

SKRIPSI

**Asisten Virtual Bagi Penderita TB (Tuberkulosis) Berbasis
Android Dengan Fitur SOS Menggunakan
Metode LBS (Location Based Service)**



MILIK
PERPUSTAKAAN
ITN MALANG

Disusun Oleh :

KURNIA ILHAM CIPTO HARTONO

13.18.098

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**

2017

LEMBAR PERSETUJUAN DAN PENGESAHAN

**Asisten Virtual Bagi Penderita TB (Tuberkulosis)
Berbasis *Android* Dengan Fitur *SOS* Menggunakan
Metode *LBS (Location Based Service)***

SKRIPSI

***Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk
Memperoleh Gelar Sarjana Komputer Strata Satu (S-1)***

Disusun Oleh :

**KURNIA ILHAM CIPTO HARTONO
13.18.098**

Diperiksa dan Disetujui,

Dosen Pembimbing I

Dosen Pembimbing II

**Suryo Adi Wibowo, ST. MT.
NIP/P. 1031000438**

Nurlaily Vendvansyah, ST.

Mengetahui,

Program Studi Teknik Informatika S-1

Ketua

**Joseph Dedy Irawan, ST. MT.
NIP. 197404162005011002**

**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**

2017

PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Kurnia Ilham Cipto Hartono
NIM : 13.18.098
Program Studi : Teknik Informatika S-1
Fakultas : Fakultas Teknologi Industri
Judul Skripsi : Asisten Virtual Bagi Penderita TB (Tuberkulosis) Berbasis
Android Dengan Fitur *SOS* Menggunakan Metode *LBS*
(*Location Based Service*)

Menyatakan dengan sebenarnya bahwa penulisan skripsi ini berdasarkan hasil penelitian, pemikiran, dan pemaparan asli dari saya sendiri, baik untuk naskah laporan maupun kegiatan pemrograman yang tercantum sebagai bagian dari skripsi ini. Jika terdapat karya orang lain, saya akan mencantumkan sumber yang jelas.

Demikian pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini dan sanksi lain sesuai dengan peraturan yang berlaku di Institut Teknologi Nasional Malang.

Demikian pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun.

Malang, 12 Januari 2017

Yang membuat pernyataan,



Kurnia Ilham Cipto Hartono
NIM. 13.18.098

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT atas rahmat, taufik dan hidayah-Nya sehingga penyusunan skripsi yang berjudul “Asisten Virtual Bagi Penderita TB (Tuberkulosis) Berbasis *Android* Dengan Fitur *SOS* Menggunakan Metode *LBS (Location Based Service)*” dapat diselesaikan dengan baik. Penulis menyadari bahwa dalam proses penulisan skripsi ini banyak mengalami kendala, namun dengan adanya bantuan, bimbingan, dan kerjasama dari berbagai pihak sehingga kendala yang dihadapi tersebut dapat diatasi dengan baik. Selanjutnya ucapan terima kasih penulis sampaikan kepada:

1. Bapak Dr. Ir. Lalu Mulyadi, MTA., selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Dr. Ir. Yudi Limpraptono, MT., selaku Dekan Fakultas Teknik Industri Institut Teknologi Nasional Malang.
3. Bapak Joseph Dedy Irawan, ST. MT., selaku Ketua Program Studi Teknik Informatika Institut Teknologi Nasional Malang.
4. Bapak Suryo Adi Wibowo, ST. MT., selaku Sekertaris Program Studi Teknik Informatika Institut Teknologi Nasional Malang dan Dosen Pembimbing I yang senantiasa memberikan bimbingan, masukan, dan solusi terhadap kendala yang penulis hadapi.
5. Ibu Nurlaily Vendyansyah, ST., selaku Dosen Pembimbing II yang selalu memberikan masukan dan perbaikan terutama pada bagian penulisan laporan skripsi ini.
6. Seluruh dosen Program Studi Teknik Informatika Institut Teknologi Nasional Malang yang telah membantu dalam penulisan dan turut memberikan masukan.
7. Dr. Moch. Zainun Sp.P., selaku dokter spesialis paru yang menjadi narasumber utama penulis dalam hal penjadwalan minum obat untuk pasien Tuberkulosis dan penerima notifikasi *SOS* dari pasien.
8. Rekan-rekan di Program Studi Teknik Informatika Institut Teknologi Nasional Malang yang telah memberikan doa dan dukungannya dalam menyelesaikan skripsi ini.

Penulis menyadari masih banyak terdapat kekurangan dan kesalahan dalam penulisan laporan skripsi ini, sehingga penulis mengharapkan adanya kritik dan saran demi kesempurnaan penulisan laporan skripsi ini.

Akhir kata, semoga skripsi ini dapat bermanfaat dan memberikan wawasan bagi penulis maupun pembaca sekalian.

Malang, 12 Januari 2017

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN DAN PENGESAHAN	ii
PERNYATAAN KEASLIAN SKRIPSI	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR GAMBAR	viii
DAFTAR TABEL	x
ABSTRAK	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	3
1.5 Manfaat	4
1.6 Metodologi Penelitian	4
1.7 Sistematika Penulisan	5
BAB II TINJAUAN PUSTAKA	6
2.1 Penelitian Terkait	6
2.2 Tuberkulosis (TB/TBC)	6
2.3 Pengawas Minum Obat (PMO)	7
2.4 Obat Anti Tuberkulosis (OAT)	8
2.5 Save Our Soul (SOS)	9
2.6 Location Based Service (LBS)	9
2.7 Push Notification : Firebase Cloud Messaging (FCM)	10
2.8 Android	10
BAB III ANALISIS DAN PERANCANGAN	11
3.1 Analisis	11
3.1.1 Sistem Terdahulu	11
3.1.2 Sistem yang Akan Dibangun	11
3.1.3 Kebutuhan Fungsional	11
3.1.4 Kebutuhan Non-Fungsional	12
3.2 Perancangan	12

3.2.1	Desain Sistem	12
3.2.2	Arsitektur Informasi	13
3.2.3	Flowchart.....	15
3.2.4	Data Flow Diagram (DFD).....	17
3.2.5	Database	18
3.2.5.1	Tabel	18
3.2.5.2	Relasi Antar Tabel.....	19
3.2.6	Rancangan Layout	20
BAB IV IMPLEMENTASI DAN PENGUJIAN		28
4.1	Implementasi	28
4.1.1	Halaman Login	28
4.1.2	Halaman Register	28
4.1.3	Halaman Beranda dan Menu	29
4.1.4	Halaman Penjadwalan	30
4.1.5	Halaman SOS	31
4.1.6	Halaman Bantuan	31
4.1.7	Halaman Tentang	32
4.1.8	Notifikasi Alarm.....	32
4.1.9	Push notification SOS	33
4.1.10	Peta dan Rute.....	34
4.1.11	Publikasi di Google Play Store.....	34
4.2	Pengujian	35
4.2.1	Pengujian Fungsional	36
4.2.2	Pengujian Kompabilitas	38
4.2.3	Pengujian Akurasi Koordinat	40
4.2.4	Pengujian Waktu Respon Push Notification SOS	40
4.2.5	Pengujian User	41
BAB V PENUTUP.....		43
5.1	Kesimpulan.....	43
5.2	Saran.....	44
DAFTAR PUSTAKA		45
LAMPIRAN-LAMPIRAN.....	

