit Umum	24 jam	24 jam	24 jam	24 jam	24 jam	24 jam	24 jam
2	Lavale Generi Hospit	Rumah Sakit	Jl. Sulawesi No 16, Kasin, Kec. Klojen, Kota Malan...	Klojen	+623414	-7.96573	112.637	RS Lavalette merupakan rumah sakit yang berada di ...	https	A	Rumah Sakit Umum	24 jam	24 jam	24 jam	24 jam	24 jam	24 jam	24 jam
6	RS Persad Hospit	Rumah Sakit	Komplek Araya Business Centre Kav. 2-4 Jalan Panj...	Blimbing	+623412	-7.93498	112.650	RS Persada Hospital merupakan rumah sakit yang b...	https cgvir	B	Rumah Sakit Umum	24 jam	24 jam	24 jam	24 jam	24 jam	24 jam	24 jam

Gambar 6.7 Implementasi Tabel Lakes

Implementasi basis data tabel Lakes tertera dalam Gambar 6.8. Pada tabel tersebut terdapat 18 kolom, pertama adalah kolom id yang otomatis dihasilkan dan bersifat unik saat data dibuat, kedua kolom nama yang berisikan nama dari objek layanan kesehatan, ketiga kolom kategori yang merupakan foreign key dari kolom kategori pada tabel Kategori, keempat kolom alamat yang berisi narasi dari alamat objek layanan kesehatan, kelima kolom telepon yang berisikan nomor telepon dari objek layanan kesehatan, keenam dan ketujuh adalah kolom latitude dan longitude yang nantinya berisikan data koordinat lokasi dari objek layanan kesehatan, kedelapan kolom deskripsi yang berisi narasi penjelasan tentang objek layanan kesehatan, kesembilan kolom foto yang berisikan alamat URL dari foto objek layanan kesehatan berada, kesepuluh kolom kelas yang mendeskripsikan kualitas dari suatu objek layanan kesehatan, kesebelas kolom jenis yang

merupakan *foreign key* dari kolom jenis pada tabel Jenis. Kedua belas hingga kedelapan belas merupakan kolom hari yang berisi informasi jam buka objek layanan kesehatan.

6.3.2 Iterasi 1

Pada iterasi 1, terdapat 3 buah implementasi tabel baru yang dihasilkan, yaitu tabel akun, tabel penanda_dokter dan tabel penanda_lakes.

id	nama	kata_sandi
1	wildan	wildan
4	wildan1	wildan1

Gambar 6.8 Implementasi Tabel Akun

Implementasi basis data tabel Akun tertera dalam Gambar 6.1. Pada tabel tersebut terdapat tiga kolom yaitu id yang dihasilkan secara unik ketika data dibuat, nama yang berasal dari masukan pengguna dan bersifat unik, dan terakhir kata sandi yang berasal dari masukan pengguna.

id	id_akun	id_dokter
3	1	6
4	1	31

Gambar 6.9 Implementasi Tabel Penanda_dokter

Implementasi basis data tabel Penanda_dokter tertera dalam Gambar 6.9. Pada tabel tersebut terdapat 3 kolom, pertama kolom id yang otomatis dihasilkan dan bersifat unik saat data dibuat, kedua kolom id_akun yang merupakan *foreign key* dari kolom id pada tabel Akun dan ketiga kolom id_dokter yang merupakan *foreign key* dari kolom id pada tabel Dokter.

id	id_akun	id_lakes
22	1	5
23	1	1

Gambar 6.10 Implementasi Tabel Penanda_lakes

Implementasi basis data tabel Penanda_lakes tertera dalam Gambar 6.10. Pada tabel tersebut terdapat 3 kolom, pertama kolom id yang otomatis dihasilkan

dan bersifat unik saat data dibuat, kedua kolom id_akun yang merupakan foreign key dari kolom id pada tabel Akun dan ketiga kolom id_lakes yang merupakan foreign key dari kolom id pada tabel Lakes.

6.4 Implementasi Kode Program

Pada bagian implementasi kode program ini berisi pemaparan kode program pada fungsi-fungsi yang terdapat dalam use case. Kode program ditulis dalam bahasa java pada Android Studio.

6.4.1 Iterasi 0

Pada iterasi 0, terdapat 3 use case yang diimplementasikan menjadi kode program, yaitu Lihat Informasi Objek Layanan Kesehatan, Lihat Informasi Dokter dan Lihat Denah Objek Layanan Kesehatan.

6.4.1.1 Kode Program Lihat Informasi Objek Layanan Kesehatan

Dalam Melihat informasi dokter, kode program akan dijabarkan menjadi 4 fungsi, yaitu : `getListLakes()`, `getDetailLakes()`, `getInfoUmumLakes()`, dan `getDokterLakes()`.

Pertama fungsi `getListLakes()` terdapat pada Tabel 6.3 dan penjelasannya terdapat pada Tabel 6.4. Fungsi pertama ini akan dijalankan apabila pengguna berada pada halaman Daftar Lakes.

Kedua fungsi `getDetailLakes()` terdapat pada Tabel 6.5 dan penjelasannya terdapat pada Tabel 6.6. Fungsi kedua ini akan dijalankan apabila pengguna memilih salah satu objek layanan kesehatan yang berada di halaman Daftar Layanan Kesehatan dan masuk ke halaman Detail Layanan Kesehatan.

Ketiga fungsi `getInfoUmumLakes()` terdapat pada Tabel 6.7 dan penjelasannya terdapat pada Tabel 6.8. Fungsi ketiga ini akan dijalankan apabila pengguna berada pada halaman Detail Layanan Kesehatan dan berada pada tabLayout InformasiUmum

Keempat fungsi `getDokterLakes()` terdapat pada Tabel 6.9 dan penjelasannya terdapat pada Tabel 6.10. Fungsi keempat ini akan dijalankan apabila pengguna berada pada halaman Detail Layanan Kesehatan dan berada pada tabLayout Dokter Tersedia

Tabel 6.3 Kode Program Fungsi `getListLakes()`

No	Kode Program
1	<code>callLakes = apiService.getLakesByMultipleFilter(kategori, nama,</code>
2	<code>pembayaran, fasilitas);</code>
3	<code>callLakes.enqueue(new Callback<List<LakesModel>>() {</code>
4	<code> @Override</code>
5	<code> public void onResponse(Call<List<LakesModel>> call,</code>
6	<code> Response<List<LakesModel>> response) {</code>
7	<code> //cek jika internet mati maka response body null</code>
8	<code> if (response.body() != null) {</code>
9	<code> //cek isi dari response body</code>
10	<code> progressBar.setVisibility(View.GONE);</code>

Tabel 6.3 Kode Program Fungsi `getListLakes ()` (lanjutan)

No	Kode Program
11	<code>if (!response.body().isEmpty()) {</code>
12	<code>textKosong.setVisibility(View.GONE);</code>
13	<code>listLakes.addAll(response.body());</code>
14	<code>RecyclerView.LayoutManager layoutManager = new</code>
15	<code>LinearLayoutManager(DaftarLakes.this);</code>
16	<code>rcDaftar.setLayoutManager(layoutManager);</code>
17	<code>adapterLakes = new AdapterLakes(listLakes);</code>
18	<code>rcDaftar.setAdapter(adapterLakes);</code>
19	<code>}else {</code>
20	<code>textKosong.setVisibility(View.VISIBLE);</code>
21	<code>}</code>
22	<code>Toast.makeText(DaftarLakes.this, "Jaringan Terputus",</code>
23	<code>Toast.LENGTH_SHORT).show();</code>
24	<code>progressBar.setVisibility(View.GONE);</code>
25	<code>}</code>
26	<code>}</code>
27	<code>@Override</code>
28	<code>public void onFailure(Call<List<LakesModel>> call, Throwable</code>
29	<code>t) {</code>
30	<code>progressBar.setVisibility(View.GONE);</code>
31	<code>Toast.makeText(DaftarLakes.this, "Gagal Menghubungkan Ke Basis</code>
32	<code>Data", Toast.LENGTH_LONG).show();</code>
33	<code>}</code>
34	<code>});</code>

Tabel 6.4 Penjelasan Kode Program Fungsi `getListLakes ()`

Baris	Penjelasan
1-2	Model api dan call berguna untuk menyambungkan ke basis data.
3-26	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi <code>onResponse()</code> akan dipanggil dan data yang diambil akan dimasukkan kedalam model response. Selanjutnya dilakukan seleksi apabila <code>response.body</code> tidak bernilai <code>null</code> maka nilainya akan dimasukkan ke dalam list lakesModel. Kemudian akan dibuat <code>recyclerView</code> yang datanya berisikan list tersebut.
27-34	Fungsi <code>onFailure()</code> adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

Tabel 6.5 Kode Program Fungsi `getDetailLakes ()`

No	Kode Program
1	<code>lakesModel = (LakesModel)</code>
2	<code>getIntent().getExtras().getSerializable("model");</code>
3	<code>telepon = lakesModel.getTelepon();</code>
4	<code>jenis = lakesModel.getJenis();</code>
5	<code>tvNama.setText(lakesModel.getNama());</code>
6	<code>tvAlamat.setText(lakesModel.getAlamat());</code>
7	<code>tvJenis.setText(jenis);</code>
8	<code>Glide</code>

Tabel 6.5 Kode Program Fungsi `getDetailLakes()` (lanjutan)

No	Kode Program
9	<code>.with(this)</code>
10	<code>.load(lakesModel.getFoto())</code>
11	<code>.diskCacheStrategy(DiskCacheStrategy.ALL)</code>
12	<code>.listener(new RequestListener<Drawable>() {</code>
13	<code>@Override</code>
14	<code>public boolean</code> <code>onLoadFailed</code> (@Nullable <code>GlideException</code>
15	<code>e, Object model, Target<Drawable> target, boolean isFirstResource)</code>
16	{
17	<code>return false;</code>
18	<code>}</code>
19	<code>@Override</code>
20	<code>public boolean</code> <code>onResourceReady</code> (<code>Drawable resource,</code>
21	<code>Object model, Target<Drawable> target, DataSource dataSource,</code>
22	<code>boolean isFirstResource) {</code>
23	<code>progressBar.setVisibility(View.GONE);</code>
24	<code>return false;</code>
25	<code>}</code>
26	<code>})</code>
27	<code>.into(foto);</code>

Tabel 6.6 Penjelasan Kode Program Fungsi `getDetailLakes()`

Baris	Penjelasan
1-2	Inisialisasi model <code>lakesModel</code> yang nilainya diambil <code>lakesModel</code> yang sudah dimuat pada halaman Daftar Lakes dan memasukkan nilainya ke <code>TextView</code>
3-7	Inisialisasi variabel menggunakan atribut dari <code>lakesModel</code>
8-27	Framework <code>Glide</code> yang berguna untuk menampilkan foto dari dokter. Aset foto yang ditampilkan diambil pada internet dengan menggunakan alamat yang terdapat dalam atribut <code>dokterModel</code> .

Tabel 6.7 Kode Program Fungsi `getInfoUmumLakes()`

No	Kode Program
1	<code>lakesModel = (LakesModel)</code>
2	<code>getActivity().getIntent().getExtras().getSerializable("model");</code>
3	<code>tvDeskripsi.setText(lakesModel.getDeskripsi());</code>
4	<code>String query = lakesModel.getId();</code>
5	<code>ApiConnectionAndCaching apiConnectionAndCaching = new</code>
6	<code>ApiConnectionAndCaching();</code>
7	<code>ApiService apiService =</code>
8	<code>apiConnectionAndCaching.getApi(getContext()).create(ApiService.class);</code>
9	<code>Call<List<itemModel>> call = apiService.getFasilitasLakes(query);</code>
10	<code>call.enqueue(new Callback<List<itemModel>>() {</code>
11	<code>@Override</code>
12	<code>public void</code> <code>onResponse</code> (<code>Call<List<itemModel>> call,</code>
13	<code>Response<List<itemModel>> response) {</code>
14	<code>if (response.body() != null && isAdded()) {</code>
15	<code>if (response.body() != null && isAdded()) {</code>

Tabel 6.7 Kode Program Fungsi getInfoUmumLakes() (lanjutan)

No	Kode Program
16	<code>fasilitasList = new ArrayList<>(response.body());</code>
17	<code>adapter = new AdapterItem(fasilitasList);</code>
18	<code>RecyclerView.LayoutManager layoutManager = new</code>
19	<code>GridLayoutManager(getContext(), 2);</code>
20	<code>rcFasilitas.setLayoutManager(layoutManager);</code>
21	<code>rcFasilitas.setAdapter(adapter);</code>
22	<code>}</code>
23	<code>}</code>
24	<code>@Override</code>
25	<code>public void onFailure(Call<List<itemModel>> call, Throwable t)</code>
26	<code>{</code>
27	<code> Toast.makeText(getContext(), "Gagal Menghubungkan ke basis</code>
28	<code>data", Toast.LENGTH_LONG).show();</code>
29	<code>}</code>
30	<code>});</code>
31	<code>call = apiService.getPembayaran(query);</code>
32	<code>call.enqueue(new Callback<List<itemModel>>() {</code>
33	<code> @Override</code>
34	<code> public void onResponse(Call<List<itemModel>> call,</code>
35	<code> Response<List<itemModel>> response) {</code>
36	<code> if (response.body() != null && isAdded()) {</code>
37	<code> dukunganPembayaranList = new</code>
38	<code> ArrayList<>(response.body());</code>
39	<code> adapter = new AdapterItem(dukunganPembayaranList);</code>
40	<code> RecyclerView.LayoutManager layoutManager = new</code>
41	<code> GridLayoutManager(getContext(), 2);</code>
42	<code> rcPembayaran.setLayoutManager(layoutManager);</code>
43	<code> rcPembayaran.setAdapter(adapter);</code>
44	<code> }</code>
45	<code>}</code>
46	<code>@Override</code>
47	<code>public void onFailure(Call<List<itemModel>> call, Throwable t)</code>
48	<code>{</code>
49	<code> Toast.makeText(getContext(), "Gagal Menghubungkan ke basis</code>
50	<code>data", Toast.LENGTH_LONG).show();</code>
51	<code>}</code>
52	<code>});</code>

Tabel 6.8 Penjelasan Kode Program Fungsi getInfoUmumLakes()

Baris	Penjelasan
1-2	Inisialisasi model lakesModel yang nilainya diambil lakesModel yang sudah dimuat pada halaman Daftar Lakes.
5-10	Model api dan call berguna untuk menyambungkan ke basis data
11-25	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi onResponse() akan dipanggil dan data yang diambil akan dimasukkan kedalam model response. Selanjutnya dilakukan seleksi apabila response.body tidak bernilai null maka nilainya akan dimasukkan ke dalam list itemModel. Kemudian akan dibuat recyclerView yang datanya berisikan arrayList spesialisModel tersebut.
24-30	Fungsi onFailure() adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

Tabel 6.9 Kode Program Fungsi `getDokterLakes()` (lanjutan)

No	Kode Program
38	<code>rcSpesialis.setLayoutManager(layoutManager2);</code>
39	<code>rcSpesialis.setAdapter(adapterDaftarSpesialis);</code>
40	<code>adapterDokter = new AdapterDokter(dokterModelList);</code>
41	<code>if (dokterModelList.size() == 0) {</code>
42	<code>textKosong.setVisibility(View.VISIBLE);</code>
43	<code>progressBar.setVisibility(View.GONE);</code>
44	<code>}</code>
45	<code>}</code>
46	<code>}</code>
47	<code>@Override</code>
48	<code>public void onFailure(Call<List<DokterModel>> call, Throwable</code>
49	<code>t) {</code>
50	<code>Toast.makeText(getContext(), "Gagal menghubungkan ke basis</code>
51	<code>data", Toast.LENGTH_LONG).show();</code>
52	<code>progressBar.setVisibility(View.GONE);</code>
53	<code>}</code>
54	<code>});</code>

Tabel 6.10 Penjelasan Kode Program Fungsi `getDokterLakes()`

Baris	Penjelasan
1-3	Inisialisasi model <code>lakesModel</code> yang nilainya diambil <code>lakesModel</code> yang sudah dimuat pada halaman Daftar Lakes.
4-11	Model api dan call berguna untuk menyambungkan ke basis data.
12-46	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi <code>onResponse()</code> akan dipanggil dan data yang diambil akan dimasukkan kedalam model <code>response</code> . Selanjutnya dilakukan seleksi apabila <code>response.body</code> tidak bernilai <code>null</code> maka nilainya akan dimasukkan ke dalam list <code>dokterModel</code> . Kemudian akan dibuat <code>recyclerView</code> yang datanya berisikan list tersebut.
47-54	Fungsi <code>onFailure()</code> adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

6.4.1.2 Kode Program Lihat Informasi Dokter

Dalam Melihat informasi dokter, kode program akan dijabarkan menjadi 5 fungsi, yaitu : `getListSpesialis()`, `getListDokter()`, `getDetailDokter()`, `getJadwalDokter()`, dan `getInfoUmumDokter()`.

Pertama fungsi `getListSpesialis()` terdapat pada Tabel 6.11 dan penjelasannya terdapat pada Tabel 6.12. Fungsi pertama ini akan dijalankan apabila pengguna berada pada halaman Daftar Spesialis.

Kedua fungsi `getListDokter()` terdapat pada Tabel 6.13 dan penjelasannya terdapat pada Tabel 6.14. Fungsi kedua ini akan dijalankan apabila pengguna berada pada halaman Daftar Dokter.

Ketiga fungsi `getDetailDokter()` terdapat pada Tabel 6.15 dan penjelasannya terdapat pada Tabel 6.16. Fungsi ketiga ini akan dijalankan apabila pengguna memilih salah satu objek dokter yang berada di halaman Daftar Dokter dan masuk ke halaman Detail Dokter.