DAFTAR GAMBAR

Gambar 3.1 Diagram blok sistem	13
Gambar 3.2 Arsitektur informasi pasien	14
Gambar 3.3 Arsitektur informasi kerabat dan dokter.....	14
Gambar 3.4 <i>Flowchart</i> aplikasi untuk hak akses pasien.....	15
Gambar 3.5 <i>Flowchart</i> aplikasi untuk hak akses pasien (lanjutan)	16
Gambar 3.6 <i>Flowchart</i> aplikasi untuk hak akses kerabat	17
Gambar 3.7 <i>DFD</i> level 0.....	17
Gambar 3.8 <i>DFD</i> level 1	18
Gambar 3.9 Relasi antar tabel	20
Gambar 3.10 Halaman <i>login</i>	20
Gambar 3.11 Halaman <i>register</i> pasien.....	21
Gambar 3.12 Halaman <i>register</i> kerabat	21
Gambar 3.13 Halaman beranda (menu) pasien	22
Gambar 3.14 Halaman beranda (menu) kerabat dan dokter	22
Gambar 3.15 Halaman penjadwalan	23
Gambar 3.16 Rancangan halaman <i>SOS</i> sebelum tombol ditekan	23
Gambar 3.17 Rancangan halaman <i>SOS</i> setelah tombol ditekan	24
Gambar 3.18 Rancangan halaman bantuan	24
Gambar 3.19 Rancangan halaman tentang.....	25
Gambar 3.20 Rancangan notifikasi <i>alarm</i>	25
Gambar 3.21 Rancangan tampilan <i>Push notification</i>	26
Gambar 3.22 Rancangan tampilan peta dan rute	26
Gambar 4.1 Halaman <i>login</i>	28
Gambar 4.2 Halaman <i>register</i> pasien.....	28
Gambar 4.3 Halaman <i>register</i> kerabat	29
Gambar 4.4 Halaman beranda pasien.....	29
Gambar 4.5 Halaman beranda kerabat atau pasien	30
Gambar 4.6 Halaman penjadwalan	30
Gambar 4.7 Halaman <i>SOS</i> sebelum ditekan	31

Gambar 4.8 Halaman <i>SOS</i> setelah ditekan.....	31
Gambar 4.9 Halaman bantuan.....	32
Gambar 4.10 Halaman tentang.....	32
Gambar 4.11 <i>Alarm</i> minum obat.....	33
Gambar 4.12 <i>Alarm</i> periksa ke dokter	33
Gambar 4.13 <i>Push notification SOS</i>	34
Gambar 4.14 Peta dan rute	34
Gambar 4.15 Publikasi di <i>Google Play Store</i>	35

DAFTAR TABEL

Tabel 2.1 OAT KDT kategori 1: 2	8
Tabel 2.2 OAT kombipak kategori 1: 2HRZE/4H3R3	9
Tabel 3.1 Tabel <i>devices</i>	18
Tabel 3.2 Tabel pasien	19
Tabel 3.3 Tabel kerabat.....	19
Tabel 3.4 Tabel notifikasi_alarm	19

ABSTRAK

Salah satu faktor yang menyebabkan lamanya waktu sembuh penderita Tuberkulosis (TB) adalah ketidakdisiplinan dalam hal minum obat dan periksa ke dokter. Ketidakdisiplinan ini juga dapat mengakibatkan penderita TB yang sudah sembuh menjadi kambuh lagi. Penulis ingin memberikan solusi terkait permasalahan ketidakdisiplinan penderita TB untuk minum obat dan periksa ke dokter atau rumah sakit. Penulis mengembangkan sebuah aplikasi yang bernama Asisten Virtual Bagi Penderita TB (Tuberkulosis) Berbasis *Android* Dengan Fitur *SOS* Menggunakan Metode *LBS* (*Location Based Service*).

Aplikasi Asisten virtual bagi penderita TB merupakan aplikasi *Android* yang dapat secara otomatis memberikan jadwal minum obat dan tanggal periksa bagi penderita TB. Fitur utamanya adalah memberikan notifikasi alarm jika waktunya minum obat dan waktunya periksa ke dokter. Tujuannya adalah untuk membantu para penderita TB agar disiplin minum obat dan periksa sesuai dengan jadwal yang telah ditentukan, membantu penderita TB agar sembuh dari sakitnya dengan tuntas dan tidak kambuh lagi akibat terlambat atau tidak minum obat, serta memberikan sumbangsih bagi dunia kesehatan. Fitur tambahan berupa tombol *SOS* yang dapat membantu pasien saat membutuhkan bantuan, aplikasi akan mengirimkan posisi pasien saat itu kepada kerabat pasien dan dokter yang telah terdaftar.

Berdasarkan hasil pengujian fungsional, fungsi penjadwalan, yaitu proses simpan jadwal sampai dengan notifikasi alarm dan fungsi *SOS*, mulai dari pengiriman *SOS* sampai dengan *push notification*, dapat berjalan dengan lancar saat *device* dalam kondisi *on*. Hasil pengujian fungsional menyimpulkan fungsi *login* dan *register* 100% sudah sesuai, fungsi berdasarkan hak akses 83.3% telah berjalan sesuai dengan dengan kebutuhan aplikasi, dan aplikasi tidak dapat dipasang pada *Android* dengan versi dibawah 4.0 akibat penggunaan *Android SDK* (*Software Development Kit*) minimal versi 14 dan library *FCM* (*Firebase Cloud Messaging*) 9.4 yang tidak mendukung *Android* versi rendah. Pada pengujian akurasi koordinat *SOS* (lokasi pasien) didapatkan selisih jarak rata-rata 100.4 meter pada *Google Maps V2*, 71 meter pada *Waze* serta rata-rata akumulasi jarak pada *Google Maps V2* dan *Waze* adalah 85.7 meter. Pada pengujian waktu respon *push notification SOS* didapatkan rata-rata waktu respon pada jaringan *EDGE* sebesar 2,8 detik, *H+* sebesar 1,6 detik, *4G* sebesar 1 detik, *WIFI* sebesar 1 detik, dan rata-rata secara keseluruhan adalah 1,6 detik. Hasil pengujian user menunjukkan bahwa 50% pengguna menilai tampilan aplikasi sudah baik, 62.5% pengguna menilai aplikasi mudah digunakan, 62.5% pengguna menilai fitur alarm sudah baik, 75% pengguna menilai fitur *SOS* sudah baik, dan 75% pengguna menilai aplikasi ini bermanfaat.

Kata kunci : Tuberkulosis (TB), *SOS*, *Android*, Asisten Virtual, *Location Based Service* (*LBS*)

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Tuberkulosis (TB/TBC) adalah suatu penyakit menular yang disebabkan oleh kuman dari kelompok *Mycobacterium* yaitu *Mycobacterium Tuberculosis* (Kemenkes RI, 2014). TB merupakan salah satu masalah kesehatan yang sangat serius di Indonesia. Pada tahun 2015 penderita TB di Indonesia mencapai 1 juta kasus per tahun meningkat 2 kali lipat dari tahun-tahun sebelumnya (WHO, 2015). Walaupun demikian TB bukanlah penyakit yang tidak dapat disembuhkan, salah satu faktor penting agar TB dapat sembuh secara total dan tidak kambuh lagi adalah dengan cara meminum OAT (Obat Anti Tuberkulosis) secara teratur sesuai dengan jadwal yang telah ditentukan.

Kepatuhan pasien TB dalam meminum obat sangat dipengaruhi oleh PMO (Pengawas Minum Obat) yang pada umumnya adalah keluarga atau asisten pribadi dari penderita TB, namun sebaiknya PMO adalah petugas kesehatan agar pengawasannya sesuai dengan standar yang ditetapkan dalam Pedoman TB Nasional. PMO adalah orang yang bertugas mengawasi pasien TB meminum obat secara teratur sampai selesai pengobatan, memberi dorongan kepada pasien agar mau berobat teratur, mengingatkan pasien untuk periksa ulang dahak pada waktu yang telah ditentukan, memberi penyuluhan pada anggota keluarga pasien TB yang mempunyai gejala-gejala mencurigakan TB untuk segera memeriksakan diri ke unit pelayanan kesehatan (Kemenkes RI, 2014).

Para penderita TB terkadang dapat sewaktu-waktu mengalami rasa sakit atau nyeri pada dadanya. Hal tersebut biasa terjadi pada pasien yang flek pada bagian dadanya sudah cukup parah. Rasa sakit yang tidak dapat diprediksi itu harus segera ditangani dengan segera karena dikhawatirkan paru-paru orang tersebut mengalami pendarahan atau hal lain yang diakibatkan oleh TB. Untuk memberikan pertolongan kepada pasien TB tentunya kerabat pasien dan dokter harus mengetahui dimana lokasi pasien saat ini.

Perkembangan teknologi menjadikan banyak pengembang, baik *hardware* atau *software*, mengembangkan asisten pribadi bagi seseorang, misalnya : *Home-Assistant Robot* (robot pembantu rumah tangga), *Yess Boss* (asisten virtual), dan *Google Now*

(asisten virtual). Bagi penderita TB yang mengharuskan disiplin dalam hal minum obat dan periksa ke dokter atau rumah sakit, asisten virtual sebagai PMO kedua tentu sangat dibutuhkan karena kelebihan komputer adalah lebih konsisten daripada manusia sehingga kemungkinan lupa jadwal minum obat atau periksa ke dokter dapat diminimalisir atau bahkan tidak akan terjadi sama sekali.