Keempat fungsi `getJadwalDokter()` terdapat pada Tabel 6.17 dan penjelasannya terdapat pada Tabel 6.18 Fungsi keempat ini akan dijalankan apabila pengguna berada pada halaman Detail Dokter dan berada pada tabLayout Praktik

Kelima fungsi `getInfoUmumDokter()` terdapat pada Tabel 6.19 dan penjelasannya terdapat pada Tabel 6.20. Fungsi kelima ini akan dijalankan apabila pengguna berada pada halaman Detail Dokter dan berada pada tabLayout Informasi Umum

Tabel 6.11 Kode Program Fungsi `getListSpesialis()`

No	Kode Program
1	<code>ApiConnectionAndCaching apiConnectionAndCaching = new</code>
2	<code>ApiConnectionAndCaching();</code>
3	<code>ApiService apiService =</code>
4	<code>apiConnectionAndCaching.getApi(this).create(ApiService.class);</code>
5	<code>Call<List<SpesialisModel>> call =</code>
6	<code>apiService.getDaftarSpesialis("");</code>
7	<code>call.enqueue(new Callback<List<SpesialisModel>>() {</code>
8	<code> @Override</code>
9	<code> public void onResponse(Call<List<SpesialisModel>> call,</code>
10	<code>Response<List<SpesialisModel>> response) {</code>
11	<code> if (response.body() != null) {</code>
12	<code> List<SpesialisModel> spesialisModels = new</code>
13	<code>ArrayList<>(response.body());</code>
14	<code> progressBar.setVisibility(View.GONE);</code>
15	<code> adapter = new</code>
16	<code>AdapterDaftarSpesialis(spesialisModels, "");</code>
17	<code> RecyclerView.LayoutManager layoutManager = new</code>
18	<code>GridLayoutManager(DaftarSpesialis.this, 3);</code>
19	<code> recyclerView.setLayoutManager(layoutManager);</code>
20	<code> recyclerView.setAdapter(adapter);</code>
21	<code> if (spesialisModels.size() == 0) {</code>
22	<code> textKosong.setVisibility(View.VISIBLE);</code>
23	<code> }</code>
24	<code> }</code>
25	<code> }</code>
26	<code> @Override</code>
27	<code> public void onFailure(Call<List<SpesialisModel>> call,</code>
28	<code>Throwable t) {</code>
29	<code> Toast.makeText(DaftarSpesialis.this, "Gagal Menghubungkan</code>
30	<code>Ke Basis Data", Toast.LENGTH_LONG).show();</code>
31	<code> }</code>
32	<code> });</code>

Tabel 6.12 Penjelasan Kode Program Fungsi `getListSpesialis()`

Baris	Penjelasan
1-7	Model api dan call berguna untuk menyambungkan ke basis data
8-25	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi <code>onResponse()</code> akan dipanggil dan data yang diambil akan dimasukkan kedalam model response. Selanjutnya dilakukan seleksi

Tabel 6.12 Penjelasan Kode Program Fungsi getListSpesialis () (lanjutan)

Baris	Penjelasan
26-32	apabila response.body tidak bernilai null maka nilainya akan dimasukkan ke dalam arrayList spesialisModel. Kemudian akan dibuat recyclerView yang datanya berisikan list spesialisModel tersebut. Fungsi onFailure() adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

Tabel 6.13 Kode Program Fungsi getListDokter ()

No	Kode Program
1	<code>progressBar.setVisibility(View.VISIBLE);</code>
2	<code>String nama = etCari.getText().toString();</code>
3	<code>String id_lakes = getIntent().getStringExtra("id_lakes");</code>
4	<code>listModel = new ArrayList<>();</code>
5	<code>adapter = new AdapterDokter(listModel);</code>
6	<code>rcDaftarDokter.setAdapter(adapter);</code>
7	<code>ApiService apiService =</code>
8	<code>apiConnectionAndCaching.getApi(this).create(ApiService.class);</code>
9	<code>call = apiService.getDaftarDokter(spesialis, id_lakes, nama,</code>
10	<code>filteredHari);</code>
11	<code>call.enqueue(new Callback<List<DokterModel>>() {</code>
12	<code> @Override</code>
13	<code> public void onResponse(Call<List<DokterModel>> call,</code>
14	<code> Response<List<DokterModel>> response) {</code>
15	<code> progressBar.setVisibility(View.GONE);</code>
16	<code> if (response.body().size() > 0) {</code>
17	<code> listModel.addAll(response.body());</code>
18	<code> textKosong.setVisibility(View.GONE);</code>
19	<code> adapter = new AdapterDokter(listModel);</code>
20	<code> RecyclerView.LayoutManager layoutManager = new</code>
22	<code>LinearLayoutManager(DaftarDokter.this);</code>
23	<code> rcDaftarDokter.setLayoutManager(layoutManager);</code>
24	<code> rcDaftarDokter.setAdapter(adapter); }</code>
25	<code> else {</code>
26	<code> textKosong.setVisibility(View.VISIBLE);</code>
27	<code> }</code>
28	<code> }</code>
29	<code> @Override</code>
30	<code> public void onFailure(Call<List<DokterModel>> call, Throwable</code>
31	<code> t) {</code>
32	<code> if (!call.isCanceled()){</code>
33	<code> Toast.makeText(DaftarDokter.this, "Gagal</code>
34	<code> menghubungkan ke basis data",</code>
35	<code> Toast.LENGTH_LONG).show();</code>
36	<code> }</code>
37	<code> }</code>
38	<code>});</code>

Tabel 6.14 Penjelasan Kode Program Fungsi getListDokter ()

Baris	Penjelasan
1-11	Instansiasi dan inisialisasi variabel dan model

Tabel 6.14 Penjelasan Kode Program Fungsi `getListDokter()` (lanjutan)

Baris	Penjelasan
12-28	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi <code>onResponse()</code> akan dipanggil dan data yang diambil akan dimasukkan kedalam model response. Selanjutnya dilakukan seleksi apabila <code>response.body</code> tidak bernilai <code>null</code> maka nilainya akan dimasukkan ke dalam list dokterModel. Kemudian akan dibuat <code>recyclerView</code> yang datanya berisikan list tersebut.
29-38	Fungsi <code>onFailure()</code> adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

Tabel 6.15 Kode Program Fungsi `getDetailDokter()`

No	Kode Program
1	<code>dokterModel = (DokterModel)</code>
2	<code>getIntent().getExtras().getSerializable("dokterModel");</code>
3	<code>tvNama.setText(dokterModel.getNama());</code>
4	<code>tvSpesialis.setText(dokterModel.getSpesialis());</code>
5	<code>Glide</code>
6	<code>.with(this)</code>
7	<code>.load(dokterModel.getFoto())</code>
8	<code>.diskCacheStrategy(DiskCacheStrategy.ALL)</code>
9	<code>.listener(new RequestListener<Drawable>() { @Override</code>
10	<code>public boolean onLoadFailed(@Nullable GlideException</code>
11	<code>e, Object model, Target<Drawable> target, boolean isFirstResource)</code>
12	<code>{</code>
13	<code>return false;</code>
14	<code>}</code>
15	<code>@Override</code>
16	<code>public boolean onResourceReady(Drawable</code>
17	<code>resource, Object model, Target<Drawable> target, DataSource</code>
18	<code>dataSource, boolean isFirstResource) {</code>
19	<code>progressBar.setVisibility(View.GONE);</code>
20	<code>return false;</code>
21	<code>}</code>
22	<code>})</code>
23	<code>.into(foto);</code>

Tabel 6.16 Penjelasan Kode Program Fungsi `getDetailDokter()`

Baris	Penjelasan
1-2	Inisialisasi model dokterModel yang nilainya diambil dokterModel yang
3-4	sudah muat pada halaman Daftar Dokter.
5-23	Mengisi text pada TextView nama dan spesialis menggunakan atribut dari dokterModel
5-23	Framework Glide yang berguna untuk menampilkan foto dari dokter. Aset foto yang ditampilkan diambil pada internet dengan menggunakan alamat yang terdapat dalam atribut dokterModel.

Tabel 6.17 Kode Program Fungsi `getJadwalDokter()`

No	Kode Program
1	<code>dokterModel= (DokterModel)</code>
2	<code>getActivity().getIntent().getExtras().getSerializable("dokterModel</code>
3	<code>");</code>
4	<code>String query = dokterModel.getId();</code>
5	<code>ApiConnectionAndCaching apiConnectionAndCaching = new</code>
6	<code>ApiConnectionAndCaching();</code>
7	<code>ApiService apiService =</code>
8	<code>apiConnectionAndCaching.getApi(getContext()).create(ApiService.cla</code>
9	<code>ss);</code>
10	<code>Call<List<JadwalDokter>> call = apiService.getJadwalDokter(query);</code>
11	<code>call.enqueue(new Callback<List<JadwalDokter>>() {</code>
12	<code> @Override</code>
13	<code> public void onResponse(Call<List<JadwalDokter>> call,</code>
14	<code>Response<List<JadwalDokter>> response) {</code>
15	<code> if (response.body() != null && isAdded()) {</code>
16	<code> JadwalDokters = new ArrayList<>(response.body());</code>
17	<code> AdapterPrakterDokter = new</code>
18	<code>AdapterDetailJadwal(JadwalDokters,getContext());</code>
19	<code>RecyclerView.LayoutManager layoutManager = new</code>
20	<code>LinearLayoutManager(getContext());</code>
21	<code>rcPraktekDokter.setLayoutManager(layoutManager);</code>
22	<code>rcPraktekDokter.setAdapter(AdapterPrakterDokter);</code>
23	<code> progressBar.setVisibility(View.GONE);</code>
24	<code> if (JadwalDokters.size()==0){</code>
25	<code> textKosong.setVisibility(View.VISIBLE);</code>
26	<code> }</code>
27	<code> }</code>
28	<code> }</code>
29	<code> @Override</code>
30	<code> public void onFailure(Call<List<JadwalDokter>> call, Throwable</code>
31	<code>t) {</code>
32	<code> Toast.makeText(getContext(), "Gagal Menghubungkan Ke Basis</code>
33	<code>Data", Toast.LENGTH_SHORT).show();</code>
34	<code> progressBar.setVisibility(View.GONE);</code>
35	<code> progressBar.setVisibility(View.GONE);</code>
36	<code> }</code>
37	<code>});</code>

Tabel 6.18 Penjelasan Kode Program Fungsi `getJadwalDokter()`

Baris	Penjelasan
1-4	Inisialisasi model <code>dokterModel</code> yang nilainya diambil <code>dokterModel</code> yang sudah dimuat pada halaman Daftar Dokter.
5-28	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi <code>onResponse()</code> akan dipanggil dan data yang diambil akan dimasukkan kedalam model <code>response</code> . Selanjutnya dilakukan seleksi apabila <code>response.body</code> tidak bernilai <code>null</code> maka nilainya akan dimasukkan ke dalam <code>arrayList JadwalDokters</code> . Kemudian akan dibuat <code>recyclerView</code> yang datanya berisikan <code>arrayList</code> tersebut.

Tabel 6.18 Penjelasan Kode Program Fungsi `getJadwalDokter()` (lanjutan)

Baris	Penjelasan
29-37	Fungsi <code>onFailure()</code> adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

Tabel 6.19 Kode Program Fungsi `getInfoUmumDokter()`

No	Kode Program
1	<code>dokterModel = (DokterModel)</code>
2	<code>getActivity().getIntent().getExtras().getSerializable("dokterModel</code>
3	<code>");</code>
4	<code>tv[0].setText(dokterModel.getProfil());</code>
5	<code>tv[1].setText(dokterModel.getPengalaman());</code>
6	<code>tv[2].setText(dokterModel.getPendidikan());</code>
7	<code>for (int i = 0; i < btBukaTutup.length; i++) {</code>
8	<code>btBukaTutup[i].setOnClickListener(Informasi_umum_dokter.this);</code>
9	<code>isOpen[i] = false;</code>
10	<code>tv[i].setVisibility(View.GONE);</code>
11	<code>}</code>

Tabel 6.20 Penjelasan Kode Program Fungsi `getInfoUmumDokter()`

Baris	Penjelasan
1-3	Inisialisasi model <code>dokterModel</code> yang nilainya diambil <code>dokterModel</code> yang sudah dimuat pada halaman Daftar Dokter.
4-7	Mengisi text pada array Textview yang berisikan Profil, Pengalaman dan Pendidikan
8-11	Membuat button yang berfungsi untuk membuka dan menutup Textview Profil, Pengalaman dan Pendidikan

6.4.1.3 Kode Program Lihat Denah Objek Layanan Kesehatan

Dalam melihat denah objek layanan kesehatan, kode program akan dijabarkan menjadi 2 fungsi, yaitu : `getKategori()` dan `getListLayananKesehatan()`.

Pertama, fungsi `getKategori()` terdapat pada Tabel 6.21 dan penjelasannya terdapat pada Tabel 6.22. Fungsi pertama ini akan dijalankan apabila pengguna membuka halaman Denah Objek Layanan Kesehatan. Fungsi ini berguna untuk mengambil kategori layanan kesehatan apa saja yang terdapat pada sistem.

Kedua, fungsi `getListLayananKesehatan()` terdapat pada Tabel 6.23 dan penjelasannya terdapat pada Tabel 6.24. Fungsi kedua ini akan dijalankan apabila pengguna membuka halaman Denah Objek Layanan Kesehatan dan telah memilih kategori layanan kesehatan untuk ditampilkan. List layanan kesehatan akan direpresentasikan dengan marker yang berbeda untuk tiap kategorinya.

Tabel 6.21 Kode Program Fungsi `getKategori()`

No	Kode Program
1	<code>Call<List<KategoriModel>> callLakes =</code>
2	<code>apiService.getDaftarKategori();</code>
3	<code>callLakes.enqueue(new Callback<List<KategoriModel>>() {</code>
4	<code> @Override</code>
5	<code> public void onResponse(Call<List<KategoriModel>> call,</code>
6	<code> Response<List<KategoriModel>> response) {</code>
7	<code> progressBar.setVisibility(View.GONE);</code>
8	<code> if (response.body() != null) {</code>
9	<code> List<KategoriModel> kategoriModels = new</code>
10	<code>ArrayList<>(response.body());</code>
11	<code> daftarKategori = new String[kategoriModels.size()];</code>
12	<code> for (int i = 0; i < daftarKategori.length; i++) {</code>
13	<code> daftarKategori[i] =</code>
14	<code> kategoriModels.get(i).getKategori();</code>
15	<code> }</code>
16	<code> checkedKategori = new boolean[daftarKategori.length];</code>
17	<code> for (int i = 0; i < checkedKategori.length; i++) {</code>
18	<code> checkedKategori[i] = false;</code>
19	<code> }</code>
20	<code> //memasukkan data satu persatu bedasarkan kategorinya</code>
21	<code> getData();</code>
22	<code> // Set alert dialog for filter</code>
23	<code> AlertDialog.Builder builder = new</code>
24	<code>AlertDialog.Builder(Denah.this);</code>
25	<code> builder.setTitle("Pilih Kategori");</code>
26	<code> builder.setMultiChoiceItems(daftarKategori,</code>
27	<code> checkedKategori, (dialog, which, isChecked) -> {</code>
28	<code> });</code>
29	<code> builder.setPositiveButton("Lihat Hasil",</code>
30	<code> Denah.this::onClick);</code>
31	<code> dialog = builder.create();</code>
32	<code> dialog.show();</code>
33	<code> }</code>
34	<code> }</code>
35	<code> @Override</code>
36	<code> public void onFailure(Call<List<KategoriModel>> call,</code>
37	<code> Throwable t) {</code>
38	<code> Toast.makeText(Denah.this, "Gagal menghubungkan ke basis</code>
39	<code>data", Toast.LENGTH_LONG).show();</code>
40	<code> }</code>
41	<code>});</code>

Tabel 6.22 Penjelasan Kode Program Fungsi `getKategori()`

Baris	Penjelasan
1-2	Model api dan call berguna untuk menyambungkan ke basis data
3-34	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi <code>onResponse()</code> akan dipanggil dan data yang diambil akan dimasukkan kedalam model response. Selanjutnya dilakukan seleksi apabila <code>response.body()</code> tidak bernilai null maka nilainya akan dimasukkan ke dalam list kategoriModel. Kemudian akan dibuatkan checkbox berdasarkan list tersebut yang nantinya digunakan pengguna dalam memilih kategori.



Tabel 6.22 Penjelasan Kode Program Fungsi `getKategori()` (lanjutan)

Baris	Penjelasan
35-41	Fungsi <code>onFailure()</code> adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

Tabel 6.23 Kode Program Fungsi `getListLayananKesehatan()`

No	Kode Program
1	<code>for (int i = 0; i < checkedKategori.length; i++) {</code>
2	<code> if (checkedKategori[i]) {</code>
3	<code> progressBar.setVisibility(View.VISIBLE);</code>
4	<code> Call<List<LakesModel>> callLakes =</code>
5	<code> apiService.getLakesByKategori(daftarKategori[i], "");</code>
6	<code> callLakes.enqueue(new Callback<List<LakesModel>>() {</code>
7	<code> @Override</code>
8	<code> public void onResponse(Call<List<LakesModel>> call,</code>
9	<code> Response<List<LakesModel>> response) {</code>
10	<code> progressBar.setVisibility(View.GONE)</code>
11	<code> if (response.body().size() > 0) {</code>
12	<code> List<LakesModel> lakesModels = new</code>
13	<code> ArrayList<>(response.body());</code>
14	<code> MarkerOptions options;</code>
15	<code> //menambahkan setiap data ke marker</code>
16	<code> for (int i = 0; i < lakesModels.size(); i++) {</code>
17	<code> LatLng latLng = new</code>
18	<code> LatLng(Double.valueOf(lakesModels.get(i).getLatitude()),</code>
19	<code> Double.valueOf(lakesModels.get(i).getLongitude()));</code>
20	<code> options = new MarkerOptions();</code>
21	<code> options.position(latLng);</code>
22	<code> options.title(lakesModels.get(i).getNama());</code>
23	<code> BitmapDescriptor icon =</code>
24	<code> BitmapDescriptorFactory.fromBitmap(getBitmapFromVectorDrawable(Den</code>
25	<code> ah.this, tentukanIcon(lakesModels.get(i).getKategori()));</code>
26	<code> options.icon(icon);</code>
27	<code> options.snippet(lakesModels.get(i).getKategori());</code>
28	<code> options.title(lakesModels.get(i).getNama());</code>
29	<code> Marker marker = map.addMarker(options);</code>
30	<code> marker.setTag(lakesModels.get(i));</code>
31	<code> markerList.add(marker);</code>
32	<code> }</code>
33	<code> }</code>
34	<code> @Override</code>
35	<code> public void onFailure(Call<List<LakesModel>> call,</code>
36	<code> Throwable t) {</code>
37	<code> Toast.makeText(Denah.this, "Gagal menghubungkan ke</code>
38	<code> basis data", Toast.LENGTH_LONG).show();</code>
39	<code> progressBar.setVisibility(View.GONE);</code>
40	<code> }</code>
41	<code> });</code>
42	<code> }</code>
43	<code> }</code>
44	<code>}</code>
45	
46	
47	

Tabel 6.24 Penjelasan Kode Program Fungsi `getListLayananKesehatan()`

Baris	Penjelasan
1	Perulangan sebanyak kategori yang dipilih pengguna
2	Seleksi kondisi pada variabel boolean <code>checkedKategori</code> apakah bernilai <code>true</code> atau <code>false</code> . Jika bernilai <code>true</code> maka akan dilakukan proses pengambilan data dari kategori tersebut pada basis data.
4-5	Model api dan call berguna untuk menyambungkan ke basis data.
6-37	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi <code>onResponse()</code> akan dipanggil dan data yang diambil akan dimasukkan kedalam model <code>response</code> . Selanjutnya dilakukan seleksi apabila <code>response.body</code> tidak bernilai <code>null</code> maka nilainya akan dimasukkan ke dalam list <code>lakesModels</code> . Selanjutnya dibuat perulangan sebanyak data pada list tersebut. Kemudian dibuat marker pada denah yang datanya berasal dari tiap-tiap <code>lakesModel</code> yang ada dalam list tersebut.
38-47	Fungsi <code>onFailure()</code> adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

6.4.2 Iterasi 1

Pada iterasi 1, terdapat 4 *use case* tambahan yang diimplementasikan menjadi kode program, yaitu Masuk Akun, Daftar Akun, Keluar Akun, dan Penanda.