Penulis menggagas sebuah aplikasi asisten virtual berbasis *Android* bagi penderita TB yang dapat memberikan informasi jadwal minum obat dan tanggal periksa dengan notifikasi *alarm* disertai fitur *Save Our Soul (SOS)* yang dapat memberikan notifikasi serta koordinat lokasi pasien kepada kerabat pasien dan dokter saat pasien butuh pertolongan. Aplikasi ini diharapkan dapat menjadikan penderita TB tidak lupa jadwal minum obat dan tanggal periksa selanjutnya sehingga memperbesar rasio kesembuhan pasien TB dan memperkecil resiko kambuh lagi akibat terlambat minum obat.

1.2 Rumusan Masalah

Rumusan masalah pada pengembangan aplikasi asisten virtual bagi penderita TB, antara lain:

1. Bagaimana cara membuat otomatisasi jadwal minum obat dan periksa ke dokter bagi penderita TB dengan akses fungsi *alarm* untuk pemberitahuan saat waktunya minum obat dan periksa menggunakan aplikasi *Android*?
2. Bagaimana cara membuat fitur *SOS* yang memanfaatkan *latitude* dan *longitude* posisi pengguna dengan menggunakan aplikasi *Android* sebagai media untuk menemukan lokasi pasien dengan cepat?
3. Bagaimana cara menampilkan *push notification* posisi pengguna dengan memanfaatkan *latitude* dan *longitude* pada peta di aplikasi *Android* untuk menginformasikan posisi pengguna (pasien) kepada kerabat dan dokter saat pasien membutuhkan pertolongan?

1.3 Batasan Masalah

Penulis membuat beberapa batasan masalah pada pengembangan aplikasi asisten virtual bagi penderita TB, antara lain:

1. Aplikasi hanya dapat digunakan untuk pasien TB dewasa dengan usia minimal 18 tahun.
2. Penjadwalan minum obat diperuntukkan bagi penderita TB paru bukan ekstra paru.
3. Penjadwalan minum obat hanya untuk pasien baru TB dan pasien kambuh.
4. Penjadwalan minum obat berdasarkan panduan paket obat kombinasi dosis tetap (OAT-KDT) kategori 1 dan kombipak (paket obat lepas) kategori 1.
5. Penjadwalan minum obat pasien TB berdasarkan klasifikasi dan kriteria yang terdapat dalam Pedoman Nasional Pengendalian Tuberkulosis tahun 2014.
6. Peta yang digunakan adalah *Google Maps V2*.
7. *Push notification* menggunakan Firebase.
8. Notifikasi yang diberikan berupa *alarm* untuk pasien dan *push notification* bagi kerabat pasien dan dokter.
9. Aplikasi dapat berjalan pada *Android* minimal versi 4.0 atau minimal *Software Development Kit (SDK)* versi 14.
10. Aplikasi hanya dapat dijalankan pada mode *portrait*.
11. Terdapat 3 hak akses pada aplikasi, yaitu pasien, kerabat pasien, dan dokter.

1.4 Tujuan

Adapun tujuan dari pengembangan aplikasi asisten virtual bagi penderita TB, antara lain:

1. Mengembangkan sebuah aplikasi asisten virtual berbasis *Android* yang dapat memberikan informasi jadwal minum obat dan tanggal periksa bagi penderita TB dengan notifikasi *alarm*.
 2. Membantu para penderita TB saat butuh pertolongan dengan fitur *SOS* agar posisinya dapat cepat ditemukan.
-

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Aplikasi *Drugs Consumption Calender* (DCC) merupakan aplikasi penjadwalan minum obat berbasis *Android* bagi penderita Tuberkulosis (TB) yang dapat memberikan notifikasi *alarm* kepada pasien saat waktunya minum obat. Aplikasi DCC hanya dipcruntukkan untuk penderita TB kategori 1 yang merupakan pasien TB baru. Pasien diharuskan membuka aplikasi setiap hari untuk melakukan *check list* bahwa telah meminum obat (Amelia et al, 2016).

Aplikasi *Reminder* Pengobatan Pasien Berbasis SMS *Gateway* merupakan aplikasi yang dapat mengingatkan waktu minum obat melalui *Short Message Service* (SMS). Pasien akan mendapatkan pesan berupa SMS saat waktunya minum obat. Aplikasi tersebut dapat digunakan untuk berbagai jenis penyakit. Admin di rumah sakit harus memasukkan data pasien dan jadwal minum obatnya sesuai dengan anjuran dokter. Saat waktunya minum obat pasien akan menerima SMS yang menginformasikan obat apa saja yang harus diminum disertai dengan dosisnya (William & Sevani, 2013).

Sos Android Application adalah aplikasi untuk melakukan pencarian lokasi penting yang telah didaftarkan oleh pengguna dan mengirimkan notifikasi SMS berupa posisi pengguna atau panggilan telepon kepada pemadam kebakaran, taksi, rumah sakit, dan polisi saat membutuhkan bantuan (Poddar, P., Singh, K. & Kumar, P., 2013).

Panic Button app merupakan aplikasi yang dikembangkan untuk aktifis, mahasiswa, dan wartawan yang hidup di bawah rezim yang menindas (Griffiths, S., 2014). Cara kerja aplikasi tersebut adalah dengan menekan 5 kali tombol power atau kunci salah untuk mengirim pesan teks kepada tiga kontak yang sebelumnya telah didaftarkan oleh pengguna.

2.2 Tuberkulosis (TB/TBC)

Tuberkulosis (TB/TBC) adalah suatu penyakit menular yang disebabkan oleh kuman dari kelompok *Mycobacterium* yaitu *Mycobacterium Tuberculosis*

(Kemenkes RI, 2014). Perlu pemeriksaan secara bakteriologis untuk menentukan atau mendiagnosis seseorang terkena TB atau tidak. Tidak semua kelompok *Mycobacterium* dapat menyebabkan TB, kelompok ini dikenal dengan sebagai MOTT (*Mycobacterium Other Than Tuberculosis*). Ada beberapa cara penularan TB, antara lain (Kemenkes RI, 2014) :

- a. Sumber penularan adalah pasien TB BTA(Bakteri Tahan Asam) positif melalui percik renik dahak yang dikeluarkannya. Namun, bukan berarti bahwa pasien TB dengan hasil pemeriksaan BTA negatif tidak mengandung kuman dalam dahaknya. Hal tersebut bisa saja terjadi oleh karena jumlah kuman yang terkandung dalam contoh uji ≤ 5.000 kuman/cc dahak sehingga sulit dideteksi melalui pemeriksaan mikroskopis langsung.
- b. Pasien TB dengan BTA negatif juga masih memiliki kemungkinan menularkan penyakit TB. Tingkat penularan pasien TB BTA positif adalah 65%, pasien TB BTA negatif dengan hasil kultur positif adalah 26% sedangkan pasien TB dengan hasil kultur negatif dan foto *Toraks* positif adalah 17%.
- c. Infeksi akan terjadi apabila orang lain menghirup udara yang mengandung percik renik dahak yang infeksius tersebut.
- d. Pada waktu batuk atau bersin, pasien menyebarkan kuman ke udara dalam bentuk percikan dahak (*droplet nuclei*/percik renik). Sekali batuk dapat menghasilkan sekitar 3000 percikan dahak.

2.3 Pengawas Minum Obat (PMO)

PMO (Pengawas Minum Obat) adalah orang yang bertugas mengawasi pasien TB agar menelan obat secara teratur sampai selesai pengobatan, memberi dorongan kepada pasien agar mau berobat teratur, mengingatkan pasien untuk periksa ulang dahak pada waktu yang telah ditentukan, memberi penyuluhan pada anggota keluarga pasien TB yang mempunyai gejala-gejala mencurigakan TB untuk segera memeriksakan diri ke unit pelayanan kesehatan (Kemenkes RI, 2014). PMO sebaiknya adalah petugas kesehatan yang memahami cara penanganan TB agar pasien TB benar-benar ditangani secara benar sesuai dengan SOP (Standar Operasi dan Prosedur) Pedoman TB Nasional yang diterbitkan oleh

Kementerian Kesehatan Republik Indonesia. Namun tidak menutup kemungkinan keluarga terdekat, kader kesehatan (guru, kepala desa, atau tokoh masyarakat), atau asisten pribadi dari pasien TB dapat menjadi PMO.

2.4 Obat Anti Tuberkulosis (OAT)

Obat Anti Tuberkulosis (OAT) merupakan obat khusus untuk penderita TB yang harus diminum dalam jangka waktu tertentu. OAT terdiri atas beberapa jenis obat yang wajib diminum oleh penderita TB, jika penderita TB tidak disiplin dalam meminum OAT maka kuman TB akan menjadi kebal dan menjadi sulit disembuhkan. OAT untuk pengendalian TB diberikan secara cuma-cuma dan dikelola dengan manajemen logistik yang efektif demi menjamin ketersediaannya (Kemenkes RI, 2014). Obat utama OAT terdiri atas *rifampisin* (R), *INH/isoniazid* (H), dan *pirazinamid* (Z), dan *Etambutol* (E) yang dosisnya disesuaikan dengan kondisi TB, usia, berat badan, penyakit lain yang menyertai, serta faktor lain sesuai dengan pertimbangan dan diagnosis dokter. Panduan OAT yang digunakan di Indonesia (sesuai rekomendasi *WHO* dan Kementerian Kesehatan Republik Indonesia) untuk kombinasi dosis tetap (OAT-KDT) kategori 1 dan kombipak (paket obat lepas) kategori 1 seperti pada Tabel 2.1 dan 2.2.

Tabel 2.1 OAT KDT kategori 1: 2(HRZE)/4(HR)3 (Sumber: Pedoman Nasional Pengendalian Tuberkulosis Tahun 2014)

Berat badan	Tahap Intensif tiap hari selama 56 hari RHZE (150/75/400/275)	Tahap Lanjutan 3 kali seminggu selama 16 minggu RH (150/150)
30 – 37 kg	2 tablet 4KDT	2 tablet 2KDT
38 – 54 kg	3 tablet 4KDT	3 tablet 2KDT
55 – 70 kg	4 tablet 4KDT	4 tablet 2KDT
≥ 71 kg	5 tablet 4KDT	5 tablet 2KDT

Tabel 2.2 OAT kombipak kategori 1: 2HRZE/4H3R3 (Sumber: Pedoman Nasional Pengendalian Tuberkulosis Tahun 2014)

Tahap	Durasi (bulan)	Dosis per hari/kali				Jumlah hari/kali
		<i>Isoniasid</i> @ 300 mg	<i>Rifampisin</i> @ 450 mg	<i>Pirazina</i> <i>mid</i> @ 500 mg	<i>Etambut</i> <i>ol</i> @ 250 mg	
Intensif	2	1	1	3	3	56
Lanjutan	4	2	1	-	-	48

2.5 *Save Our Soul (SOS)*

SOS merupakan kode dari sandi *Morse* untuk menentukan sebuah tanda bahaya. Kode *SOS* digunakan ketika suatu kelompok atau seseorang membutuhkan pertolongan. Kode *SOS* merupakan kode yang dikenal secara global sehingga memungkinkan dipakai dimana saja. Awalnya *SOS* digunakan oleh pelaut dan militer untuk meminta bantuan, namun saat ini *SOS* dikenal oleh banyak orang dan diaplikasikan untuk berbagai hal. Contoh penerapan *SOS* pada aplikasi *Android* adalah untuk mengirimkan notifikasi kepada polisi, rumah sakit, pemadam kebakaran, dan keluarga saat ada kondisi darurat (Poddar, P., Singh, K. & Kumar, P., 2013).

2.6 *Location Based Service (LBS)*

LBS merupakan suatu layanan yang memiliki kemampuan mendeteksi perubahan posisi suatu objek dalam waktu yang nyata (*realtime*). *LBS* adalah layanan yang bergantung pada lokasi tertentu dan didefinisikan pula sebagai layanan informasi dengan memanfaatkan teknologi untuk mengetahui posisi sesuatu. Layanan berbasis lokasi menggunakan teknologi *positioning system*, teknologi ini memungkinkan para pengguna dapat memperoleh informasi lokasi sesuai dengan kebutuhannya. Terdapat 4 komponen dalam *LBS*, antara lain: perangkat *mobile*, jaringan komunikasi, komponen *positioning* (penunjuk posisi), serta penyedia layanan dan konten (*service and content provider*) (Anwar, B., Jaya, H., & Kusuma P.I., 2014).

2.7 *Push Notification : Firebase Cloud Messaging (FCM)*

Firebase Cloud Messaging (FCM) merupakan layanan *cross-platform* untuk berkirim pesan yang disediakan oleh *Google* secara gratis. FCM juga menyediakan fungsi untuk melakukan *push notification*, yaitu notifikasi yang muncul di bagian atas layar *smartphone* dan dapat diseret ke bawah, untuk mengakses pesan lengkapnya pengguna cukup menekan pesan yang tampil pada notifikasinya. Penggunaan Fitur *push notification* dengan FCM sangat membantu karena FCM akan mengirimkan notifikasi secara *realtime* (Google Developer, 2016).

2.8 *Android*

Android merupakan sistem operasi berbasis *Linux* (Gogoi, D., & Sharma, R.K., 2013) pada perangkat *mobile*. *Android* menyediakan *platform* yang terbuka (*open source*) bagi para pengembang untuk mengembangkan aplikasi. *Android* menyediakan *framework* aplikasi yang memungkinkan *developer* untuk membangun aplikasi inovatif dan *game* untuk perangkat *mobile* dengan menggunakan bahasa pemrograman *Java*. Perkembangan *Android* saat ini memungkinkan *developer* untuk mengembangkan aplikasi *Android* selain menggunakan *Java*, misalnya saja muncul *IDE Xamarin Studio* dengan Bahasa pemrograman *C#* dan *Basic4Android* dengan *VB.Net*.

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis

3.1.1 Sistem Terdahulu

Secara umum, saat ini penderita Tuberkulosis (TB) masih menggunakan cara manual untuk mengingat jadwal minum obat maupun periksa ke dokter. Kebanyakan dari penderita TB atau PMO (Pengawas Minum Obat) menggunakan ingatan, catatan pribadi di memo, atau catatan dari dokter. Hal tersebut tentu kurang efektif karena kemampuan ingatan manusia yang terbatas dan tidak konsisten. Jika penderita TB kambuh dan membutuhkan pertolongan belum ada media khusus yang dapat memberikan informasi kepada kerabat dan dokter. Ada beberapa aplikasi yang serupa dengan asisten virtual bagi penderita TB namun hanya terbatas pada penjadwalan KDT kategori I atau pasien baru dan tidak terdapat fitur *Save Our Soul (SOS)* jika pasien dalam kondisi darurat.

3.1.2 Sistem yang Akan Dibangun

Asisten Virtual Bagi Penderita TB (Tuberkulosis) Berbasis *Android* Dengan Fitur *SOS* Menggunakan Metode *LBS (Location Based Service)* harus mampu melakukan fungsi sebagai berikut:

1. Otomatisasi penjadwalan minum obat dan periksa ke dokter.
2. Saat waktunya minum obat dan periksa ke dokter akan ada notifikasi *alarm* yang muncul untuk mengingatkan para penderita TB.
3. Jika suatu saat pasien mengalami sesuatu atau kambuh maka disediakan tombol *SOS* yang dapat mengirimkan notifikasi serta lokasi pasien saat ini kepada kerabat dan dokter.

3.1.3 Kebutuhan Fungsional

Kebutuhan fungsional dibagi berdasarkan hak akses pengguna terhadap aplikasi. Kebutuhan fungsi antara lain :

1. Pasien
 - a. Registrasi pasien.
 - b. *Login*.
 - c. *SOS* (mengirim koordinat/lokasi pasien).

- d. Simpan jadwal minum obat dan jadwal periksa ke dokter.
 - e. Notifikasi *alarm* minum obat dan periksa ke dokter.
2. Kerabat dan dokter
 - a. Registrasi untuk kerabat.
 - b. *Login* untuk dokter dan kerabat.
 - c. *Push notification*.
 - d. Menampilkan lokasi kerabat/dokter dan pasien pada peta disertai dengan rutenya.