6.4.2.1 Kode Program Masuk Akun

Fungsi `masukAkun()` berada di dalam Kelas `MasukAkun_activity`. Fungsi ini akan dijalankan ketika pengguna menekan tombol Masuk pada halaman Masuk Akun. Untuk kode program fungsi ini terdapat pada Tabel 6.25 dan penjelasannya terdapat pada Tabel 6.26.

Tabel 6.25 Kode Program Fungsi `masukAkun()`

No	Kode Program
1	<code>progressBar.setVisibility(View.VISIBLE);</code>
2	<code>ApiService apiService =</code>
3	<code>apiConnectionAndCaching.getApi(this).create(ApiService.class);</code>
4	<code>call = apiService.Login(etNama.getText().toString(),</code>
5	<code>etKata_sandi.getText().toString());</code>
6	<code>call.enqueue(new Callback<LoginModel>() {</code>
7	<code> @Override</code>
8	<code> public void onResponse(Call<LoginModel> call,</code>
9	<code> Response<LoginModel> response) {</code>
10	<code> progressBar.setVisibility(View.INVISIBLE);</code>
11	<code> if (response.body() != null) {</code>
12	<code> LoginModel loginModel = response.body();</code>
13	<code> Toast.makeText(LoginActivity.this,</code>
14	<code> loginModel.getMessage(),</code>
15	<code> Toast.LENGTH_LONG).show();</code>
16	<code> if (loginModel.isStatus())</code>

Tabel 6.25 Kode Program Fungsi `masukAkun()` (lanjutan)

No	Kode Program
17	<code>sessionManager=new</code>
18	<code>SessionManager(LoginActivity.this);</code>
19	<code>LoginData loginData =</code>
20	<code>response.body().getLoginData()</code>
21	<code>sessionManager.createLoginSession(loginData);</code>
22	<code>LoginActivity.this.startActivity(new</code>
23	<code>Intent(LoginActivity.this,</code>
24	<code>BerandaActivity.class));</code>
25	<code>LoginActivity.this.finish();</code>
26	<code>}</code>
27	<code>}</code>
28	<code>}</code>
29	<code>@Override</code>
30	<code>public void onFailure(Call<LoginModel> call, Throwable t) {</code>
31	<code>progressBar.setVisibility(View.INVISIBLE);</code>
32	<code>Toast.makeText(LoginActivity.this, "Gagal Menghubungkan</code>
33	<code>Ke Basis Data", Toast.LENGTH_LONG).show();</code>
34	<code>}</code>
35	<code>});</code>

Tabel 6.26 Penjelasan Kode Program Fungsi `masukAkun()`

Baris	Penjelasan
2-7	Model api dan call berguna untuk menyambungkan ke basis data
8-28	Fungsi <code>onResponse()</code> adalah fungsi yang akan dipanggil saat berhasil menyambungkan ke basis data, kemudian data yang diambil akan dimasukkan kedalam model response. Selanjutnya dilakukan seleksi apabila <code>response.body</code> tidak bernilai <code>null</code> maka nilainya akan dimasukkan ke dalam model loginModel. Kemudian akan ditampilkan pesan dari api apakah proses login berhasil atau tidak. Selanjutnya dilakukan seleksi apabila nilai kembalian fungsi <code>isStatus()</code> pada loginModel bernilai <code>true</code> , maka data login akan disimpan pada session dan halaman akan berpindah menuju beranda.
29-35	Fungsi <code>onFailure()</code> adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

6.4.2.2 Kode Program Daftar Akun

Fungsi `daftarAkun()` berada di dalam Kelas `daftarActivity`. Fungsi ini akan dijalankan ketika pengguna menekan tombol Daftar pada halaman Daftar Akun. Untuk kode program fungsi ini terdapat pada Tabel 6.27 dan penjelasannya terdapat pada Tabel 6.28.

Tabel 6.27 Kode Program Fungsi `daftarAkun()`

No	Kode Program
1	<code>if (etNama.getText().toString().equals("") </code>
2	<code>etKata_sandi.getText().toString().equals("")) {</code>
3	<code>Toast.makeText(this, "Nama / Kata Sandi tidak boleh kosong",</code>



Tabel 6.27 Kode Program Fungsi `daftarAkun()` (Lanjutan)

No	Kode Program
4	<code>Toast.LENGTH_SHORT).show();</code>
5	<code>} else if</code>
6	<code>(!etKata_sandi.getText().toString().equals(etKata_sandi2.getText()</code>
7	<code>.toString())) {</code>
8	<code> Toast.makeText(this, "Kata Sandi Tidak Sama",</code>
9	<code>Toast.LENGTH_SHORT).show();</code>
10	<code>} else {</code>
11	<code> ApiService apiService =</code>
12	<code>apiConnectionAndCaching.getApi(this).create(ApiService.class);</code>
13	<code> call = apiService.Daftar(etNama.getText().toString(),</code>
14	<code>etKata_sandi.getText().toString());</code>
15	<code> call.enqueue(new Callback<DaftarModel>() {</code>
16	<code> @Override</code>
17	<code> public void onResponse(Call<DaftarModel> call,</code>
18	<code>Response<DaftarModel> response) {</code>
19	<code> DaftarModel daftarModel = response.body();</code>
20	<code> Toast.makeText(DaftarActivity.this,</code>
21	<code>daftarModel.getMessage(), Toast.LENGTH_LONG).show();</code>
22	<code> if (daftarModel.isStatus()) {</code>
23	<code> Intent intent = new Intent(DaftarActivity.this,</code>
24	<code>LoginActivity.class);</code>
25	<code> startActivity(intent);</code>
26	<code> }</code>
27	<code> }</code>
28	<code> @Override</code>
29	<code> public void onFailure(Call<DaftarModel> call, Throwable t)</code>
30	<code>{</code>
31	<code> Toast.makeText(DaftarActivity.this, "Gagal</code>
32	<code>Menghubungkan Ke Basis Data", Toast.LENGTH_LONG).show();</code>
33	<code> }</code>
34	<code> });</code>
35	<code>}</code>

Tabel 6.28 Penjelasan Kode Program Fungsi `daftarAkun()`

Baris	Penjelasan
1-4	Seleksi kondisi untuk mengecek apakah nama / kata sandi kosong, ketika kosong maka akan ditampilkan pesan "Nama / Kata Sandi tidak boleh kosong".
5-8	Seleksi kondisi untuk mengecek apakah text pada kolom Ulangi Kata Sandi apakah sudah sama dengan text pada kolom Kata Sandi, jika berbeda maka akan ditampilkan pesan Kata Sandi Tidak Sama.
10-28	Ketika kedua seleksi kondisi sebelumnya bernilai <code>false</code> maka sistem akan disambungkan ke basis data, jika sambungan berhasil maka fungsi <code>onResponse()</code> akan dipanggil. Data yang diambil akan dimasukkan kedalam model response. Selanjutnya dilakukan seleksi apabila <code>response.body</code> tidak bernilai <code>null</code> maka nilainya akan dimasukkan ke dalam model registrasiModel. Kemudian akan ditampilkan pesan dari api apakah proses login berhasil atau tidak. Selanjutnya dilakukan seleksi apabila nilai kembalian fungsi <code>isStatus()</code> pada <code>daftarModel</code>

Tabel 6.28 Penjelasan Kode Program Fungsi `daftarAkun()` (lanjutan)

Baris	Penjelasan
29-35	bernilai true, maka data pendaftaran akan disimpan pada basis data dan halaman akan berpindah menuju halaman Masuk. Fungsi <code>onFailure()</code> adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

6.4.2.3 Kode Program Daftar Akun

Fungsi `keluarAkun()` berada di dalam Kelas `penandaActivity`. Fungsi ini akan dijalankan ketika pengguna menekan tombol Keluar pada halaman Penanda. Untuk kode program fungsi ini terdapat pada Tabel 6.29 dan penjelasannya terdapat pada Tabel 6.30.

Tabel 6.29 Kode Program Fungsi `keluarAkun()`

No	Kode Program
1	<code>sessionManager.logoutSession();</code>
2	<code>finish();</code>

Tabel 6.30 Penjelasan Kode Program Fungsi `keluarAkun()`

Baris	Penjelasan
1	Memanggil fungsi yang berguna untuk menghapus sesi yang sedang berjalan
2	Kode untuk menghentikan Activity yang sedang berjalan sekarang, dan kembali ke halaman awal

6.4.2.4 Kode Program Penanda

Dalam melihat penanda, kode program akan dijabarkan menjadi 2 fungsi, yaitu `getListDokterDitandai()` dan `getListLakesDitandai()`.

Pertama, fungsi `getListDokterDitandai()` terdapat pada Tabel 6.31 dan penjelasannya terdapat pada Tabel 6.32. Fungsi pertama ini akan dijalankan apabila pengguna berada pada halaman Penanda dan memilih opsi Dokter.

Kedua, fungsi `getListLakesDitandai()` terdapat pada Tabel 6.33 dan penjelasannya terdapat pada Tabel 6.34. Fungsi kedua ini akan dijalankan apabila pengguna berada pada halaman Penanda dan memilih opsi Layanan Kesehatan.

Tabel 6.31 Kode Program Fungsi `getListDokterDitandai()`

No	Kode Program
1	<code>List<DokterModel> listDokter = new ArrayList<>();</code>
2	<code>final AdapterDokter[] adapterDokter = {new</code>
3	<code>AdapterDokter(listDokter)};</code>
4	<code>rcDaftar.setAdapter(adapterDokter[0]);</code>
5	<code>progressBar.setVisibility(View.VISIBLE);</code>

Tabel 6.31 Kode Program Fungsi getListDokterDitandai () (lanjutan)

No	Kode Program
6	<code>callDokter =</code>
7	<code>apiService.getDokterByAkunId(sessionManager.getAkunDetail().get(Se</code>
8	<code>ssionManager.ID));</code>
9	<code>callDokter.enqueue(new Callback<List<DokterModel>>() {</code>
10	<code>@Override</code>
11	<code>public void onResponse(Call<List<DokterModel>> call,</code>
12	<code>Response<List<DokterModel>> response) {</code>
13	<code>if (response.body() != null) {</code>
14	<code>//cek isi dari response body</code>
15	<code>progressBar.setVisibility(View.GONE);</code>
16	<code>if (!response.body().isEmpty()) {</code>
17	<code>tvKosong.setVisibility(View.GONE);</code>
18	<code>listDokter.addAll(response.body());</code>
19	<code>RecyclerView.LayoutManager layoutManager = new</code>
20	<code>LinearLayoutManager(PenandaActivity.this);</code>
21	<code>rcDaftar.setLayoutManager(layoutManager);</code>
22	<code>adapterDokter[0] = new AdapterDokter(listDokter);</code>
23	<code>rcDaftar.setAdapter(adapterDokter[0]);</code>
24	<code>} else {</code>
25	<code>tvKosong.setVisibility(View.VISIBLE);</code>
26	<code>}</code>
27	<code>} else {</code>
28	<code>progressBar.setVisibility(View.GONE);</code>
29	<code>}</code>
30	<code>}</code>
31	<code>@Override</code>
32	<code>public void onFailure(Call<List<DokterModel>> call, Throwable</code>
33	<code>t) {</code>
34	<code>Toast.makeText(PenandaActivity.this, "Gagal Menghubungkan</code>
35	<code>ke Basis Data", Toast.LENGTH_LONG).show();</code>
36	<code>progressBar.setVisibility(View.GONE);</code>
37	<code>}</code>
38	<code>});</code>

Tabel 6.32 Penjelasan Kode Program Fungsi getListDokterDitandai ()

Baris	Penjelasan
1-2	Instansiasi model
3-4	Mengatur adapter dan RecyclerView untuk daftar dokter
6-8	Model api dan call berguna untuk menyambungkan ke basis data, pada request api tersebut diberi masukan parameter berupa id pengguna yang sedang masuk ke sistem agar nilai kembalian nantinya berupa list dokter yang pengguna tersebut telah tandai.
9-30	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi onResponse() akan dipanggil dan data yang diambil akan dimasukkan kedalam model response. Selanjutnya dilakukan seleksi apabila response.body tidak bernilai null maka nilainya akan dimasukkan ke dalam list dokterModel. Kemudian akan dibuat recyclerView yang datanya berisikan list tersebut
31-38	Fungsi onFailure() adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

Tabel 6.33 Kode Program Fungsi getListLakesDitandai ()

No	Kode Program
1	List<LakesModel> listLakes = new ArrayList<>();
2	final AdapterLakes[] adapterLakes = {new
3	AdapterLakes(listLakes)};
4	rcDaftar.setAdapter(adapterLakes[0]);
5	progressBar.setVisibility(View.VISIBLE);
6	callLakes =
7	apiService.getLakesByAkunId(sessionManager.getAkunDetail().g
8	et(SessionManager.ID));
9	callLakes.enqueue(new Callback<List<LakesModel>>() {
10	@Override
11	public void onResponse(Call<List<LakesModel>> call,
12	Response<List<LakesModel>> response) {
13	if (response.body() != null) {
14	//cek isi dari response body
15	progressBar.setVisibility(View.GONE);
16	if (!response.body().isEmpty()) {
17	tvKosong.setVisibility(View.GONE);
18	listLakes.addAll(response.body());
19	RecyclerView.LayoutManager layoutManager = new
20	LinearLayoutManager(PenandaActivity.this);
21	rcDaftar.setLayoutManager(layoutManager);
22	adapterLakes[0] = new AdapterLakes(listLakes);
23	rcDaftar.setAdapter(adapterLakes[0]);
24	} else {
25	tvKosong.setVisibility(View.VISIBLE);
26	}
27	} else {
28	Toast.makeText(PenandaActivity.this, "Gagal
29	Menghubungkan ke Basis Data", Toast.LENGTH_LONG).show();
30	progressBar.setVisibility(View.GONE);
31	}
32	}
33	@Override
34	public void onFailure(Call<List<LakesModel>> call, Throwable
35	t) {
36	progressBar.setVisibility(View.GONE);
37	Toast.makeText(PenandaActivity.this, "Gagal Menghubungkan ke Basis
38	Data", Toast.LENGTH_LONG).show();
39	}
40	}
41	});

Tabel 6.34 Penjelasan Kode Program Fungsi getListLakesDitandai ()

Baris	Penjelasan
1-2	Instansiasi model
3-4	Mengatur adapter dan RecyclerView untuk daftar lakes.
6-8	Model api dan call berguna untuk menyambungkan ke basis data, pada request api tersebut diberi masukan parameter berupa id pengguna yang sedang masuk ke sistem agar nilai kembalian nantinya berupa list lakes yang pengguna tersebut telah tandai.

Tabel 6.35 Penjelasan Kode Program Fungsi `getListLakesDitandai()` (lanjutan)

Baris	Penjelasan
9-32	Sistem disambungkan ke basis data, jika sambungan berhasil maka fungsi <code>onResponse()</code> akan dipanggil dan data yang diambil akan dimasukkan kedalam model response. Selanjutnya dilakukan seleksi apabila <code>response.body</code> tidak bernilai <code>null</code> maka nilainya akan dimasukkan ke dalam <code>list_lakesModel</code> . Kemudian akan dibuat <code>recyclerView</code> yang datanya berisikan list tersebut
33-41	Fungsi <code>onFailure()</code> adalah fungsi yang akan dipanggil saat gagal menyambungkan ke basis data. Berisi perintah untuk menampilkan pesan "Gagal Menghubungkan Ke Basis Data".

6.5 Implementasi Antarmuka

Implementasi antarmuka dibuat berdasarkan perancangan antarmuka yang telah dibuat sebelumnya. Pada Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat, implementasi antarmuka dikembangkan menggunakan bahasa markup XML dengan komponen view yang didukung oleh Android Studio.

6.5.1 Iterasi 0

Implementasi antarmuka iterasi 0 berisikan 8 kategori Implementasi yaitu antarmuka halaman *Splash Screen*, antarmuka halaman beranda, antarmuka halaman Daftar Spesialis, antarmuka halaman Daftar Dokter, antarmuka halaman Detail Dokter, antarmuka halaman Daftar Objek Layanan Kesehatan, antarmuka halaman Detail Objek Layanan Kesehatan, dan antarmuka halaman Denah Objek Layanan Kesehatan.

Pada kategori antarmuka halaman Daftar Spesialis dan antarmuka halaman Daftar Objek Layanan Kesehatan terdapat antarmuka *pop-up* untuk menyaring daftar yang ditampilkan. Kemudian pada kategori antarmuka halaman Detail Dokter dan antarmuka halaman Detail Objek Layanan Kesehatan terdapat 3 macam antarmuka di dalamnya, yaitu 1 antarmuka untuk bagian atas halaman dan 2 antarmuka untuk bagian bawah halaman.

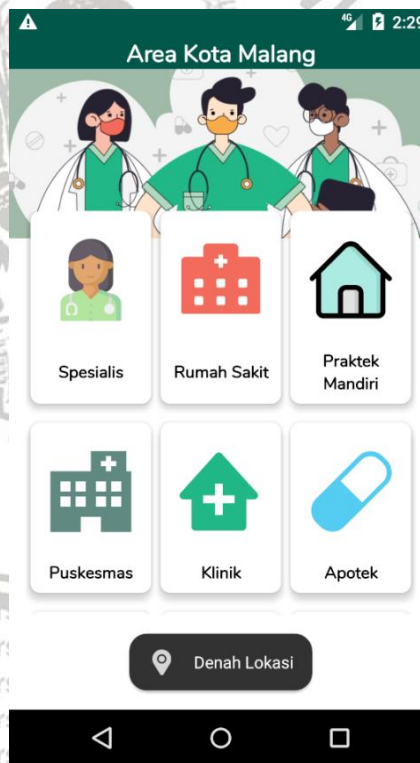
6.5.1.1 Halaman *Splash Screen*

Halaman *Splashscreen* dapat dilihat dalam Gambar 6.11. Pada halaman tersebut terdapat latar hijau dan logo di tengahnya guna memperjelas identitas aplikasi saat dibuka aktor.



Gambar 6.11 Antarmuka Halaman *Splash Screen*

6.5.1.2 Halaman Beranda



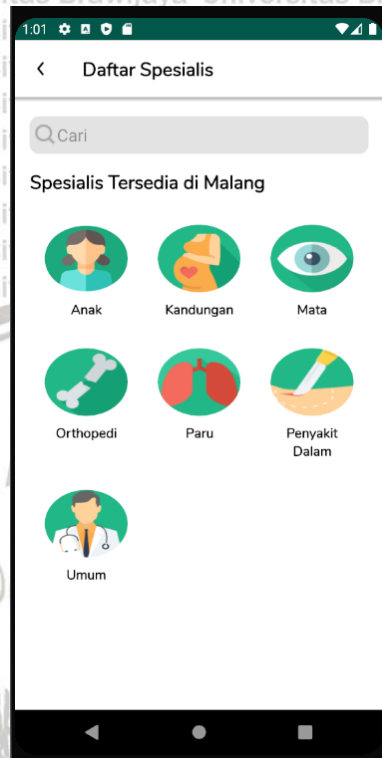
Gambar 6.12 Antarmuka halaman Beranda

Implementasi Antarmuka halaman Beranda terdapat dalam Gambar 6.12.