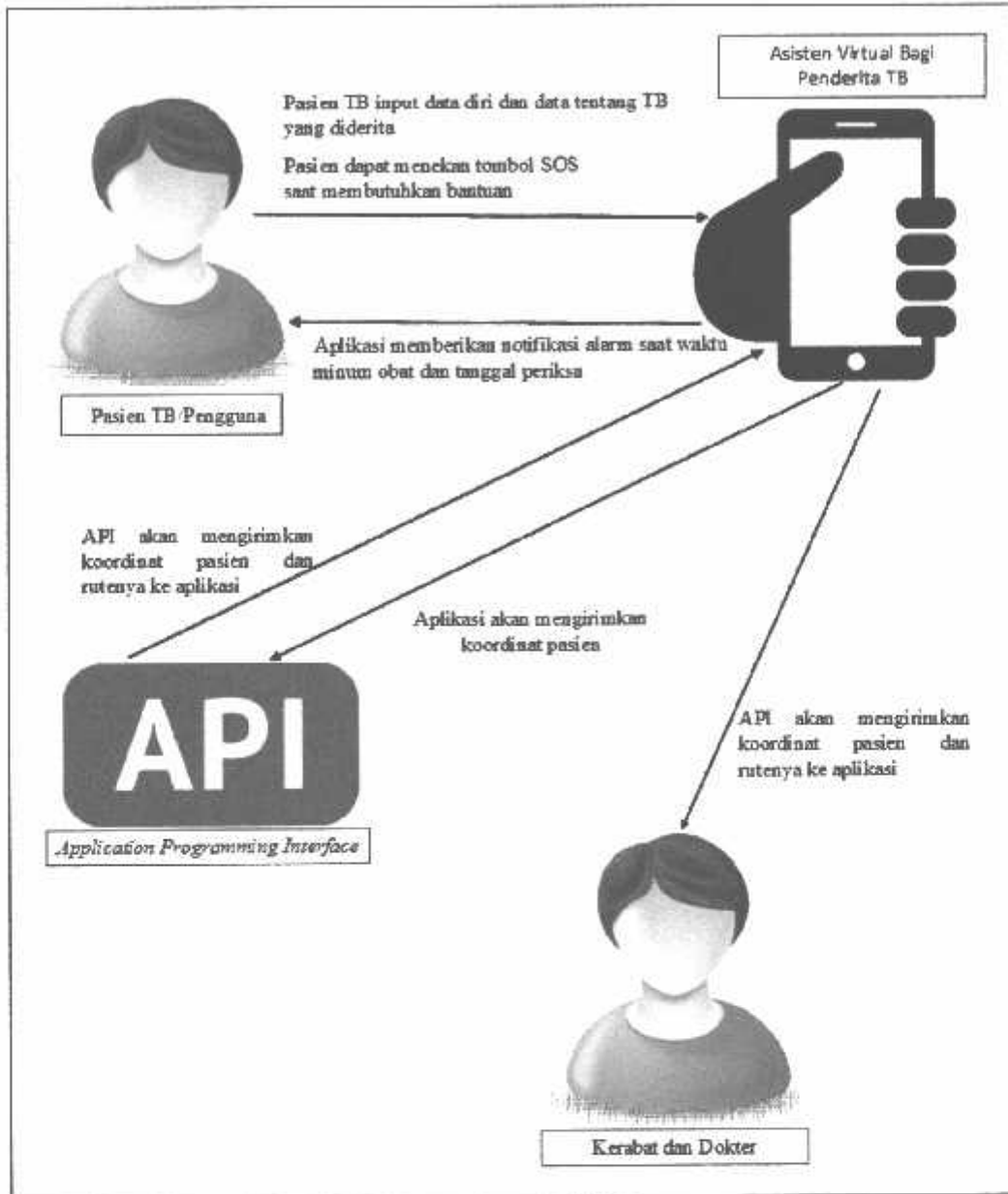
3.1.4 Kebutuhan Non-Fungsional

1. Kebutuhan Perangkat Lunak
 - a. Windows 7 Home Premium 64-bit
 - b. *Android Studio* 1.5
 - c. *Android SDK (Software Development Kit)*
 - d. ADT (*Android Development Tools*)
 - e. Xampp 5.6.3
 - f. Notepad++
 - g. Mozilla Firefox
2. Kebutuhan Perangkat Keras
 - a. Laptop: Prosesor Core i3 2.10 GHz, RAM 4 GB, Kartu grafis NVIDIA 1 GB.
 - b. *Smartphone Android Ice Cream Sandwich (ICS) 4.0* sampai dengan *Android Marshmallow (M) 6.0* dengan layar 3", 4.0", 4.5", 4.7", 5", dan 5.5".
 - c. *Tablet Android Lollipop 5.0* dengan layar 7".

3.2 Perancangan

3.2.1 Desain Sistem

Desain sistem pada aplikasi Asisten Virtual Bagi Penderita TB (Tuberkulosis) Berbasis *Android* Dengan Fitur *SOS* Menggunakan Metode *LBS (Location Based Service)* ditunjukkan pada diagram blok seperti Gambar 3.1.



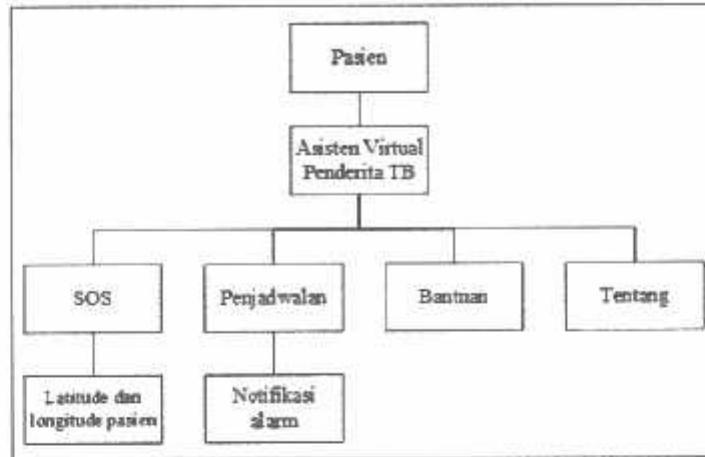
Gambar 3.1 Diagram blok sistem

3.2.2 Arsitektur Informasi

Arsitektur informasi pada sistem dibagi menjadi 2 bagian, yaitu: Arsitektur informasi pasien dan kerabat.

a. Pasien

Rancangan struktur menu dengan hak akses pasien pada aplikasi yang akan dikembangkan ditunjukkan pada Gambar 3.2.



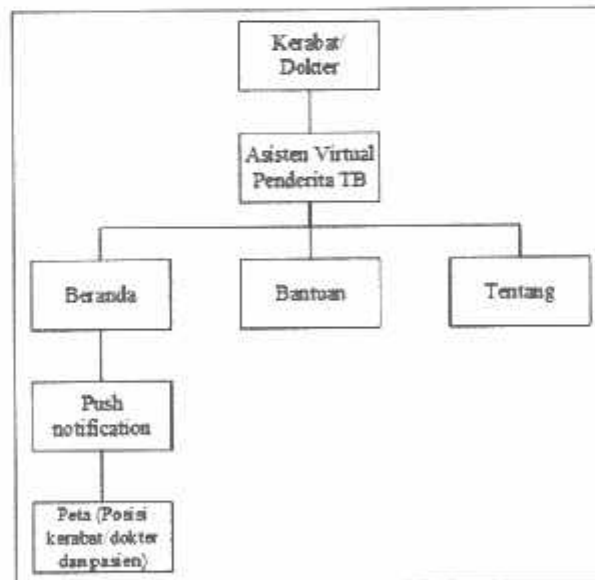
Gambar 3.2 Arsitektur informasi pasien

Keterangan :

1. *SOS* : Menu yang berisi tombol yang dapat diakses ketika pasien berada dalam kondisi darurat.
2. *Penjadwalan* : Menu untuk perhitungan otomatis jadwal minum obat dan periksa pasien TB.
3. *Bantuan* : Memuat panduan penggunaan aplikasi.
4. *Tentang* : Memuat informasi pengembang.

b. Kerabat dan dokter

Rancangan struktur menu dengan hak akses kerabat dan dokter pada aplikasi yang akan dikembangkan ditunjukkan pada Gambar 3.3.