Pada halaman ini terdapat tombol icon-icon kategori objek layanan kesehatan dan tombol Denah Lokasi, tombol icon-icon kategori objek layanan kesehatan jika ditekan maka halaman akan berpindah menuju halaman daftar objek layanan kesehatan dengan kategori terpilih, terkecuali untuk kategori Spesialis maka

halaman akan berpindah menuju halaman Daftar Spesialis. sedangkan tombol Denah Lokasi jika ditekan maka halaman akan berpindah menuju halaman Denah Lokasi Objek Layanan Kesehatan.

6.5.1.3 Halaman Daftar Spesialis

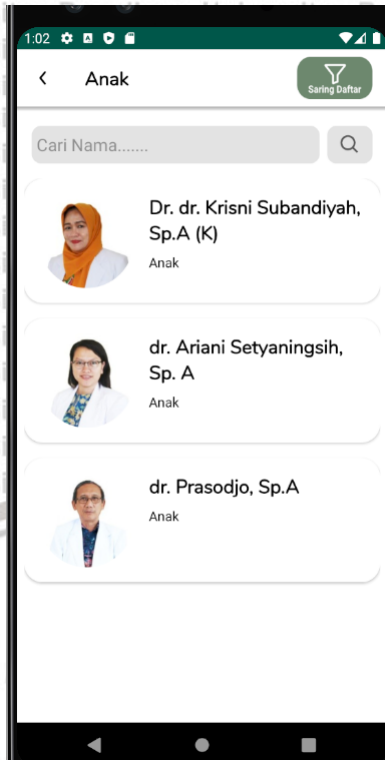


Gambar 6.13 Implementasi Antarmuka Halaman Daftar Spesialis

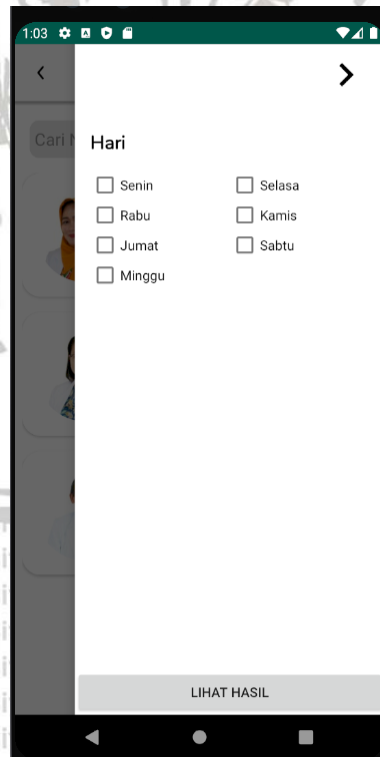
Implementasi Antarmuka halaman Daftar Spesialis dapat dilihat dalam Gambar 6.13. Dalam halaman tersebut terdapat kolom Cari yang berguna sebagai alat pencarian nama spesialis, Setelah itu terdapat daftar spesialis yang tersedia beserta dengan icon yang merepresentasikannya, saat icon tersebut ditekan, maka halaman akan berpindah ke halaman Daftar Dokter.

6.5.1.4 Halaman Daftar Dokter

Implementasi Antarmuka halaman Daftar Dokter dapat dilihat dalam Gambar 6.19. Dalam halaman tersebut terdapat kolom Cari Nama yang berguna sebagai alat pencarian nama dokter, Setelah itu terdapat objek kartu dokter yang berisikan foto, nama, dan Spesialisasi dokter guna merepresentasikan identitas pada daftar dokter. Ketika objek kartu tersebut ditekan maka halaman akan berpindah menuju halaman Detail Dokter. Pada halaman ini terdapat juga tombol saring daftar yang akan memunculkan *pop-up* Saring Daftar Dokter guna melakukan penyaringan pada daftar dokter yang ada.



Gambar 6.14 Implementasi Antarmuka Halaman Daftar Dokter

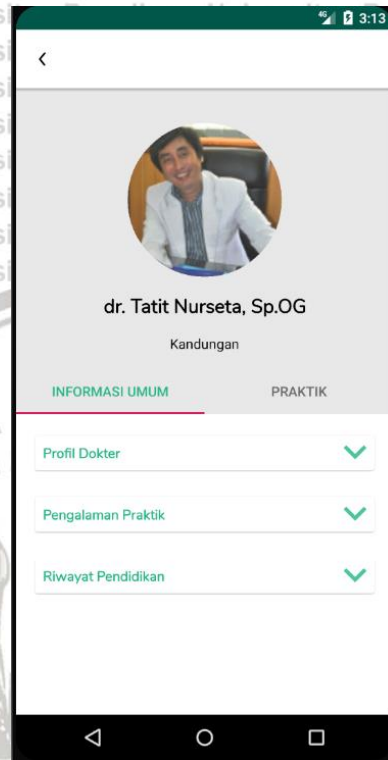


Gambar 6.15 Antarmuka *Pop-up* Saring Daftar Dokter

Implementasi Antarmuka *Pop-up* Saring Daftar Dokter dapat dilihat dalam Gambar 6.20. Dalam Gambar 6.20. Pada halaman tersebut terdapat *checkbox* pilihan hari yang dapat dicentang. Setelah pilihan hari dicentang, tombol Lihat Hasil ditekan dan

proses penyaringan akan dilakukan terhadap daftar dokter yang ada berdasarkan hari praktiknya sesuai dengan centang pada *checkbox*.

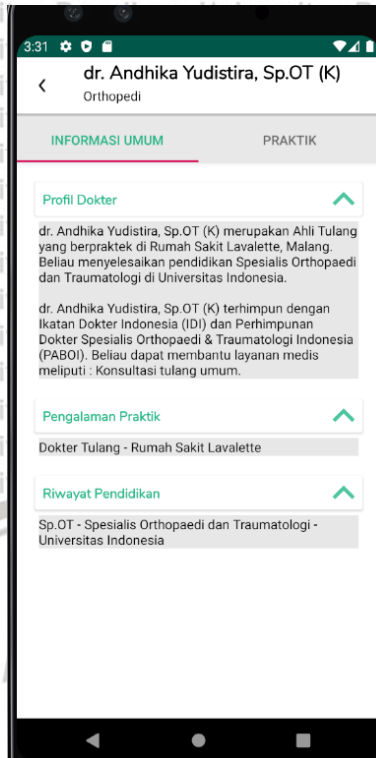
6.5.1.5 Halaman Detail Dokter



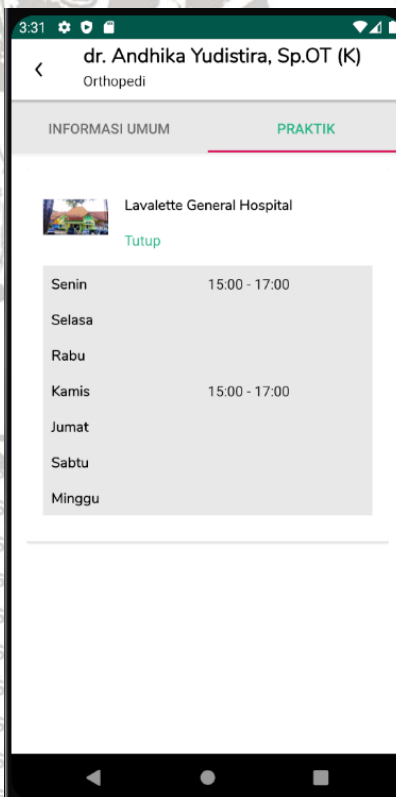
Gambar 6.16 Antarmuka Halaman Detail Dokter

Implementasi antarmuka halaman Detail Dokter dapat dilihat dalam Dalam Gambar 6.16. Pada halaman Detail Dokter terdapat tombol untuk kembali ke halaman sebelumnya, foto dokter, nama dokter, nama spesialis dokter, tab Informasi Umum, dan tab Praktik

Selanjutnya Implementasi Antarmuka halaman Detail Dokter Tab Informasi Umum dapat dilihat dalam Dalam Gambar 6.17. Pada tab tersebut terdapat kolom Informasi Profil Dokter, Pengalaman Praktik, dan Riwayat Pendidikan serta tombol untuk membuka dan menutup setiap kolomnya.



Gambar 6.17 Antarmuka Halaman Detail Dokter Tab Informasi Umum

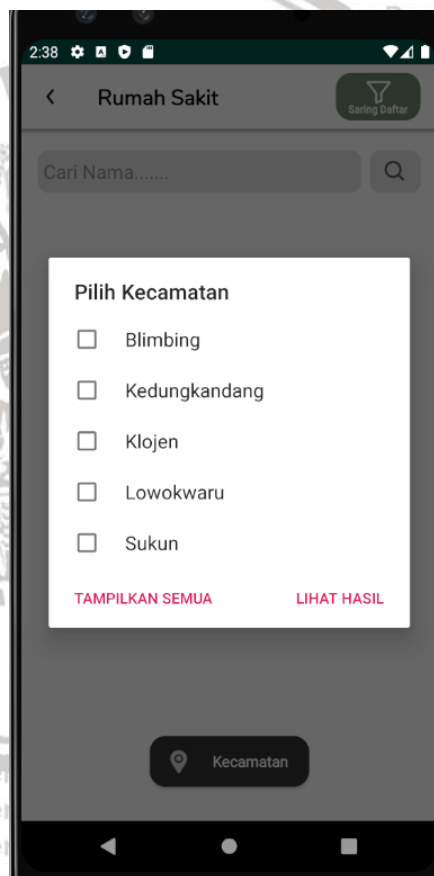


Gambar 6.18 Antarmuka Halaman Detail Dokter Tab Praktik

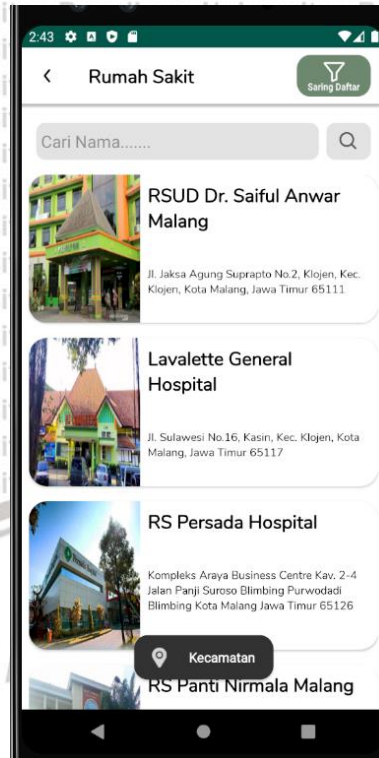
Implementasi Antarmuka halaman Detail Dokter Tab Praktik dapat dilihat dalam Dalam Gambar 6.18. Pada tab tersebut terdapat objek kartu yang berisikan nama dan foto objek layanan kesehatan, serta jadwal praktik dokter pada objek layanan kesehatan tersebut.

6.5.1.6 Halaman Daftar Objek Layanan Kesehatan

Implementasi antarmuka *Pop-up* Pilih Kecamatan dapat dilihat dalam Dalam Gambar 6.19. Pada *pop-up* tersebut terdapat *checkbox* pilihan kecamatan yang dapat dicentang. Setelah pilihan kecamatan dicentang, tombol Lihat Hasil ditekan dan proses penyaringan akan dilakukan terhadap daftar objek layanan kesehatan yang ada berdasarkan kecamatan yang dipilih. Selanjutnya terdapat juga tombol Tampilkan Semua, tombol ini berguna untuk menampilkan objek layanan kesehatan dari semua kecamatan.



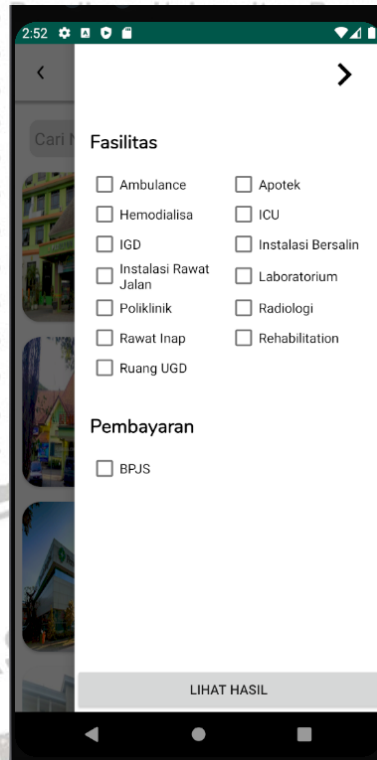
Gambar 6.19 Antarmuka Pop-up Pilih Kecamatan



Gambar 6.20 Antarmuka Halaman Daftar Objek Layanan Kesehatan

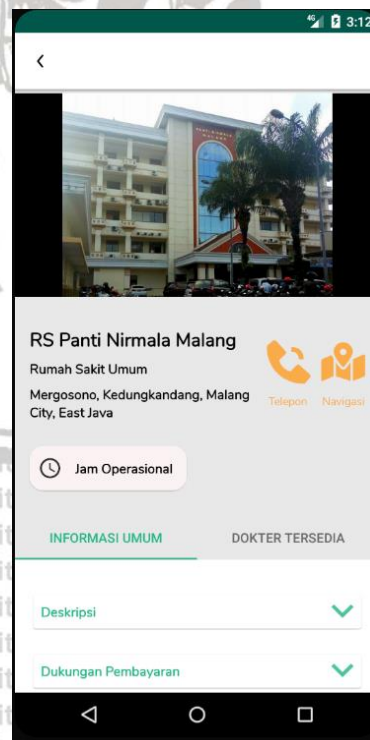
Implementasi antarmuka halaman Daftar Objek Layanan Kesehatan dapat dilihat dalam Dalam Gambar 6.20. Pada halaman tersebut terdapat kolom Cari Nama yang berguna sebagai alat pencarian nama objek layanan kesehatan, Setelah itu terdapat objek kartu layanan kesehatan yang berisikan foto, nama, dan alamat layanan kesehatan guna merepresentasikan identitasnya. Ketika objek kartu tersebut ditekan maka halaman akan berpindah menuju halaman Detail Layanan kesehatan. Pada halaman ini terdapat juga tombol saring daftar yang akan memunculkan *pop-up* Saring Daftar Layanan Kesehatan guna melakukan penyaringan pada daftar objek layanan kesehatan yang ada.

Kemudian, Implementasi Antarmuka *Pop-up* Saring Daftar Daftar Objek Layanan Kesehatan dapat dilihat dalam Dalam Gambar 6.21. Pada halaman tersebut terdapat *checkbox* pilihan fasilitas dan pembayaran yang dapat dicentang. Setelah pilihan pada *checkbox* dicentang, tombol Lihat Hasil ditekan dan proses penyaringan akan dilakukan terhadap daftar objek layanan kesehatan yang ada.



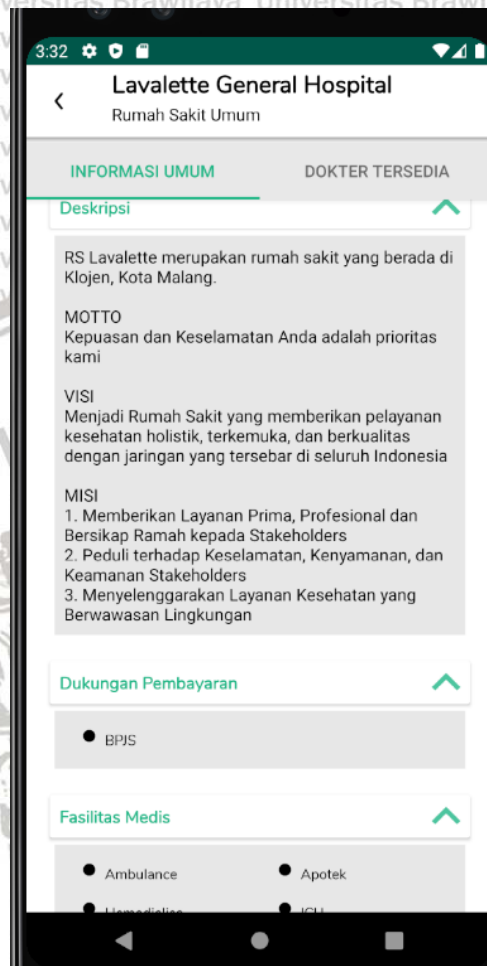
Gambar 6.21 Antarmuka *Pop-up Saring Daftar Objek Layanan Kesehatan*

6.5.1.7 Halaman Detail Objek Layanan Kesehatan



Gambar 6.22 Antarmuka Halaman Detail Objek Layanan Kesehatan

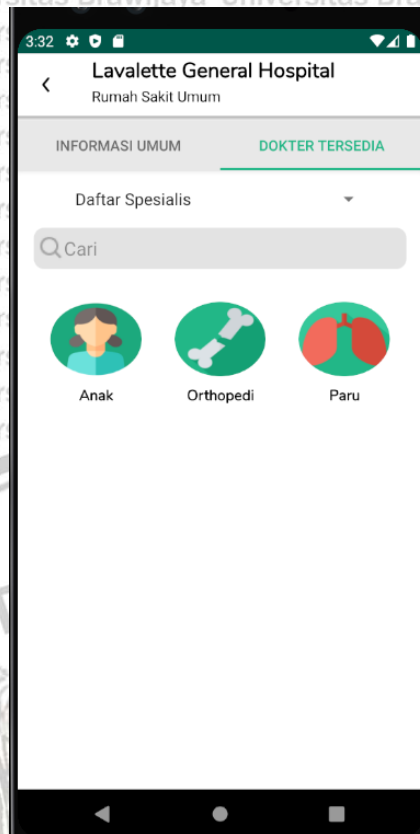
Implementasi antarmuka halaman Detail Objek Layanan Kesehatan dapat dilihat dalam Gambar 6.22. Pada halaman Detail Objek Layanan Kesehatan terdapat tombol untuk kembali ke halaman sebelumnya, foto dokter, nama dokter, nama spesialis dokter, tab Informasi Umum, dan tab Dokter Tersedia.



Gambar 6.23 Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Informasi Umum

Implementasi Antarmuka halaman Detail Objek Layanan Kesehatan Tab Informasi Umum dapat dilihat dalam Gambar 6.23. Pada tab tersebut terdapat kolom Deskripsi, Dukungan Pembayaran, dan Fasilitas Medis serta tombol untuk membuka dan menutup setiap kolomnya.

Kemudian Implementasi Antarmuka halaman Detail Objek Layanan Kesehatan Tab Dokter Tersedia dapat dilihat dalam Gambar 6.24. Pada tab tersebut terdapat objek kartu yang berisikan nama dan foto daftar spesialis yang tersedia di layanan kesehatan tersebut. Ketika objek kartu tersebut ditekan, maka halaman akan berpindah menuju halaman Daftar Dokter dengan kategori spesialis yang dipilih.

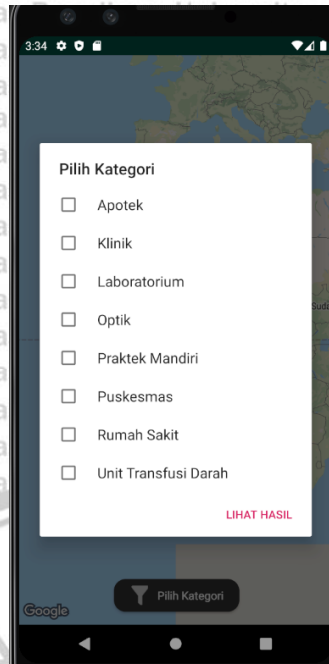


Gambar 6.24 Antarmuka Halaman Detail Objek Layanan Kesehatan Tab Dokter Tersedia

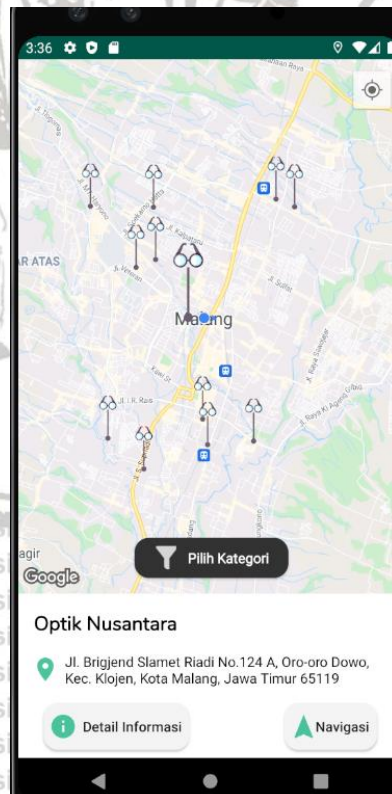
6.5.1.8 Halaman Denah Objek Layanan Kesehatan

Implementasi Antarmuka *Pop-up* Pilih Kategori Halaman Denah Objek Layanan Kesehatan dapat dilihat dalam Dalam Gambar 6.25. Pada halaman tersebut terdapat *checkbox* pilihan kategori yang dapat dicentang. Setelah pilihan kategori dicentang, tombol Lihat Hasil ditekan dan proses penyaringan akan dilakukan terhadap daftar objek layanan kesehatan yang ada.

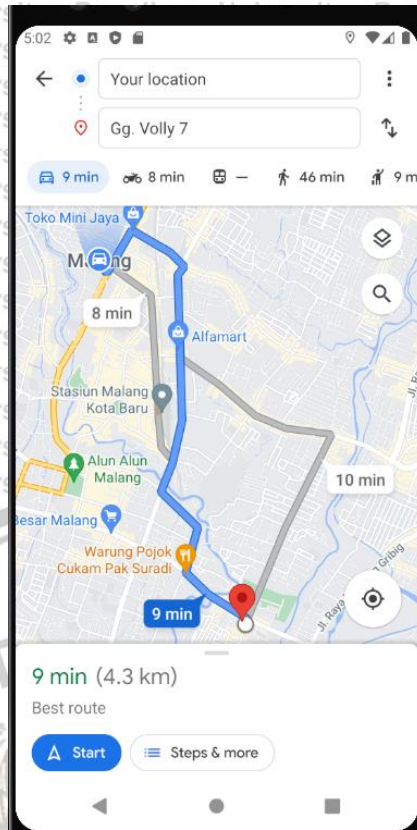
Implementasi antarmuka halaman Denah Objek Layanan Kesehatan dapat dilihat dalam Dalam Gambar 6.26. Pada halaman tersebut terdapat tombol Lokasi pengguna, *icon marker* dan tombol Pilih Kategori. Selanjutnya ketika pengguna menekan salah satu marker, maka akan muncul *pop-up* di bagian bawah layar yang berisikan nama objek layanan kesehata, alamat, tombol Detail Informasi yang jika ditekan akan memindahkan halaman menuju halaman Detail Objek Layanan Kesehatan, dan tombol Navigasi yang jika ditekan akan memindahkan halaman menuju halaman Navigasi pada aplikasi Google Maps seperti terlihat dalam Gambar 6.27.



Gambar 6.25 Antarmuka *Pop-up* Pilih Kategori Halaman Denah Objek Layanan Kesehatan



Gambar 6.26 Antarmuka Halaman Denah Objek Layanan Kesehatan



Gambar 6.27 Antarmuka Halaman Navigasi Aplikasi Google Maps

6.5.2 Iterasi 1

Implementasi antarmuka iterasi 1 berisikan penambahan 3 antarmuka yaitu antarmuka halaman Masuk Akun, antarmuka halaman Daftar Akun dan antarmuka halaman Penanda. Selanjutnya juga terdapat perubahan pada halaman Beranda, halaman Detail Dokter dan halaman Detail Objek Layanan Kesehatan.

6.5.2.1 Halaman Beranda

Implementasi Antarmuka halaman Beranda sebelum masuk ke dalam akun terdapat dalam Gambar 6.28 dan setelah masuk ke dalam akun terdapat dalam Gambar 6.29. Perbedaannya terletak pada tombol yang awalnya tombol Akun menjadi tombol Penanda. Selain perbedaan tombol tersebut, terdapat tombol lain yang sama yaitu tombol icon-icon kategori objek layanan kesehatan dan tombol Denah Lokasi, tombol icon-icon kategori objek layanan kesehatan jika ditekan maka halaman akan berpindah menuju halaman daftar objek layanan kesehatan dengan kateogri terpilih, terkecuali untuk kategori dokter maka halaman akan berpindah menuju halaman Daftar Spesialis. sedangkan tombol Denah Lokasi jika ditekan maka halaman akan berpindah menuju halaman Denah Lokasi Objek Layanan Kesehatan.

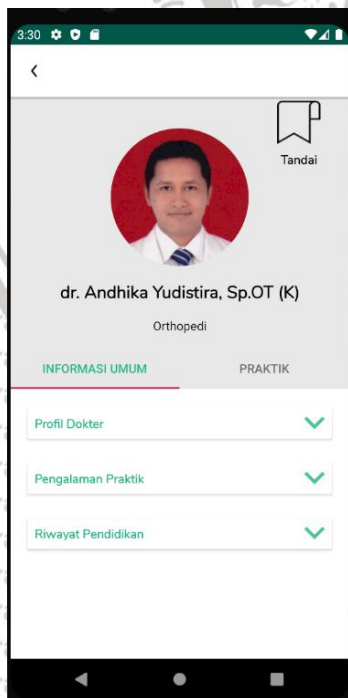


Gambar 6.29 Implementasi Antarmuka Halaman Beranda Belum Masuk Akun

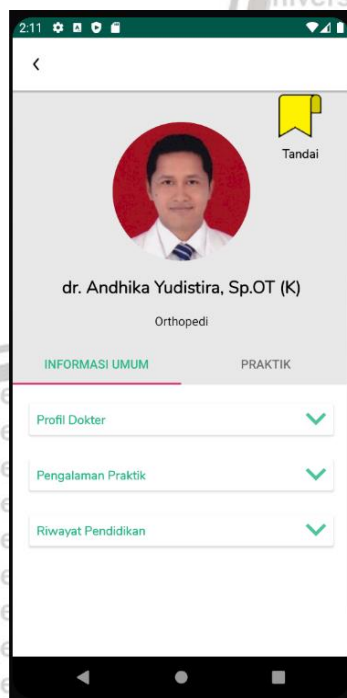


Gambar 6.28 Implementasi Antarmuka Halaman Beranda Sudah Masuk Akun

6.5.2.2 Halaman Detail Dokter



Gambar 6.30 Antarmuka Halaman Detail Dokter Tidak Ditandai



Gambar 6.31 Antarmuka Halaman Detail Dokter Ditandai

Implementasi antarmuka halaman Detail Dokter yang sedang tidak ditandai dapat dilihat dalam Dalam Gambar 6.31 dan yang sedang ditandai dapat dilihat dalam Gambar 6.30. Pada halaman Detail Dokter terdapat tombol untuk kembali ke halaman sebelumnya, foto dokter, nama dokter, nama spesialis dokter, tab Informasi Umum, tab Praktik, serta tombol Tandai dengan icon transparan apabila sedang tidak ditandai dan tombol berwarna kuning apabila sedang ditandai.

6.5.2.3 Halaman Detail Objek Layanan Kesehatan



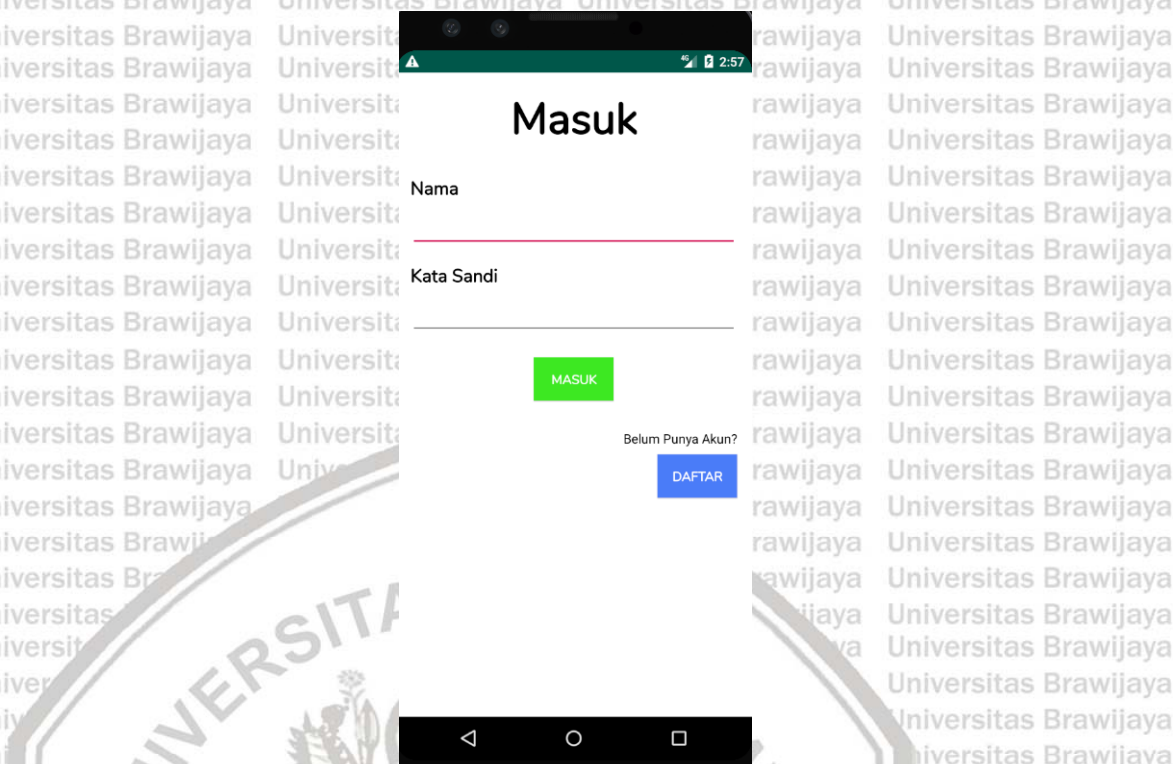
Gambar 6.32 Antarmuka Halaman Detail Objek Layanan Kesehatan Tidak Ditandai



Gambar 6.33 Antarmuka Halaman Detail Objek Layanan Kesehatan Ditandai

Implementasi antarmuka halaman Detail Objek Layanan Kesehatan yang sedang tidak ditandai dapat dilihat dalam Dalam Gambar 6.32 dan yang sedang ditandai dapat dilihat dalam Gambar 6.33. Pada halaman Detail Objek Layanan Kesehatan terdapat tombol untuk kembali ke halaman sebelumnya, foto dokter, nama dokter, nama spesialis dokter, tab Informasi Umum, tab Dokter Tersedia, serta tombol Tandai dengan icon transparan apabila sedang tidak ditandai dan tombol berwarna kuning apabila sedang ditandai.

6.5.2.4 Halaman Masuk Akun

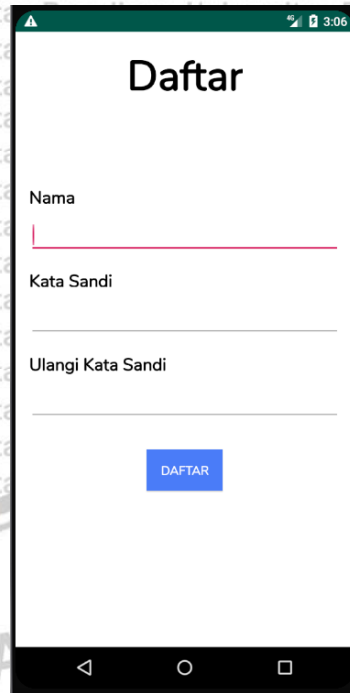


Gambar 6.34 Implementasi Antarmuka Halaman Masuk Akun

Implementasi Antarmuka halaman Masuk Akun dapat dilihat dalam Gambar 6.34. Pada halaman tersebut terdapat dua kolom yaitu kolom Nama dan kolom Kata Sandi disertai tombol Masuk untuk proses Masuk Akun. Pada bagian bawah layar telah disediakan juga tombol Daftar yang akan membuat halaman berpindah ke halaman Daftar Akun.

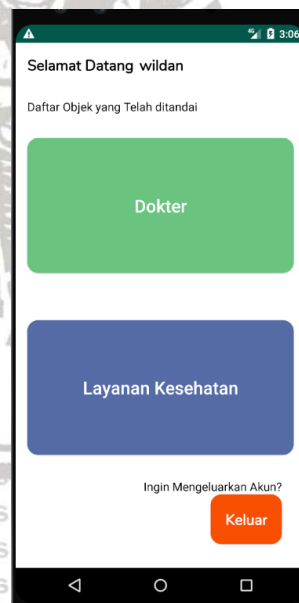
6.5.2.5 Halaman Daftar Akun

Implementasi Antarmuka halaman Daftar Akun dapat dilihat dalam Gambar 6.35. Pada halaman tersebut terdapat form dengan tiga kolom yaitu Nama, Kata Sandi dan Ulangi Kata Sandi beserta tombol Daftar yang berguna untuk proses pendaftaran akun. Kolom nama diisi dengan nama yang ingin didaftarkan, kolom kata sandi diisi dengan kata sandi pilihan guna keamanan, sedangkan kolom Ulangi Kata Sandi diisi sama dengan kata sandi, hal ini bertujuan agar tidak terjadi kesalahan penulisan kata sandi dikarenakan saat proses mengetik kata sandi, karakter akan berubah menjadi bintang. Setelah data diisikan maka ditekan tombol Daftar, dan isi dari form tersebut akan dikirimkan ke basis data. Setelah berhasil melakukan pendaftaran maka halaman akan berpindah ke halaman Masuk Akun.



Gambar 6.35 Implementasi Antarmuka Halaman Daftar Akun

6.5.2.6 Halaman Penanda



Gambar 6.36 Antarmuka Halaman Penanda

Implementasi antarmuka halaman Penanda dapat dilihat dalam Dalam Gambar 6.35. Pada halaman tersebut terdapat ucapan selamat datang yang disertai penyebutan nama pengguna. Selanjutnya terdapat tombol dokter dan layanan kesehatan. Apabila tombol dokter ditekan maka akan ditampilkan

halaman yang berisi daftar dokter yang telah ditandai, begitupun jika tombol Layanan Kesehatan ditekan maka akan ditampilkan halaman yang berisi daftar objek layanan kesehatan yang telah ditandai. Terakhir terdapat tombol Keluar yang berguna untuk mengeluarkan akun pengguna dari sistem.

6.6 Umpan Balik Pemangku Kepentingan

6.6.1 Iterasi 0

Pada iterasi 0, pemangku kepentingan menyatakan bahwa kebutuhan yang dirumuskan sebelumnya sudah baik dan tidak membutuhkan perubahan, namun dibutuhkan penambahan kebutuhan baru yaitu kebutuhan guna personalisasi pengguna. Hasil wawancara pemangku kepentingan iterasi 0 terlampir pada Lampiran A.3

6.6.2 Iterasi 1

Pada iterasi 1, pemangku kepentingan menyatakan bahwa aplikasi sudah memenuhi kebutuhannya dan tidak perlu ada perubahan lagi. Hasil wawancara pemangku kepentingan iterasi 1 terlampir pada Lampiran A.5



BAB 7 PENGUJIAN DAN ANALISIS

Pada bab ini dibahas mengenai pengujian yang dilakukan pada Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat. Setelah melewati proses implementasi, selanjutnya akan dilakukan proses pengujian dengan beberapa jenis pengujian, yaitu pengujian Fungsional (*blackbox*), pengujian usability dan pengujian compatibility. Analisis pengujian akan dilakukan setiap satu pengujian selesai dilakukan.

7.1 Pengujian Fungsional (*Blackbox*)

Pengujian Fungsional dilakukan pada Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat dengan tujuan untuk memastikan seluruh fungsionalitas pada aplikasi telah berjalan sesuai dengan pemodelan *use case* skenario. Untuk menentukan suatu fungsionalitas valid atau tidak valid, setiap fungsionalitas akan dijalankan, apabila fungsionalitas tersebut dapat berjalan dengan baik sesuai dengan *use case* skenario maka statusnya adalah valid, namun apabila gagal dijalankan maka statusnya adalah tidak valid.

Pada tabel 7.1 dijabarkan hasil pengujian Fungsional pada fungsionalitas Lihat Informasi Objek Layanan Kesehatan. Pada pengujian ini hasil pengujian dengan alur utama ditandai dengan point A dan hasil pengujian dengan alur alternatif ditandai dengan point B. Pengujian tersebut membuahkan hasil yang sesuai dengan hasil yang diharapkan dan dapat disimpulkan berstatus valid.

Tabel 7.1 Pengujian Fungsional Lihat Informasi Objek Layanan Kesehatan

Kode Kebutuhan	MS-F01
Nama Kasus Uji	Lihat Informasi Objek Layanan Kesehatan
Aktor	Tamu, Pengguna
Prosedur	<ol style="list-style-type: none"> 1. Aktor memilih kategori objek layanan kesehatan. 2. Aktor memilih objek layanan kesehatan.
Hasil yang Diharapkan	<ol style="list-style-type: none"> 1. A. Sistem menampilkan daftar objek layanan kesehatan yang dapat disaring dan dilakukan pencarian nama. B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data. 2. A. Sistem menampilkan informasi objek layanan kesehatan beserta tombol untuk memanggil nomor, melakukan navigasi, dan menandai objek layanan kesehatan. B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.

Tabel 7.1 Pengujian Fungsional Lihat Informasi Objek Layanan Kesehatan (lanjutan)

Hasil	<ol style="list-style-type: none"> 1. A. Sistem menampilkan daftar objek layanan kesehatan yang dapat disaring dan dilakukan pencarian nama. B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data. 2. A. Sistem menampilkan informasi objek layanan kesehatan beserta tombol untuk memanggil nomor, melakukan navigasi, dan menandai objek layanan kesehatan. B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.
Status	valid

Pada tabel 7.2 dijabarkan hasil pengujian Fungsional pada fungsionalitas Lihat Informasi Dokter. Pada pengujian ini hasil pengujian dengan alur utama ditandai dengan point A dan hasil pengujian dengan alur alternatif ditandai dengan point B. Pengujian tersebut membuahkan hasil yang sesuai dengan hasil yang diharapkan dan dapat disimpulkan berstatus valid.