Gambar 3.3 Arsitektur informasi kerabat dan dokter

Keterangan :

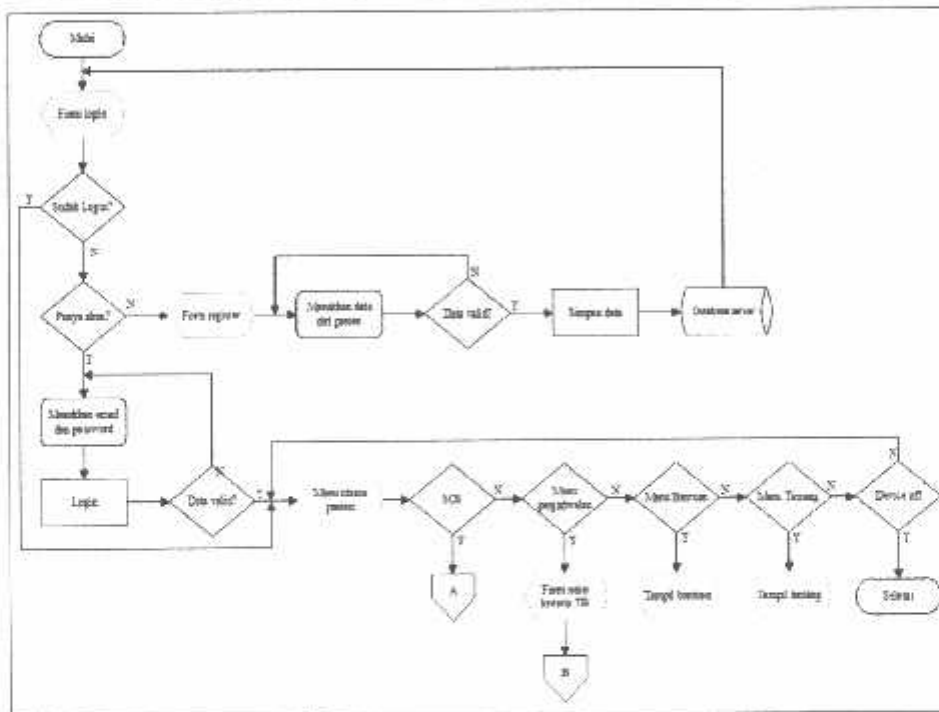
1. Beranda : Halaman utama aplikasi yang berfungsi untuk menjalankan *realtime Push notification*.
2. Bantuan : Memuat panduan penggunaan aplikasi.
3. Tentang : Memuat informasi pengembang.

3.2.3 Flowchart

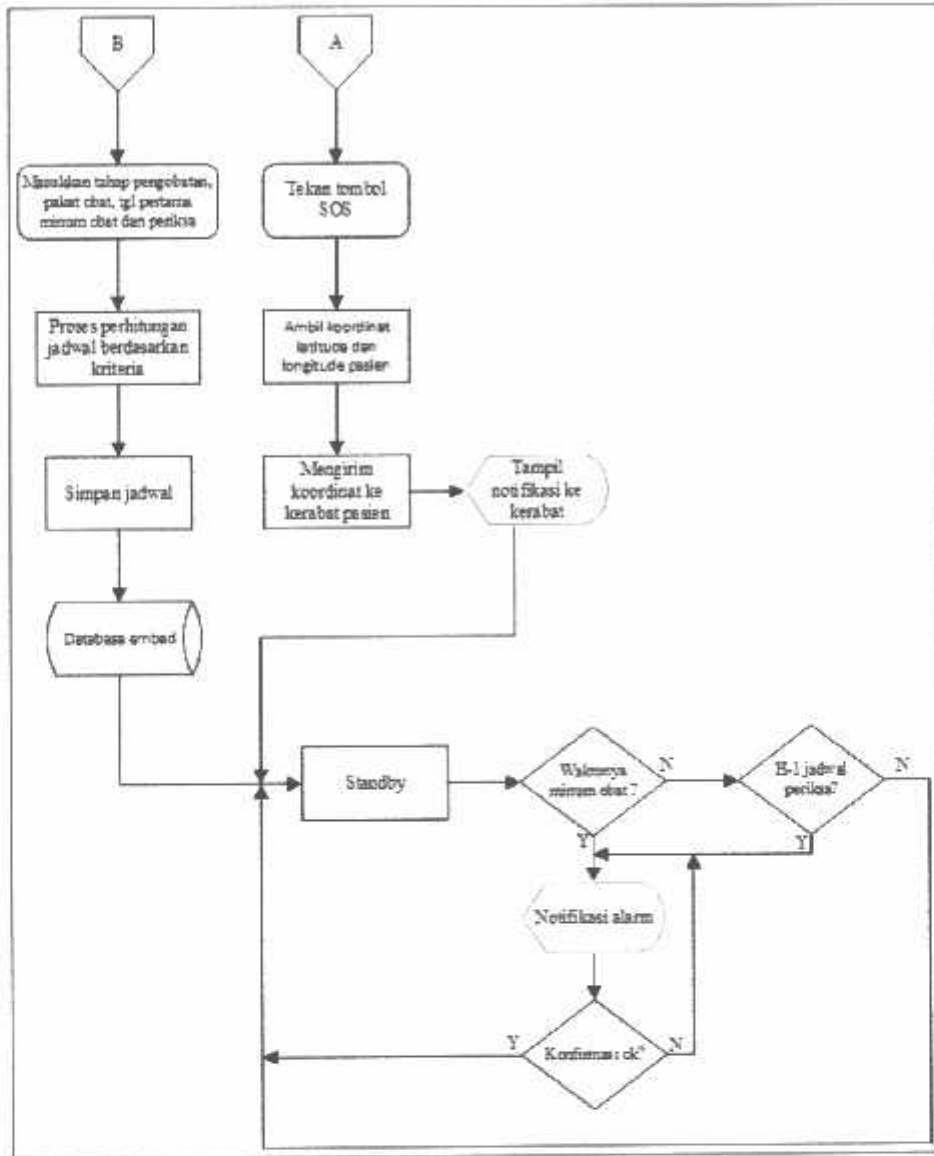
Untuk memperjelas proses-proses yang terjadi pada aplikasi penulis membagi menjadi 2 bagian, yaitu *flowchart* pada pasien serta kerabat dan dokter.

a. Pasien

Gambar 3.4 dan 3.5 merupakan *flowchart* aplikasi dengan hak akses pasien.



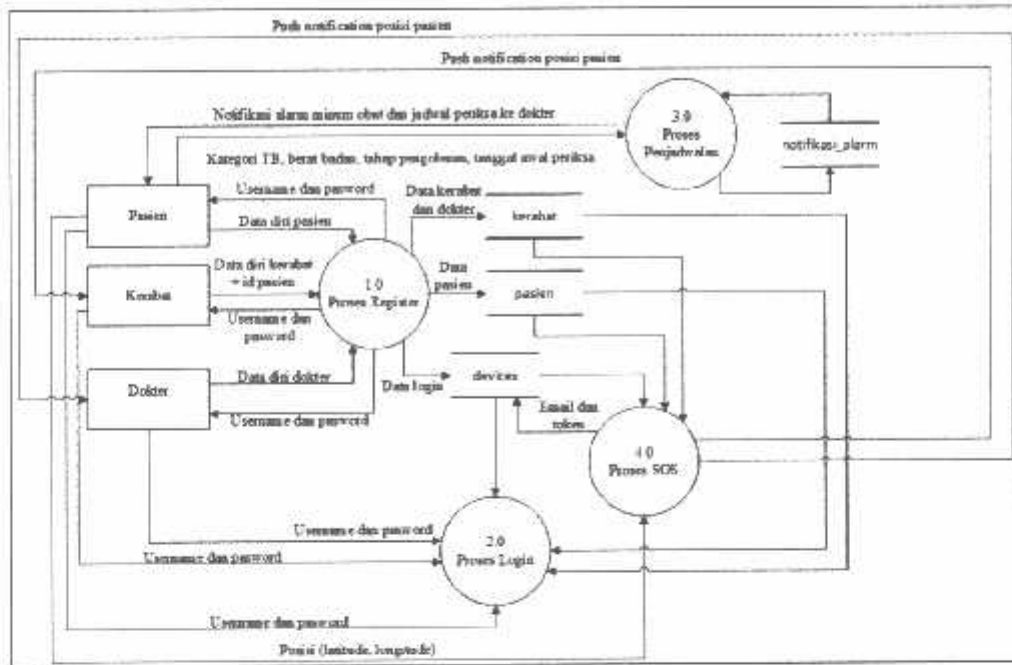
Gambar 3.4 *Flowchart* aplikasi untuk hak akses pasien



Gambar 3.5 Flowchart aplikasi untuk hak akses pasien (lanjutan)

b. Kerabat pasien dan dokter

Flowchart untuk kerabat pasien dan dokter ditunjukkan pada Gambar 3.6.