Tabel 7.2 Pengujian Fungsional Lihat Informasi Dokter

Kode Kebutuhan	MS-F02
Nama Kasus Uji	Lihat Informasi Dokter
Aktor	Tamu, Pengguna
Prosedur	<ol style="list-style-type: none"> 1. Aktor menekan tombol spesialis 2. Aktor memilih spesialisasi dokter 3. Aktor memilih dokter
Hasil yang Diharapkan	<ol style="list-style-type: none"> 1. A. Sistem menampilkan daftar spesialis. B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data. 2. A. Sistem menampilkan informasi dokter beserta tombol untuk menandai dokter. B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.

Tabel 7.2 Pengujian Fungsional Lihat Informasi Dokter (lanjutan)

Hasil	<ol style="list-style-type: none"> 1. A. Sistem menampilkan daftar spesialis. B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data. 2. A. Sistem menampilkan informasi dokter beserta tombol untuk menandai dokter. B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.
Status	Valid

Pada tabel 7.3 dijabarkan hasil pengujian Fungsional pada fungsionalitas Lihat Denah Objek Layanan Kesehatan. Pada pengujian ini hasil pengujian dengan alur utama ditandai dengan point A dan hasil pengujian dengan alur alternatif ditandai dengan point B. Pengujian tersebut membuahkan hasil yang sesuai dengan hasil yang diharapkan dan dapat disimpulkan berstatus valid.

Tabel 7.3 Pengujian Fungsional Lihat Denah Objek Layanan Kesehatan

Kode Kebutuhan	MS-F03
Nama Kasus Uji	Lihat Denah Objek Layanan Kesehatan
Aktor	Tamu, Pengguna
Prosedur	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Denah Lokasi” 2. Aktor memilih kategori apa saja yang ingin ditampilkan 3. Aktor memilih marker 4. Aktor memilih untuk melihat detail informasi atau melakukan navigasi
Hasil yang Diharapkan	<ol style="list-style-type: none"> 1. A. Sistem menampilkan pilihan <i>checkbox</i> kategori objek layanan kesehatan B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data. 2. A. Sistem menampilkan marker layanan kesehatan B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data. 3. A. Sistem memberikan tombol untuk menuju halaman informasi dan tombol navigasi

Tabel 7.3 Pengujian Fungsional Lihat Denah Objek Layanan Kesehatan (lanjutan)

	<p>B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.</p> <p>4. A. Sistem menampilkan halaman detail informasi bila aktor memilih detail informasi atau menampilkan halaman Navigasi pada Google Maps apabila memilih Navigasi</p> <p>B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.</p>
<p>Hasil</p>	<p>1. A. Sistem menampilkan pilihan <i>checkbox</i> kategori objek layanan kesehatan</p> <p>B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.</p> <p>2. A. Sistem menampilkan marker layanan kesehatan</p> <p>B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.</p> <p>3. A. Sistem memberikan tombol untuk menuju halaman informasi dan tombol navigasi</p> <p>B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.</p> <p>4. Sistem menampilkan halaman detail informasi bila aktor memilih detail informasi atau menampilkan halaman Navigasi pada Google Maps apabila memilih Navigasi</p> <p>B. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.</p>
<p>Status</p>	<p>Valid</p>

Pada tabel 7.4 dijabarkan hasil pengujian Fungsional pada fungsionalitas Masuk Akun. Pada pengujian ini hasil pengujian dengan alur utama ditandai dengan point A dan hasil pengujian dengan alur alternatif ditandai dengan point B. Pengujian tersebut membuahkan hasil yang sesuai dengan hasil yang diharapkan dan dapat disimpulkan berstatus valid.

Tabel 7.4 Pengujian Fungsional Masuk Akun

Kode Kebutuhan	MS-F04
Nama Kasus Uji	Masuk Akun
Aktor	Tamu
Prosedur	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Masuk” 2. Sistem menampilkan form dengan kolom “Nama”, “Kata Sandi”, dan tombol “Masuk” 3. Aktor mengisi nama dan kata sandi 4. Aktor menekan tombol “Masuk”
Hasil yang Diharapkan	<ol style="list-style-type: none"> 1. Aktor berhasil masuk ke sistem dan halaman berpindah ke Beranda. 2. Sistem menampilkan “Nama / Kata Sandi salah” saat nama/ kata sandi yang dimasukkan salah 3. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Hasil	<ol style="list-style-type: none"> 1. Aktor berhasil masuk ke sistem dan halaman berpindah ke Beranda. 2. Sistem menampilkan “Nama / Kata Sandi salah” saat nama/ kata sandi yang dimasukkan salah 3. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data
Status	Valid

Pada tabel 7.5 dijabarkan hasil pengujian Fungsional pada fungsionalitas Daftar Akun. Pada pengujian ini hasil pengujian dengan alur utama ditandai dengan point A dan hasil pengujian dengan alur alternatif ditandai dengan point B. Pengujian tersebut membuahkan hasil yang sesuai dengan hasil yang diharapkan dan dapat disimpulkan berstatus valid.

Tabel 7.5 Pengujian Fungsional Daftar Akun

Kode Kebutuhan	MS-F05
Nama Kasus Uji	Daftar Akun
Aktor	Tamu
Prosedur	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Masuk” pada halaman Beranda 2. Sistem menyediakan tombol “Daftar” 3. Aktor menekan tombol “Daftar”

Tabel 7.5 Pengujian Fungsional Daftar Akun (lanjutan)

	<ol style="list-style-type: none"> 4. Sistem menampilkan form yang terdiri dari kolom “nama”, kolom “Kata Sandi”, kolom “Ulangi Kata Sandi”, dan tombol “Daftar” 5. Aktor mengisi form. 6. Aktor menekan tombol “Daftar”
Hasil yang Diharapkan	<ol style="list-style-type: none"> 1. Aktor berhasil mendaftar dan halaman kembali ke Masuk. 2. Sistem menampilkan “Nama Sudah Terpakai” apabila nama yang dimasukkan sudah ada dalam basis data. 3. Sistem menampilkan “Nama / Kata Sandi tidak boleh kosong” apabila nama / kata sandi kosong. 4. Sistem menampilkan toast “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.
Hasil	<ol style="list-style-type: none"> 1. Aktor berhasil mendaftar dan halaman kembali ke Masuk. 2. Sistem menampilkan “Nama Sudah Terpakai” apabila nama yang dimasukkan sudah ada dalam basis data. 3. Sistem menampilkan “Nama / Kata Sandi tidak boleh kosong” apabila nama / kata sandi kosong. 4. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.
Status	Valid

Pada tabel 7.6 dijabarkan hasil pengujian Fungsional pada fungsionalitas Keluar Akun. Pada pengujian ini hasil pengujian dengan alur utama ditandai dengan point A dan hasil pengujian dengan alur alternatif ditandai dengan point B. Pengujian tersebut membuahkan hasil yang sesuai dengan hasil yang diharapkan dan dapat disimpulkan berstatus valid.

Tabel 7.6 Pengujian Fungsional Keluar Akun

Kode Kebutuhan	MS-F06
Nama Kasus Uji	Keluar Akun
Aktor	Pengguna
Prosedur	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Penanda” pada halaman Beranda. 2. Sistem menampilkan tombol “Keluar”.

Tabel 7.6 Pengujian Fungsional Keluar Akun (lanjutan)

	3. Aktor menekan tombol “Keluar”.
Hasil yang Diharapkan	1. Aktor berhasil mengeluarkan akun yang sedang masuk ke sistem.
Hasil	1. Aktor berhasil mengeluarkan akun yang sedang masuk ke sistem.
Status	Valid

Pada tabel 7.7 dijabarkan hasil pengujian Fungsional pada fungsionalitas Penanda. Pada pengujian ini hasil pengujian dengan alur utama ditandai dengan point A dan hasil pengujian dengan alur alternatif ditandai dengan point B. Pengujian tersebut membuahkan hasil yang sesuai dengan hasil yang diharapkan dan dapat disimpulkan berstatus valid.

Tabel 7.7 Pengujian Fungsional Penanda

Kode Kebutuhan	MS-F07
Nama Kasus Uji	Penanda
Aktor	Pengguna
Prosedur	<ol style="list-style-type: none"> 1. Aktor menekan tombol “Penanda” pada halaman Beranda. 2. Sistem akan menampilkan tombol pilihan “Dokter” dan “Layanan Kesehatan” yang berisi dokter atau objek layanan kesehatan yang telah ditandai. 3. Aktor memilih tombol “Dokter” atau “Layanan Kesehatan”.
Hasil yang Diharapkan	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar objek layanan kesehatan atau daftar dokter yang telah ditandai. 2. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.
Hasil	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar objek layanan kesehatan atau daftar dokter yang telah ditandai. 2. Sistem menampilkan pesan “Gagal menghubungkan ke basis data” saat sistem mengalami masalah mengakses basis data.
Status	Valid

7.1.1 Analisis Hasil Pengujian Fungsional

Hasil yang diperoleh dari pengujian Fungsional pada aplikasi ini adalah seluruh kasus uji fungsional dapat dijalankan dengan baik tanpa terkendala. Status setiap pengujian adalah valid, sehingga dapat ditarik kesimpulan tingkat keberhasilan pengujian ini adalah 100%.

7.2 Pengujian *Compatibility*

Pengujian *Compatibility* ditujukan untuk mengetahui apakah modul Informasi Objek Layanan Kesehatan pada aplikasi Malang Sehat dapat bekerja pada lingkungan yang berbeda. Pengujian yang dilakukan yakni dengan menjalankan aplikasi pada emulator yang memiliki sistem operasi android dengan versi yang berbeda. Kemudian seluruh perangkat uji harus dapat menyelesaikan kasus uji berjumlah 7 kasus yang mangacu pada *use case*. Jika kasus uji dapat dijalankan dengan baik dan tidak menyebabkan *crash* pada aplikasi maka dianggap valid, namun jika gagal dalam menjalankan kasus uji maka dianggap tidak valid. Perangkat uji yang digunakan berjumlah 2 buah yang memiliki tingkatan API yang berbeda. Berikut merupakan hasil pengujian *Compatibility* yang dijabarkan pada Tabel 7.8.

Tabel 7.8 Hasil Pengujian *Compatibility* Secara Manual

Versi \ Use Case	API Level 19 (Android 4.4)	API Level 21 (Android 5)	API Level 30 (Android 11)
Masuk Akun	✗	✓	✓
Daftar Akun	✗	✓	✓
Keluar Akun	✗	✓	✓
Lihat Informasi Dokter	✗	✓	✓
Lihat Informasi Objek Layanan Kesehatan	✗	✓	✓
Lihat Denah Objek Layanan Kesehatan	✗	✓	✓
Penanda	✗	✓	✓

7.2.1 Analisis Pengujian *Compatibility*

Bedasarkan hasil pengujian *compatibility* yang ditunjukkan pada Tabel 7.8, seluruh hasil pengujian dapat dijalankan sesuai dengan batasan pengujian pengembangan aplikasi ini yaitu pada minimal Android 5 dengan API 21. Dengan demikian dapat ditarik kesimpulan bahwa Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat dapat berjalan sesuai dengan target versi android yang telah ditentukan.

7.3 Pengujian Usability

Pengujian usability merupakan salah satu pengujian dari sisi non fungsional yang dilakukan pada Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat. Pengujian usability dibagi menjadi dua tahap yaitu tugas uji usabilitas dan pengisian kuesioner System Usability Scale (SUS). Pengujian tugas uji dilakukan dengan memberikan tugas uji yang mengacu pada fungsionalitas. Tugas uji yang berikan akan jabarkan pada Tabel 7.9.

Tabel 7.9 Tugas Uji Usability

No	Tugas	Detail
1	Daftar Akun	Anda diminta untuk mendaftar akun baru.
2	Masuk Akun	Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan.
3	Melihat Informasi Dokter	Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem.
4	Melihat Informasi Objek Layanan Kesehatan	Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem. sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan.
5	Melihat Denah Objek Layanan Kesehatan	Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi
6	Melihat Penanda	Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya.
7	Keluar Akun	Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem.

Setelah dilakukan pengujian pada responden, hasil pengujian tersebut dijabarkan pada tabel 7.10. Bukti dari hasil kuesioner ini terdapat pada bagian lampiran B. Dengan menggunakan data hasil pengujian tersebut dilakukan

perhitungan untuk mencari nilai *success rate* yang dihitung menggunakan Persamaan 7.1.

$$Success\ rate = \frac{\text{jumlah B} + (\text{jumlah HB} \times 0,5)}{\text{jumlah tugas} \times \text{jumlah responden}} \times 100\ % \quad (7.1)$$

$$Success\ rate = \frac{68 + (2 \times 0,5)}{7 \times 10} \times 100\ %$$

$$Success\ rate = 98,57\ %$$

Persamaan 7.1

Tabel 7.10 Hasil Pengerjaan Tugas Oleh Responden

Responden	Tugas						
	1	2	3	4	5	6	7
R1	B	B	B	B	B	B	B
R2	B	B	B	B	B	B	B
R3	B	B	B	HB	B	HB	B
R4	B	B	B	B	B	B	B
R5	B	B	B	B	B	B	B
R6	B	B	B	B	B	B	B
R7	B	B	B	B	B	B	B
R8	B	B	B	B	B	B	B
R9	B	B	B	B	B	B	B
R10	B	B	B	B	B	B	B

Keterangan : B = Berhasil, HB = Hampir Berhasil, G = Gagal

Selanjutnya terdapat kuesioner SUS dengan 10 pernyataan, dimana pernyataan pada posisi ganjil merupakan pertanyaan positif dan posisi genap merupakan pertanyaan negatif. Pernyataan ganjil akan bernilai baik apabila responden memberikan skor tinggi, sedangkan untuk pernyataan genap akan bernilai bagus apabila responden memberikan skor rendah. Daftar pernyataan yang diberikan kepada responden akan dijabarkan pada Tabel 7.11.

Tabel 7.11 Daftar Pernyataan Kuesioner SUS

No	Pertanyaan
1	Saya merasa aplikasi ini akan sering saya gunakan
2	Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu)

Tabel 7.11 Daftar Pernyataan Kuesioner SUS (lanjutan)

No	Pertanyaan
3	Aplikasi ini memiliki fungsi yang berguna bagi saya
4	Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya
5	Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang
6	Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini
7	Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan
8	Saya merasa aplikasi ini butuh banyak perbaikan pada fungsi-fungsinya
9	Saya merasa akan menggunakan aplikasi ini untuk seterusnya
10	Saya merasa kebanyakan orang tidak membutuhkan aplikasi ini kedepannya

Berikutnya dijabarkan hasil dari pengisian kuesioner SUS pada sepuluh responden yang ditunjukkan pada Tabel 7.12. Bukti dari hasil kuesioner SUS yang asli terdapat pada bagian lampiran B.

Tabel 7.12 Hasil pengisian kuesioner SUS

Responden	Pernyataan									
	1	2	3	4	5	6	7	8	9	10
R1	3	2	4	1	5	2	4	2	4	1
R2	5	3	5	1	5	1	5	3	5	3
R3	4	2	5	2	4	1	4	1	4	2
R4	5	1	5	1	5	1	5	1	5	1
R5	4	2	5	2	5	2	4	2	4	2
R6	4	1	5	1	4	1	5	1	5	1
R7	5	3	5	2	5	3	5	2	4	2
R8	4	2	5	1	5	2	4	1	4	1
R9	4	3	4	1	4	1	4	2	4	1
R10	4	2	4	2	4	2	4	1	3	1

Tabel 7.12 Hasil pengisian kuesioner SUS (lanjutan)

*keterangan	R1 : Dwi Endah Puspitasari	R6 : Annanda Dian
	R2 : Hari Wibowo	R7 : Satya Eko Prasetyo
	R3 : Nanda Annisa R	R8 : Rizal
	R4 : Trias Ervina	R9 : Titania
	R5 : Nina Fardisrianesti	R10 : Sandra Elanda

Perhitungan skor SUS dilakukan pada setiap responden berdasarkan hasil kuesioner yang telah diisi oleh seluruh responden. Perhitungan skor SUS dilakukan dengan teori yang telah dijabarkan pada bab sebelumnya, maka didapatkan hasil perhitungan yang ditunjukkan pada Tabel 7.13.

Tabel 7.13 Perhitungan akumulasi skor SUS

Responden	Pernyataan										Jumlah	Jumlah x 2,5
	1	2	3	4	5	6	7	8	9	10		
R1	2	3	3	4	4	3	3	3	3	4	32	80
R2	4	2	4	4	4	4	4	2	4	2	34	85
R3	3	3	4	3	3	4	3	4	3	3	33	82,5
R4	4	4	4	4	4	4	4	4	4	4	40	100
R5	3	3	4	3	4	3	3	3	3	3	32	80
R6	3	4	4	4	3	4	4	4	4	4	38	95
R7	4	2	4	3	4	2	4	3	3	3	32	80
R8	3	3	4	4	4	3	3	4	3	4	35	87,5
R9	3	2	3	4	3	4	3	3	3	4	32	80
R10	3	3	3	3	3	3	3	4	2	4	31	77,5
Rata-rata												84,75

7.3.1 Analisis Pengujian Usability

Berdasarkan perhitungan skor akhir SUS pada Tabel 7.13, rata-rata yang didapat adalah 84,75, yang berarti sistem memiliki *adjectiva rating* "Good" dan responden memiliki *success rate* dalam melakukan tugas sebesar 98,57%. Dengan hasil tersebut maka Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat dapat dikatakan berhasil digunakan oleh responden dengan baik.

BAB 8 PENUTUP

Pada bab ini terdapat kesimpulan dan saran yang dibuat berdasarkan penelitian yang telah dilakukan pada bab-bab sebelumnya. Pada bagian kesimpulan berisi hasil dari penelitian dan bagian saran berisi saran untuk penelitian selanjutnya.

8.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan dimulai dari proses analisis kebutuhan, perancangan, implementasi dan pengujian, maka dapat dibuat kesimpulan sebagai berikut :

1. Analisis kebutuhan Modul Informasi Objek Layanan Kesehatan Pada Aplikasi Malang Sehat didapatkan berdasarkan proses iterasi *Prototyping* yang menghasilkan total 7 kebutuhan fungsional dan 3 kebutuhan non-fungsional. Kebutuhan-kebutuhan Fungsional ini dapat dijalankan oleh aktor yang dibagi menjadi 2 bagian, yakni tamu dan pengguna. Tamu adalah seseorang yang menggunakan aplikasi namun tidak melakukan fungsionalitas Masuk Akun, sedangkan pengguna adalah seseorang yang menggunakan aplikasi dan telah melakukan fungsionalitas Masuk Akun. Kebutuhan fungsional yang hanya dapat dijalankan Tamu berjumlah 2, kebutuhan fungsional yang hanya dapat dijalankan Pengguna berjumlah 2 dan kebutuhan fungsional yang dapat dijalankan oleh kedua aktor berjumlah 3.
2. Hasil rancangan Modul Informasi Objek Layanan Kesehatan Pada Aplikasi Malang Sehat yakni berupa perancangan arsitektur sistem untuk merancang bagaimana sistem nantinya akan berjalan, *sequence* diagram untuk merancang alur waktu dari setiap fungsionalitas berjalan, *class* diagram untuk merancang bagaimana tiap class pada program nantinya saling terkait, perancangan tabel basis data untuk merancang bagaimana data nantinya dikelompokkan pada basis data, dan terakhir perancangan antarmuka untuk merancang tampilan antarmuka yang nantinya berguna sebagai jembatan pemakai aplikasi dengan sistem aplikasi.
3. Implementasi Modul Informasi Objek Layanan Kesehatan Pada Aplikasi Malang Sehat memanfaatkan GPS pada perangkat, aplikasi Google Maps dan layanan Google Maps API untuk akses denah lokasi. Selanjutnya pada Implementasi kode program pada aplikasi Android bahasa java dipilih sebagai bahasa pemrogramannya. Lalu dari sisi olah data, MySql dipilih untuk sistem basis datanya dan bahasa pemrograman PHP untuk API-nya. Kemudian, Pengambilan data dari basis data dilakukan dengan program android yang mengirimkan request ke API, kemudian API akan mengembalikan data yang nantinya diterima oleh *framework* Retrofit pada aplikasi android. Setelah kode program selesai dibuat selanjutnya akan diimplementasikan antarmuka guna menghubungkan pemakai aplikasi dengan fungsionalitas-fungsionalitas aplikasi yang ada.

4. Pengujian Fungsional Modul Informasi Objek Layanan Kesehatan Pada Aplikasi Malang Sehat memiliki tingkat keberhasilan 100%. Selanjutnya pada pengujian *Compatibility*, didapatkan kesimpulan bahwa aplikasi dapat dijalankan dengan lancar pada tingkat API minimal 21 dikarenakan terdapat *framework* yang memiliki syarat penggunaan dengan tingkat API tersebut, sedangkan untuk maksimalnya aplikasi dapat berjalan pada versi terbaru yaitu Android 11 dengan API 30. Terakhir pada tugas uji pengujian Usabilitas didapatkan *success rate* sebesar 98,57% dan skor kuesioner *System Usability Scale (SUS)* menghasilkan nilai rata-rata sebesar 84,75 yang berarti memiliki *adjectiva rating "Good"*.

8.2 Saran

Saran yang diberikan pada Modul Informasi Objek Layanan Kesehatan Aplikasi Malang Sehat untuk proses pengembangan selanjutnya yaitu sebagai berikut :

1. Merubah fungsi tombol Navigasi yang awalnya mengalihkan aktor menuju aplikasi Google Maps menjadi tombol yang dapat melakukan navigasi di dalam aplikasi malang sehat itu sendiri.
2. Menambahkan fitur untuk mengurutkan daftar objek layanan kesehatan dan dokter berdasarkan jarak terdekat.
3. Menambah fungsionalitas untuk memberikan komentar kepada dokter dan objek layanan kesehatan.

DAFTAR REFERENSI

- Anon., t.thn. *KBBI Daring*. [Online]
Tersedia di: <<https://kbbi.kemdikbud.go.id>> [Diakses 17 Februari 2020].
- Bangor, A., Philip, K. & Miller, J., 2009. Determining What Individual SUS. *Juournal Of Usability Studies*.
- Booch, G., Rumbaugh, J. & Jacobson, I., 1999. *Unified Modeling Language User Guide, The (2nd Edition) (Addison-Wesley Object Technology Series)*. 2nd penyunt. s.l.:Addison Wesley.
- Brooke, J., 1996. *Usability Evaluation in Industry*. London: Taylor and Francis.
- Christensson, P., 2016. *API Definition*. [Online]
Tersedia di: <<https://techterms.com/definition/api>> [Diakses 14 Juni 2020].
- Dameri, R. P. & Rosenthal-Sabroux, C., 2014. *Smart City: How to Create Public and Economic Value with High Technology in Urban Space*. New York: Springer.
- Ekanayake, N., 2018. Android Operating System. *Android Operating System*, Mei.pp. 1-11.
- Google Developers, 2020. *Ringkasan*. [Online]
Tersedia di:
<<https://developers.google.com/maps/documentation/android-sdk/intro>> [Diakses 15 Juni 2020].
- Jackson, W., 2011. *Android Apps for Absolute Beginners*. s.l.:APRESS.
- Klug, B., 2017. An Overview of the System Usability Scale in Library Website and System Usability Testing. 1(6).
- Komputer, W., 2010. Panduan Belajar MySQL Database Server. [e-book] Jakarta Selatan: Mediakita. Tersedia di: Google Books
<<https://books.google.co.id/books?id=rR1VNYpc08wC&lpg=PA21&ots=TKnha5jcfU&dq=apa%20itu%20mysql&lr&hl=id&pg=PP1#v=onepage&q=apa%20itu%20mysql&f=false>> [Diakses 20 Juli 2021].
- Meijer, A. & Bolívar, M. P. R., 2015. International Review of Administrative Sciences. *Governing the smart city: a review of the literature on smart urban*, 29 April, 82(2), pp. 392-408.
- Miles, R. & Hamilton, K., 2006. *Learning UML 2.0*. Sebastopol: O'Reilly Media, Inc..
- Moran, K., 2019. *Usability Testing 101*. [Online]
Tersedia di: <<https://www.nngroup.com/articles/usability-testing-101>> [Diakses 17 Juni 2020].
- Nielsen, J., 2012. *Usability 101: Introduction to Usability*. [Online]
Tersedia di: <<https://www.nngroup.com/articles/usability-101-introduction-to-usability>> [Diakses 17 Juni 2020].

Permana, K. S., 2019. *Ketika Pengguna Internet dan Smartphone Terus Meningkat, Android Dominasi Pasar Indonesia dan Dunia*. [Online] Tersedia di: <<https://jabar.tribunnews.com/2019/01/24/ketika-pengguna-internet-dan-smartphone-terus-meningkat-android-dominasi-pasar-indonesia-dan-dunia>> [Diakses 29 Januari 2020].

Pressman, R. S., 2010. *Software Engineering: A Practitioner's App*. 7th penyunt. New York: Raghothaman Srinivasan.

Rosa, A. S. & Shalahuddin, M., 2018. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Revisi penyunt. Bandung: Informatika Bandung.

Sauro, J. & Lewis, J. R., 2012. *Quantifying the User Experience: Practical Statistics*. San Francisco(CA): Morgan Kaufmann.

Schiller, J. & Voisard, A., 2004. *Location-Based Services*. San Francisco: Elsevier.

Sommerville, I., 2011. *Software_Engineering*. 9th penyunt. Boston: Pearson Education, Inc..

Susanto, T. D. ed., 2019. *SMART CITY*. Surabaya: AISINDO.

Yanto, R., 2016. *Manajemen Basis Data Menggunakan MySQL*. [e-book] Yogyakarta: Deepublish. Tersedia di: Google Books <<https://books.google.co.id/books?id=VMReDwAAQBAJ&pg=PR6&ots=4t1z7bgAxI&dq=apa%20itu%20basis%20data&lr&hl=id&pg=PA5#v=onepage&q=apa%20itu%20basis%20data&f=false>>[Diakses 20 Juli 2021]

Zhang, T., Gao, J., Cheng, J. & Uehara, T., 2015. *Compatibility Testing Service for Mobile Applications*. s.l., s.n.

LAMPIRAN A HASIL WAWANCARA

A.1 Wawancara Kelompok Pengguna

Nama Responden : Restian Parengga

Usia : 22 Tahun

Pekerjaan : Penulis

No.	Pertanyaan	Jawaban
1	Apakah anda pernah mengalami permasalahan terkait kesehatan selama berada di kota Malang?	Pernah, waktu itu sakit demam tinggi.
2	Apa yang anda lakukan Ketika anda mengalami gangguan terkait Kesehatan?	Istirahat. Setelah itu langsung cari obat sendiri ke apotik. Kalau sudah tidak kuat biasanya langsung ke puskesmas.
3	Apakah anda pernah mengunjungi fasilitas pelayanan kesehatan selama berada di kota Malang?	Di Malang pernah ke Rumah Sakit dan Puskesmas.
4	Seberapa sering anda mengunjungi fasilitas pelayanan kesehatan?	Beberapa kali kalau sedang sakit, menemani teman berobat dan menjenguk temen sakit.
5	Fasilitas pelayanan kesehatan apa yang biasa anda kunjungi?	Lebih sering ke Puskesmas. Namanya puskesmas Kendal sari, Malang.
6	Bagaimana anda mendapatkan informasi tersebut?	Searching di google, terus cari yang terdekat dari kost.
7	Kenapa anda memilih fasilitas pelayanan kesehatan tersebut?	Karena yang paling terdekat dari lokasi saya puskesmas Kendal sari setelah saya searching.

8	Bagaimana pengalaman anda saat mencari informasi tersebut dan apa yang anda rasakan?	Panik saat mencari tempatnya, harus mencari satu per-satu dulu disaat sudah dalam keadaan genting. Saya juga tidak tau sebenarnya disana ada dokter yang menangani atau dan sampai sana ngantrinya panjang.
9	Apakah anda pernah menggunakan aplikasi bidang kesehatan?	Pernah. Rumah sakit Hermina mobile app
10	Bagaimana pendapat anda jika terdapat informasi objek kesehatan dalam bentuk aplikasi secara mobile?	Bagus, setuju. Memudahkan dan membantu buat orang perantau seperti saya yang tiba-tiba sakit butuh serba cepat dan praktis.
11	Fitur apa yang anda harapkan pada aplikasi ini?	Informasi fasilitas pelayanan kesehatan sesuai lokasi pengguna, Informasi jadwal praktik dokter, Reservasi online, dan informasi apotek.

Nama Responden : Farisa Aidilla Faradilla

Usia : 22 Tahun

Pekerjaan : Mahasiswa

No.	Pertanyaan	Jawaban
1	Apakah anda pernah mengalami permasalahan terkait kesehatan selama berada di kota Malang?	Pernah. Soalnya Malang dingin jadinya sering sakit terutama sinus
2	Apa yang anda lakukan Ketika anda mengalami gangguan terkait Kesehatan?	Istirahat kalo tidak, ya minum obat. Kalo sudah parah langsung ke dokter.
3	Apakah anda pernah mengunjungi fasilitas pelayanan kesehatan selama	Pernah, kerumah sakit Hermina sama Rumah Sakit universitas brawijaya.

	berada di kota Malang?	
4	Seberapa sering anda mengunjungi fasilitas pelayanan kesehatan?	Semester bisa dua kali kerumah sakit, sisanya jenguk temen jika ada temen yang sakit.
5	Fasilitas pelayanan kesehatan apa yang biasa anda kunjungi?	Untuk berobat biasanya selalu memilih rumah sakit Hermina, Malang
6	Bagaimana anda mendapatkan informasi tersebut?	Searching di google, cari rumah sakit Hermina.
7	Kenapa anda memilih fasilitas pelayanan kesehatan tersebut?	Soalnya cuman tahu rumah sakit Hermina aja, di Jakarta juga ada. Waktu itu mau rumah sakit universitas brawijaya tapi sepertinya belum jadi, ya. Kalau mau coba rumah sakit lain pada jauh dari lokasi tempat kost dan rumah sakitnya lainnya saya kurang mengenal. Jadinya yang saya tahu hanya Hermina saja.
8	Bagaimana pengalaman anda saat mencari informasi tersebut dan apa yang anda rasakan?	Karena sudah darurat, jadi langsung aja pilih Hermina. Pernah juga menghubungi rumah sakit terus gak akurat gitu sama data di website.
9	Apakah anda pernah menggunakan aplikasi bidang kesehatan?	Tidak pernah
10	Bagaimana pendapat anda jika terdapat informasi objek kesehatan dalam bentuk aplikasi secara mobile?	Bagus, setuju. Bisa membantu dan berguna
11	Fitur apa yang anda harapkan pada aplikasi ini?	Bisa chat dengan pihak rumah sakitnya. Harus tetep update, kalau di malang datanya suka tidak akurat.

Nama Responden : Irhamsyah Faishal

Usia : 22 Tahun

Pekerjaan : Mahasiswa

No.	Pertanyaan	Jawaban
1	Apakah anda pernah mengalami permasalahan terkait kesehatan selama berada di kota Malang?	Pernah. Sakit demam berdarah, sakit panas, sinus.
2	Apa yang anda lakukan Ketika anda mengalami gangguan terkait Kesehatan?	Kalau riwayat penyakit seperti dbd, panas biasanya tahu apa yang harus dilakukan seperti membeli obat ke apotik
3	Apakah anda pernah mengunjungi fasilitas pelayanan kesehatan selama berada di kota Malang?	Pernah. Ke klinik dan rumah sakit
4	Seberapa sering anda mengunjungi fasilitas pelayanan kesehatan?	2 semester sekali. Jadi setahun bisa sekali
5	Fasilitas pelayanan kesehatan apa yang biasa anda kunjungi?	Klinik Kimia Farma, ijen dan rumah sakit Permata Bunda.
6	Bagaimana anda mendapatkan informasi tersebut?	Kalau klinik kimia farma, mencari dokter THT dan cari jam praktiknya dokternya di google. Kalau mencari informasi rumah sakit permata bunda searching juga lalu pilih yang paling terdekat dari lokasi kontrakan.
7	Kenapa anda memilih fasilitas pelayanan kesehatan tersebut?	Kalau kimia farma waktu itu lagi sakit sinus, sehingga terjadi gangguan pada telinga. Jadinya mencari dokter THT di Malang. Ternyata tidak ada dokter dimana rumah sakit tersebut khusus dokter yang lengkap seperti di jakarta. Ternyata yang sesuai dicari di klinik kimia farma. Lalu dilihat jam praktiknya di google karena rekomendasinya paling atas dan ulasannya juga bagus. Kalau rumah

		sakit permata bunda, paling dekat dari kontrakan disitu. Sempet tadinya mau rumah lainnya, ternyata masih kurang 93 informasi terkait asuransi. Kurang informasi juga terkait rumah sakitnya bagus apa tidak (review) dan pernah nemenin temen juga berobat, menurut saya manajemen penanganannya kurang baik. Dan saya tidak tahu ini rs swasta apa negeri. Jadinya ke permata bunda yang pastipasti saja.
8	Bagaimana pengalaman anda saat mencari informasi tersebut dan apa yang anda rasakan?	Pertama, waktu sakit DBD saya yakin untuk tidak pergi ke klinik. jadinya pergi ke RS permata bunda. Kalau permata bunda karena dekat dengan lokasi. Setelah dirawat, beberapa kali nemenin temen dan jenguk temen ternyata baru tau rumah sakit lavalette, persada ternyata bagus juga. Karena riwayat penyakit dikit jadinya less information. Saya mengira-ngira dalam memilih rumah sakit apa yang sekiranya berkompeten dalam menangani Kalau klinik sakit sinus, karena jam operasional rs sibuk dan saya merasa penyakit ini tidak perlu sampai di rawat. Jadinya pilih klinik. Menurut saya, pengalaman orang yang pernah ngalamin sakit atau berobat (mouth to mouth) lebih meningkatkan kepercayaan.
9	Apakah anda pernah menggunakan aplikasi bidang kesehatan?	Tidak pernah
10	Bagaimana pendapat anda jika terdapat informasi objek kesehatan dalam bentuk aplikasi secara mobile?	Iya setuju.
11	Fitur apa yang anda harapkan pada aplikasi ini?	Hotline service, jadwal praktik dokter. Reviewnya di kategorikan, seperti review secara umum rumah sakit dan fasilitas, review dengan riwayat penyakitnya sendiri, review dokter.

Nama Responden : Asmi Afdillah

Usia : 22 Tahun

Pekerjaan : Fresh Graduate

No.	Pertanyaan	Jawaban
1	Apakah anda pernah mengalami permasalahan terkait kesehatan selama berada di kota Malang?	Pernah. Gejala tipis sama sakit gigi
2	Apa yang anda lakukan Ketika anda mengalami gangguan terkait Kesehatan?	Berobat klinik atau rumah sakit
3	Apakah anda pernah mengunjungi fasilitas pelayanan kesehatan selama berada di kota Malang?	Pernah. Ke puskesmas, klinik sama rumah sakit
4	Seberapa sering anda mengunjungi fasilitas pelayanan kesehatan?	Jarang, setahun bisa 2-3 kali. Tidak menentu, tergantung kondisi. Tapi akhir-akhir ini lagi sering, tahun kemarin tidak.
5	Fasilitas pelayanan kesehatan apa yang biasa anda kunjungi?	Lebih sering ke klinik
6	Bagaimana anda mendapatkan informasi tersebut?	Semua informasi klinik dan rumah sakit yang saya datengin dapat dari mamah, tidak pernah nyari sendiri. Kalau puskesmas waktu itu lagi nyari yang terdekat.
7	Kenapa anda memilih fasilitas pelayanan kesehatan tersebut?	Kalau klinik yang aku datengin waktu itu karena ada BPJS. Beda cerita waktu kejadian puskesmas, soalnya lagi darurat sehabis kecelakaan, jadi cari yang terdekat yang bisa cepat menangani.
8	Bagaimana pengalaman anda saat mencari informasi tersebut dan apa yang anda rasakan?	Semua informasi yang saya dapat dari mamah. Pernah juga waktu itu dateng ke klinik terus di rujuk kerumah sakit karena klinik tersebut nggak ada dokter yang menangani. Kalau kecelakaan waktu itu, teman saya yang mengantar sampai puskesmas. Saya pasrah saja yang 95 penting bisa

		tertangani dengan cepat. Yang jelas terdekat dari lokasi kecelakaan.
9	Apakah anda pernah menggunakan aplikasi bidang kesehatan?	Pernah. Aplikasi Malang ker
10	Bagaimana pendapat anda jika terdapat informasi objek kesehatan dalam bentuk aplikasi secara mobile?	Setuju. Bisa mempermudah. Soalnya pernah buka salah satu aplikasi, lagi nyari praktik dokter tapi ga lengkap informasinya. Terus nge scroll sampe bawah secara manual, soalnya tidak ada fitur search dan menampilkan berbentuk tabel.
11	Fitur apa yang anda harapkan pada aplikasi ini?	Fitur "search" penting banget, rating, jam operasional dan foto dokter biar tahu apakah dokter yang kita pilih sesuai apa tidak.