Gambar 3.8 DFD level 1

3.2.5 Database

3.2.5.1 Tabel

Terdapat 4 tabel pada sistem asisten virtual untuk penderita TB. Tabel *devices* digunakan sebagai data untuk mengirimkan *push notification* dan *login*. Tabel pasien dan kerabat dipakai sebagai penyimpanan data pasien dan kerabat agar tidak terjadi redundansi data. Tabel *notifikasi_alarm* digunakan untuk penyimpanan penjadwalan minum obat dan periksa ke dokter. Tabel 3.1, 3.2, 3.3, dan 3.4 adalah tabel-tabel yang ada pada sistem.

a. *Devices*

Tabel 3.1 Tabel *devices*

Nama Kolom	Tipe Data
<i>id</i>	<i>Integer (primary key)</i>
<i>email</i>	<i>Varchar (100)</i>
<i>token</i>	<i>Text</i>
<i>role</i>	<i>Enum (pasien, kerabat, dokter)</i>
<i>status</i>	<i>Enum (Aktif, Menunggu, Ditolak)</i>

b. Pasien

Tabel 3.2 Tabel pasien

Nama Kolom	Tipe Data
id	<i>Integer (primary key)</i>
id_pasien	<i>Varchar (10)</i>
email	<i>Varchar (100)</i>
nama	<i>Varchar (100)</i>

c. Kerabat

Tabel 3.3 Tabel kerabat

Nama Kolom	Tipe Data
id	<i>Integer (primary key)</i>
id_kerabat	<i>Varchar (10)</i>
email	<i>Varchar (100)</i>
id_pasien	<i>Varchar (10)</i>
nama	<i>Varchar (100)</i>

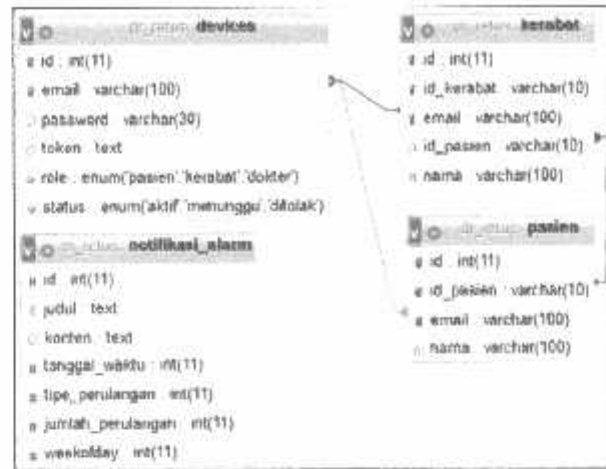
d. Notifikasi_alarm

Tabel 3.4 Tabel notifikasi_alarm

Nama Kolom	Tipe Data
id	<i>Integer (primary key)</i>
judul	<i>Text</i>
konten	<i>Text</i>
tanggal_waktu	<i>Integer</i>
tipe_perulangan	<i>Integer</i>
jumlah_perulangan	<i>Integer</i>
weekofday	<i>Integer</i>

3.2.5.2 Relasi Antar Tabel

Relasi antar tabel adalah bagan yang menggambarkan hubungan antar tabel yang satu dengan tabel yang lain sehingga membentuk basis data. Relasi antar tabel pada Gambar 3.9 menggambarkan bagaimana hubungan semua tabel yang ada dalam basis data sistem asisten virtual untuk penderita TB.



Gambar 3.9 Relasi antar tabel

3.2.6 Rancangan *Layout*

a. Halaman *login*

Halaman *login* digunakan untuk masuk ke menu utama aplikasi sesuai dengan hak aksesnya. Rancangan halaman *login* ditunjukkan seperti pada Gambar 3.10.

Gambar 3.10 Halaman *login*

b. Halaman *register*

Halaman *register* digunakan untuk mendaftarkan data pengguna. Rancangan halaman *register* pasien seperti pada Gambar 3.11 dan Gambar 3.12 merupakan halaman *register* untuk kerabat.

Gambar 3.11 Halaman *register* pasien

Gambar 3.12 Halaman *register* kerabat

c. Halaman beranda akses pasien, kerabat, dan dokter

Halaman beranda akan tampil sesuai dengan hak akses masing-masing pengguna. Rancangan halaman beranda pasien seperti pada Gambar 3.13 serta halaman beranda untuk kerabat dan dokter seperti pada Gambar 3.14.



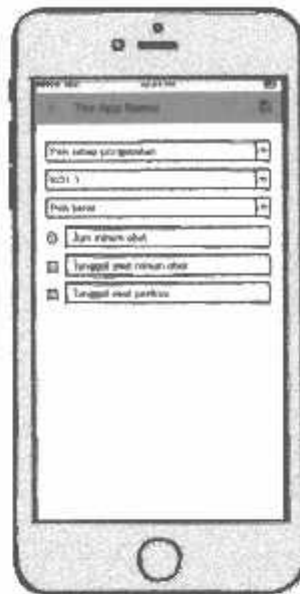
Gambar 3.13 Halaman beranda (menu) pasien



Gambar 3.14 Halaman beranda (menu) kerabat dan dokter

d. Halaman penjadwalan

Halaman penjadwalan merupakan halaman untuk menghitung jadwal minum obat dan periksa ke dokter, seperti pada Gambar 3.15.



Gambar 3.15 Halaman penjadwalan

e. *SOS*

Rancangan halaman *SOS* digunakan jika pasien dalam kondisi darurat. Jika *SOS* ditekan akan mengirimkan posisi pasien saat itu kepada kerabat dan dokter, seperti pada Gambar 3.16 dan 3.17.



Gambar 3.16 Rancangan halaman *SOS* sebelum tombol ditekan



Gambar 3.17 Rancangan halaman *SOS* setelah tombol ditekan

f. Bantuan

Halaman bantuan menampilkan informasi tentang cara menggunakan aplikasi baik dari sisi pasien maupun kerabat. Rancangan halaman bantuan seperti pada Gambar 3.18.



Gambar 3.18 Rancangan halaman bantuan

g. Tentang

Halaman bantuan menampilkan informasi tentang penulis. Rancangan halaman tentang seperti pada Gambar 3.19.



Gambar 3.19 Rancangan halaman tentang

h. Notifikasi *alarm*

Notifikasi *alarm* akan tampil saat waktunya minum obat dan periksa ke dokter. Gambar 3.20 merupakan rancangan notifikasi *alarm* minum obat.



Gambar 3.20 Rancangan notifikasi *alarm*

i. *Push notification SOS*

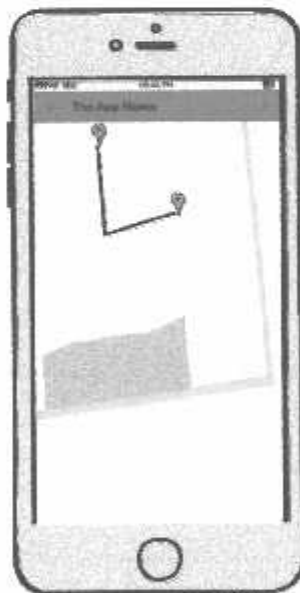
Rancangan tampilan *Push notification* ditampilkan di kerabat dan dokter saat pasien mengirimkan *SOS*, seperti pada Gambar 3.21.



Gambar 3.21 Rancangan tampilan *Push notification*

j. *Peta dan Rute*

Rancangan lokasi pasien dan kerabat atau dokter akan ditampilkan pada peta saat kerabat atau dokter menekan *push notification*, seperti pada Gambar 3.22.



Gambar 3.22 Rancangan tampilan peta dan rute

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

4.1.1 Halaman *Login*

Halaman *login* digunakan untuk mengakses menu-menu pada aplikasi sesuai dengan hak akses pengguna. Halaman *login* seperti pada Gambar 4.1.