Nama Responden : Rita Matilda

Usia : 45 Tahun

Pekerjaan : Wiraswasta

No.	Pertanyaan	Jawaban
1	Apakah anda pernah mengalami permasalahan terkait kesehatan selama berada di kota Malang?	Pernah.
2	Apa yang anda lakukan Ketika anda mengalami gangguan terkait Kesehatan?	Kalau sakit coba herbal dulu. Jika tidak ada perubahan, baru ke dokter. Kecuali penyakit – penyakit yang berbahaya seperti jantung. Itu langsung ke dokter
3	Apakah anda pernah mengunjungi fasilitas pelayanan kesehatan selama berada di kota Malang?	Pernah. Lebih sering ke puskesmas dan klinik
4	Seberapa sering anda mengunjungi	Sering. Selain saya yang sakit, terkadang mengantar 2 anak saya kalau sedang sakit. Kalau

	fasilitas pelayanan kesehatan?	anak sakit setahun bisa 15 kali. Puskesmas di arjosari biasanya karena dokternya bagus. Cumin praktik di puskesmas itu
5	Fasilitas pelayanan kesehatan apa yang biasa anda kunjungi?	Puskesmas dan klinik
6	Bagaimana anda mendapatkan informasi tersebut?	Mouth to mouth, lebih sering dari temen sama nyari. Dokter yang bagus apa rumah sakit yang bagus apa. Nyari di internet profil dokter ahli apa pengalaman apa praktik dimana. Saya selalu melakukan mouth to mouth. Dulu pernah menghubungi 108 lewat penerangan lalu bertanya untuk meminta rekomendasi yang paling bagus. Pernah juga ke apotik yang terkenal dan bertanya ke pelayanan. 97 Kalau tidak dapat dari orang-orang, searching di internet. Tapi tetap prioritas mouth to mouth
7	Kenapa anda memilih fasilitas pelayanan kesehatan tersebut?	Pertama dokter yang sedang praktik di tempat tersebut, kedua pembayarannya, ketiga jaraknya.
8	Bagaimana pengalaman anda saat mencari informasi tersebut dan apa yang anda rasakan?	Mouth to mouth karena percaya dengan teman atau saudara. Karena saya selalu menggali informasi sejauh mungkin lalu di yakinkan oleh internet.
9	Apakah anda pernah menggunakan aplikasi bidang kesehatan?	Tidak pernah. Selalu via telepon
10	Bagaimana pendapat anda jika terdapat informasi objek kesehatan dalam bentuk aplikasi secara mobile?	Setuju.
11	Fitur apa yang anda harapkan pada aplikasi ini?	Informasi dokter seperti jadwal praktik, alamat praktik dan keahlian dokter. Informasi fasilitas pelayanan kesehatan, nomer telepon



Nama Responden : Ayu Paramitha

Usia : 31 Tahun

Pekerjaan : Wiraswasta

No.	Pertanyaan	Jawaban
1	Apakah anda pernah mengalami permasalahan terkait kesehatan selama berada di kota Malang?	Pernah. Sejak tahun 2017 saya mengalami beberapa penyakit. Tapi lebih sering semenjak tahun lalu saya sakit tbc usus.
2	Apa yang anda lakukan Ketika anda mengalami gangguan terkait Kesehatan?	Pertama saya membeli obat di apotik. Tapi jika 1-2 hari tidak ada tanda-tanda membaik atau makin drop, saya pergi ke rumah sakit
3	Apakah anda pernah mengunjungi fasilitas pelayanan kesehatan selama berada di kota Malang?	Pernah. Lebih sering ke rumah sakit dan klinik
4	Seberapa sering anda mengunjungi fasilitas pelayanan kesehatan?	Sering, sebulan bisa 1-2 kali. Setahun bisa 12-24 kali.
5	Fasilitas pelayanan kesehatan apa yang biasa anda kunjungi?	Rumah Sakit dan Klinik
6	Bagaimana anda mendapatkan informasi tersebut?	Dari teman, lalu mencari di internet tentang dokter yang kira-kira yang saya datangi itu praktik dimana
7	Kenapa anda memilih fasilitas pelayanan kesehatan tersebut?	Saya melihat tempat dokter yang di rekomendasi dan fasilitas rumah sakitnya
8	Bagaimana pengalaman anda saat mencari informasi tersebut	Untuk mencari jadwal praktik dokter mudah karena bis acari di internet, namun di beberapa tempat sudah tidak update dan terkadang tidak sesuai jadinya saya perlu menkonfirmasi kembali

	dan apa yang anda rasakan?	dengan menghubungi rumah sakit atau klinik tersebut
9	Apakah anda pernah menggunakan aplikasi bidang kesehatan?	Aplikasi mobile app hermina
10	Bagaimana pendapat anda jika terdapat informasi objek kesehatan dalam bentuk aplikasi secara mobile?	Setuju.
11	Fitur apa yang anda harapkan pada aplikasi ini?	Nomor antrian biar tidak mengantri dan jadwal praktik dokter

A.2 Wawancara Komunikasi Pemangku Kepentingan iterasi 0

No.	Pertanyaan	Jawaban
1	Siapakah nama Anda dan apa jabatan Anda?	Staff Kominfo Kepala Bidang APTIKA.
2	Apa tujuan dari sistem informasi objek kesehatan Kota Malang ini?	Untuk memudahkan masyarakat Kota Malang dalam mengakses semua informasi terkait kesehatan yang berada di lingkungan sekitar mereka, yaitu di Kota Malang. Selain itu informasi yang ditampilkan pada sistem ini merupakan informasi yang valid dan lengkap, sehingga masyarakat dapat mengakses dan mendapatkan informasi secara efektif dan efisien.
3	Apa permasalahan dari sistem informasi objek kesehatan saat ini?	Saat ini sistem yang ada (fiturnya) masih terbatas dan informasi yang ditampilkan kurang tertata. Informasi yang disediakan pada platform di Kota Malang khususnya bidang kesehatan terdapat di situs web opendata.makangkota dan situs web dinkes.malangkota.go .

4	Apa yang anda harapkan dari sistem informasi objek kesehatan?	Dengan adanya sistem ini diharapkan masyarakat dapat mengakses dan mendapatkan informasi dengan mudah dalam mencari pelayanan kesehatan yang terbaik baik itu melalui website maupun aplikasi, selain itu dapat meningkatkan tata kinerja Pemerintah dan juga layanan masyarakat yang efisien, efektif serta berintegritas. Penyusunan Smart City ini juga melibatkan seluruh Organisasi Perangkat Daerah (OPD) di lingkungan Pemerintah Kota Malang (Pemkot). Terutama kondis saat ini yang sedang pandemi, jadi perlu di tingkatan lagi pelayanannya. Jaman 87 digital sekarang sudah menjadi kebutuhan masyarakat dalam kegiatan aktivitas dalam kesehariannya.
5	Apa kendala yang dialami pengguna ketika menggunakan sistem informasi fasilitas pelayanan kesehatan di Kota Malang?	Sejauh ini belum ada.

A.3 Wawancara Umpan Balik Pemangku Kepentingan Iterasi 0

No.	Pertanyaan	Jawaban
1	Siapakah nama Anda dan apa jabatan Anda?	Herman Tolle, Dr. Eng., S.T, M.T. selaku perwakilan kominfo untuk studi kasus Smart City di kota Malang
2	Apakah aplikasi yang dipresentasikan sudah memenuhi kebutuhan dalam informasi layanan kesehatan di Kota Malang?	Untuk kebutuhan yang ada sudah bagus tidak perlu dilakukan perubahan, akan tetapi dibutuhkan penambahan kebutuhan baru untuk personalisasi pengguna.

A.4 Wawancara Komunikasi Pemangku Kepentingan iterasi 1

No.	Pertanyaan	Jawaban
1	Siapakah nama Anda dan apa jabatan Anda?	Herman Tolle, Dr. Eng., S.T, M.T. selaku perwakilan kominfo untuk studi kasus Smart City di kota Malang

2	Saat proses wawancara umpan balik aplikasi iterasi 0, anda berkata diperlukan penambahan kebutuhan untuk personalisasi pengguna, bagaimana detail dari personalisasi pengguna tersebut?	Personalisasi pengguna yang dimaksud adalah fungsionalitas sistem dimana pengguna dapat melakukan fungsi untuk login dan pengguna dapat menyimpan informasi yang ada pada sistem ke dalam akun yang dibuatnya.
---	---	--

A.5 Wawancara Umpan Balik Pemangku Kepentingan Iterasi 1

No.	Pertanyaan	Jawaban
1	Siapakah nama Anda dan apa jabatan Anda?	Herman Tolle, Dr. Eng., S.T, M.T. selaku perwakilan kominfo untuk studi kasus Smart City di kota Malang
2	Apakah aplikasi yang dipresentasikan sudah memenuhi kebutuhan dalam informasi layanan kesehatan di Kota Malang?	Semua kebutuhan yang diperlukan sudah tersedia dalam aplikasi.

LAMPIRAN B HASIL KUESIONER SUS

Responden 1

Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan

Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut

Email *

lin.puspitasari73@gmail.com

Nama *

Dwi Endah Puspitasari

Tempat tinggal jika berada di kota Malang *

Rumah / Kontrakan / Kos pribadi
 Rumah / Kontrakan / Kos milik saudara
 Hotel / Villa (Wisatawan / pendatang)
 Pernah ke malang tetapi tidak menetap/ menginap
 Tidak pernah ke Malang

Anda diminta untuk mendaftar akun baru. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem. sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

Berhasil
 Hampir Berhasil
 Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan Untitled Section

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

<p>Aplikasi ini memiliki fungsi yang berguna bagi saya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Sangat Setuju</p>	<p>Saya merasa akan menggunakan aplikasi ini untuk seterusnya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p> <p><small>This content is neither created nor endorsed by Google.</small></p>

Responden 2

<p>Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan</p> <p>Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut</p> <p>Email *</p> <p>wibowosub@gmail.com</p> <p>Nama *</p> <p>hari wibowo</p> <p>Tempat tinggal jika berada di kota Malang *</p> <p><input type="radio"/> Rumah / Kontrakan / Kos pribadi</p> <p><input type="radio"/> Rumah / Kontrakan / Kos milik saudara</p> <p><input checked="" type="radio"/> Hotel / Villa (Wisatawan / pendatang)</p> <p><input type="radio"/> Pernah ke malang tetapi tidak menetap/ menginap</p> <p><input type="radio"/> Tidak pernah ke Malang</p>	<p>Anda diminta untuk mendaftar akun baru. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p> <p>Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p> <p>Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p>
--	---

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem. sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

Berhasil
 Hampir Berhasil
 Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan KesehatanUntitled Section

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Aplikasi ini memiliki fungsi yang berguna bagi saya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa akan menggunakan aplikasi ini untuk seterusnya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

This content is neither created nor endorsed by Google.

Responden 3

Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan

Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut

Email *
nandaannisar@gmail.com

Nama *
Nanda Annisa R

Tempat tinggal jika berada di kota Malang *

Rumah / Kontrakan / Kos pribadi

Rumah / Kontrakan / Kos milik saudara

Hotel / Villa (Wisatawan / pendatang)

Pernah ke malang tetapi tidak menetap/ menginap

Tidak pernah ke Malang

Anda diminta untuk mendaftar akun baru. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem. sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

Berhasil

Hampir Berhasil

Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan Untitled Section

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

<p>Aplikasi ini memiliki fungsi yang berguna bagi saya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Sangat Setuju</p>	<p>Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa akan menggunakan aplikasi ini untuk seterusnya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>

This content is neither created nor endorsed by Google.

Responden 4

<p>Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan</p> <p>Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut</p> <p>Email *</p> <p>ibuedamar@gmail.com</p> <p>Nama *</p> <p>Trias Ervina</p> <p>Tempat tinggal jika berada di kota Malang *</p> <p><input type="radio"/> Rumah / Kontrakan / Kos pribadi</p> <p><input type="radio"/> Rumah / Kontrakan / Kos milik saudara</p> <p><input checked="" type="radio"/> Hotel / Villa (Wisatawan / pendatang)</p> <p><input type="radio"/> Pernah ke malang tetapi tidak menetap/ menginap</p> <p><input type="radio"/> Tidak pernah ke Malang</p>	<p>Anda diminta untuk mendaftar akun baru. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p> <p>Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p> <p>Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p>
---	---

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem, sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

Berhasil
 Hampir Berhasil
 Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan Untitled Section

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Aplikasi ini memiliki fungsi yang berguna bagi saya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa akan menggunakan aplikasi ini untuk seterusnya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

This content is neither created nor endorsed by Google.

Responden 5

Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan

Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut

Email *

nileen1609@gmail.com

Nama *

Nina Fardisrianesti

Tempat tinggal jika berada di kota Malang *

Rumah / Kontrakan / Kos pribadi

Rumah / Kontrakan / Kos milik saudara

Hotel / Villa (Wisatawan / pendatang)

Pernah ke malang tetapi tidak menetap/ menginap

Tidak pernah ke Malang

Anda diminta untuk mendaftar akun baru. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem, sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

Berhasil

Hampir Berhasil

Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan Untitled Section

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Aplikasi ini memiliki fungsi yang berguna bagi saya *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa akan menggunakan aplikasi ini untuk seterusnya *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Responden 6

Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan

Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut

Email *

annandadian04@gmail.com

Nama *

annanda dian

Tempat tinggal jika berada di kota Malang *

Rumah / Kontrakan / Kos pribadi

Rumah / Kontrakan / Kos milik saudara

Hotel / Villa (Wisatawan / pendatang)

Pernah ke malang tetapi tidak menetap/ menginap

Tidak pernah ke Malang

Anda diminta untuk mendaftar akun baru. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem, sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

Berhasil
 Hampir Berhasil
 Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan KesehatanUntitled Section

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Aplikasi ini memiliki fungsi yang berguna bagi saya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa akan menggunakan aplikasi ini untuk seterusnya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

This content is neither created nor endorsed by Google.

Responden 7

Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan

Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut

Email *

satyaprasetyo96@gmail.com

Nama *

Satya Eko Prasetyo

Tempat tinggal jika berada di kota Malang *

- Rumah / Kontrakan / Kos pribadi
- Rumah / Kontrakan / Kos milik saudara
- Hotel / Villa (Wisatawan / pendatang)
- Pernah ke malang tetapi tidak menetap/ menginap
- Tidak pernah ke Malang

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem, sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan. *

- Berhasil
- Hampir Berhasil
- Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

- Berhasil
- Hampir Berhasil
- Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

- Berhasil
- Hampir Berhasil
- Gagal

Anda diminta untuk mendaftar akun baru. *

- Berhasil
- Hampir Berhasil
- Gagal

Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *

- Berhasil
- Hampir Berhasil
- Gagal

Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *

- Berhasil
- Hampir Berhasil
- Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

- Berhasil
- Hampir Berhasil
- Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

- 1 2 3 4 5
- Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

- 1 2 3 4 5
- Sangat Tidak Setuju Sangat Setuju



<p>Aplikasi ini memiliki fungsi yang berguna bagi saya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Sangat Setuju</p>	<p>Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Sangat Setuju</p>
<p>Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Sangat Setuju</p>	<p>Saya merasa akan menggunakan aplikasi ini untuk seterusnya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>

Responden 8

<p>Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan</p> <p>Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut</p>	<p>Anda diminta untuk mendaftar akun baru. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p>
<p>Email *</p> <p>rijal.msidiq@gmail.com</p>	<p>Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p>
<p>Nama *</p> <p>Rizal</p>	<p>Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p>
<p>Tempat tinggal jika berada di kota Malang *</p> <p><input type="radio"/> Rumah / Kontrakan / Kos pribadi</p> <p><input type="radio"/> Rumah / Kontrakan / Kos milik saudara</p> <p><input checked="" type="radio"/> Hotel / Villa (Wisatawan / pendatang)</p> <p><input type="radio"/> Pernah ke malang tetapi tidak menetap/ menginap</p> <p><input type="radio"/> Tidak pernah ke Malang</p>	

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem, sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

Berhasil
 Hampir Berhasil
 Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan KesehatanUntitled Section

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Aplikasi ini memiliki fungsi yang berguna bagi saya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa akan menggunakan aplikasi ini untuk seterusnya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Responden 9

Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan

Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut

Email *

luqyanalathifa0804@gmail.com

Nama *

Luqyana Lathifa Prasetyo Putri

Tempat tinggal jika berada di kota Malang *

Rumah / Kontrakan / Kos pribadi

Rumah / Kontrakan / Kos milik saudara

Hotel / Villa (Wisatawan / pendatang)

Pernah ke malang tetapi tidak menetap/ menginap

Tidak pernah ke Malang

Anda diminta untuk mendaftar akun baru. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem. sekaligus melakukan percobaan menekan tombol panggilan, Navigasi, dan Tandai objek layanan kesehatan. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

Berhasil

Hampir Berhasil

Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

Berhasil

Hampir Berhasil

Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan Untitled Section

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

1 2 3 4 5

Sangat Tidak Setuju Sangat Setuju

<p>Aplikasi ini memiliki fungsi yang berguna bagi saya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Sangat Setuju</p>
<p>Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa akan menggunakan aplikasi ini untuk seterusnya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>
<p>Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>	<p>Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *</p> <p>1 2 3 4 5</p> <p>Sangat Tidak Setuju <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Sangat Setuju</p>

This content is neither created nor endorsed by Google.

Responden 10

<p>Pengujian Usabilitas Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan</p> <p>Lakukanlah tugas dibawah ini kemudian beri respon balik apakah anda berhasil, hampir berhasil, atau gagal dalam melaksanakan tugas tersebut</p>	<p>Anda diminta untuk mendaftar akun baru. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p>
<p>Email *</p> <p>elanda385@gmail.com</p>	<p>Anda diminta untuk masuk kedalam sistem menggunakan akun yang telah didaftarkan. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p>
<p>Nama *</p> <p>Sandra Elanda</p>	<p>Anda diminta untuk melihat detail informasi salah satu dokter yang terdapat pada sistem. *</p> <p><input checked="" type="radio"/> Berhasil</p> <p><input type="radio"/> Hampir Berhasil</p> <p><input type="radio"/> Gagal</p>
<p>Tempat tinggal jika berada di kota Malang *</p> <p><input checked="" type="radio"/> Rumah / Kontrakan / Kos pribadi</p> <p><input type="radio"/> Rumah / Kontrakan / Kos milik saudara</p> <p><input type="radio"/> Hotel / Villa (Wisatawan / pendatang)</p> <p><input type="radio"/> Pernah ke malang tetapi tidak menetap/ menginap</p> <p><input type="radio"/> Tidak pernah ke Malang</p>	

Anda diminta untuk melihat detail informasi salah satu objek layanan kesehatan yang terdapat pada sistem, sekaligus melakukan percobaan menekan tombol panggil, Navigasi, dan Tandai objek layanan kesehatan. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat denah dari Objek Layanan Kesehatan yang terdapat pada sistem sekaligus memilih salah satu objek kemudian menekan tombol Detail Informasi dan Navigasi *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk melihat daftar objek layanan kesehatan atau dokter yang telah ditandai sebelumnya. *

Berhasil
 Hampir Berhasil
 Gagal

Anda diminta untuk keluar dari akun yang sedang masuk ke dalam sistem. *

Berhasil
 Hampir Berhasil
 Gagal

Kuesioner System Usability Scale (SUS) Aplikasi Malang Sehat Modul Informasi Objek Layanan Kesehatan Untitled Section

Isilah kuesioner berikut setelah selesai menggunakan aplikasi

Saya merasa aplikasi ini akan sering saya gunakan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini terlalu kompleks (banyak hal yang tidak perlu) *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Aplikasi ini memiliki fungsi yang berguna bagi saya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Dalam menggunakan aplikasi ini saya merasa kesulitan mengakses fungsi-fungsinya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa penggunaan aplikasi ini akan dengan cepat dipahami oleh kebanyakan orang *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa membutuhkan bantuan orang yang lebih ahli dalam menggunakan aplikasi ini *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa aplikasi ini memiliki fitur seperti yang saya harapkan pada aplikasi di bidang informasi layanan kesehatan *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa fungsi-fungsi yang ada dalam aplikasi ini banyak yang TIDAK berguna *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa akan menggunakan aplikasi ini untuk seterusnya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

Saya merasa kebanyakan orang TIDAK membutuhkan aplikasi ini kedepannya *

1 2 3 4 5
 Sangat Tidak Setuju Sangat Setuju

This content is neither created nor endorsed by Google.