Gambar 4.1 Halaman *login*

4.1.2 Halaman *Register*

a. Pasien

Halaman *register* pasien digunakan oleh pasien untuk mendaftarkan nama, *email*, dan kata sandi yang dapat digunakan untuk *login*. Halaman *register* pasien seperti pada Gambar 4.2.



Gambar 4.2 Halaman *register* pasien

b. Kerabat

Halaman *register* kerabat prinsipnya sama saja dengan *register* pasien, perbedaannya terletak pada id pasien yang dipakai sebagai acuan agar pasien dan kerabat dapat terhubung. Halaman *register* kerabat seperti pada Gambar 4.3.



Gambar 4.3 Halaman *register* kerabat

4.1.3 Halaman Beranda dan Menu

a. Pasien

Halaman beranda pasien adalah halaman yang hanya dapat diakses oleh pasien saat melakukan *login* sebagai pasien. Pada menu dengan hak akses pasien terdapat menu penjadwalan dan *SOS* yang sekaligus menjadi beranda. Halaman beranda dan menu pasien seperti pada Gambar 4.4.



Gambar 4.4 Halaman beranda pasien

b. Kerabat atau dokter

Halaman beranda kerabat adalah halaman yang hanya dapat diakses oleh kerabat saat melakukan *login* sebagai kerabat. Pada menu dengan hak akses kerabat menu beranda merupakan halaman yang menjalankan *service* yang bertugas untuk memberikan notifikasi saat ada *Push notification* dari pasien. Halaman beranda dan menu kerabat seperti pada Gambar 4.5.



Gambar 4.5 Halaman beranda kerabat atau pasien

4.1.4 Halaman Penjadwalan

Halaman penjadwalan adalah halaman untuk melakukan penjadwalan minum obat dan periksa ke dokter. Halaman penjadwalan seperti pada Gambar 4.6.



Gambar 4.6 Halaman penjadwalan

4.1.5 Halaman *SOS*

a. Sebelum tombol ditekan

Tombol *SOS* pada halaman *SOS* dapat digunakan saat pasien membutuhkan bantuan. Aplikasi akan mengambil posisi pasien saat itu dan mengirimkannya kepada kerabat pasien dan dokter. Tampilan *SOS* sebelum dan setelah ditekan seperti pada Gambar 4.7 dan 4.8.



Tekan Dalam Kondisi Darurat



Gambar 4.7 Halaman *SOS* sebelum ditekan

b. Setelah tombol ditekan



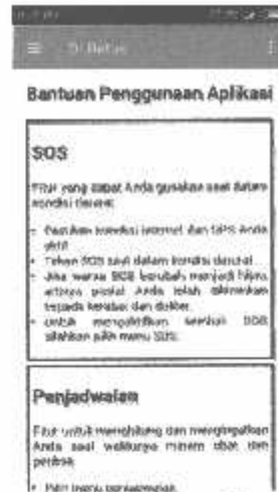
Tekan Dalam Kondisi Darurat



Gambar 4.8 Halaman *SOS* setelah ditekan

4.1.6 Halaman Bantuan

Halaman bantuan berisi panduan penggunaan aplikasi. Halaman bantuan seperti pada Gambar 4.9.



Gambar 4.9 Halaman bantuan

4.1.7 Halaman Tentang

Halaman tentang berisi informasi tentang penulis. Halaman tentang seperti pada Gambar 4.10.



Gambar 4.10 Halaman tentang

4.1.8 Notifikasi Alarm

Notifikasi *alarm* akan tampil saat waktunya minum obat dan periksa ke dokter. Notifikasi *alarm* minum obat dan periksa ke dokter seperti pada Gambar 4.11 dan 4.12.

a. *Alarm* minum obat



Gambar 4.11 *Alarm* minum obat

b. *Alarm* periksa ke dokter



Gambar 4.12 *Alarm* periksa ke dokter

4.1.9 *Push notification* SOS

Push notification akan ditampilkan di kerabat dan dokter saat pasien mengirimkan *SOS*. Tampilan *push notification* seperti pada Gambar 4.13.

store/apps/details?id=dev.elte.dokterretus. Gambar 4.15 adalah tampilan aplikasi telah terpublikasi di *Play Store*.



Gambar 4.15 Publikasi di *Google Play Store*

4.2 Pengujian

Pengujian fungsional dan kompatibilitas dilakukan menggunakan *device* dengan merk, tipe, versi *Android*, *Random Access Memory* (RAM), dan ukuran layar *device* yang berbeda-beda untuk mengetahui apakah fungsi sudah sesuai atau tidak, serta menguji kompatibilitas aplikasi terhadap berbagai *device* tersebut. Pengujian dilakukan menggunakan metode pengujian *black-box*. Tabel 4.1 merupakan daftar dari *device* yang dipakai untuk menguji aplikasi.

Tabel 4.1 *Device* pengujian

No	Nama <i>device</i>	Versi <i>Android</i>	RAM	Ukuran layar (Inch)
1	Samsung Galaxy Mini 2	<i>Gingerbread</i> 2.3.6	512 MB	3.27
2	Lenovo A390	<i>Ice Cream Sandwich</i> 4.0.4	512 MB	4.0
3	Xiaomi Redmi 1S	<i>Kitkat</i> 4.4.2	1 GB	4.7
4	Asus Zenfone 4	<i>Kitkat</i> 4.4.2	1 GB	4.0
5	Asus Zenfone 5	<i>Kitkat</i> 4.4.2	1 GB	5.0
6	Asus Fonepad 7	<i>Jelly Bean</i> 4.2	1 GB	7.0
7	Xiaomi Redmi 3	<i>Lollipop</i> 5.1.1	2 GB	5.0
8	Lenovo Vibe K5	<i>Lollipop</i> 5.1.1	2 GB	5.0

9	Meizu M2 Note	Lollipop 5.1	2 GB	5.5
10	Samsung J2 Prime	Marshmallow 6.0.1	1 GB	5.0

Pengujian dilakukan menggunakan 10 *device* dengan merk, tipe, versi *Android*, *Random Access Memory* (RAM), dan ukuran layar *device* yang berbeda-beda untuk mengetahui apakah fungsi sudah sesuai atau tidak, serta menguji komparabilitas aplikasi terhadap berbagai *device* tersebut.

4.2.1 Pengujian Fungsional

Pengujian fungsional dilakukan untuk mengetahui apakah fungsi sudah sesuai dengan kebutuhan atau belum. Pengujian ini juga dilakukan untuk mengatasi *error* dan menguji *validation* yang seringkali dilakukan tidak sesuai dengan ketentuan penggunaan aplikasi. Tabel 4.2 adalah pengujian fungsi *login* dan *register*. Tabel 4.3 merupakan pengujian fungsional berdasarkan hak akses.

Tabel 4.2 Pengujian fungsi *login* dan *register*

Fungsi	Aksi	Hasil yang diharapkan	Device		
			2	3	7
Register	<i>Input email</i> tidak sesuai dengan format	Gagal menyimpan dan memberikan notifikasi <i>email</i> tidak sesuai	√	√	√
	<i>Input password</i> dengan karakter <6	Gagal menyimpan dan memberikan notifikasi karakter <i>password</i> kurang	√	√	√
	Data <i>input</i> tidak lengkap	Gagal menyimpan dan memberikan notifikasi data tidak lengkap	√	√	√
	Tidak ada koneksi <i>internet</i>	Memberikan notifikasi bahwa tidak ada koneksi <i>internet</i> dan gagal menyimpan data	√	√	√
	Data <i>input</i> lengkap dan format sesuai	Berhasil menyimpan data dan pindah ke halaman <i>login</i>	√	√	√
Login	<i>Input email</i> salah	Gagal masuk ke aplikasi dan memberikan notifikasi gagal	√	√	√

		<i>login</i>			
	<i>Input password salah</i>	Gagal masuk ke aplikasi dan memberikan notifikasi gagal <i>login</i>	√	√	√
	Tidak ada ada koneksi <i>internet</i>	Memberikan notifikasi bahwa tidak ada koneksi <i>internet</i> dan gagal <i>login</i>	√	√	√
	<i>Input email dan password yang benar dengan hak akses user</i>	Masuk ke menu dengan akses <i>user</i>	√	√	√
	<i>Input email dan password yang benar dengan hak akses kerabat atau dokter</i>	Masuk ke menu dengan akses kerabat atau dokter	√	√	√
	<i>Session login</i>	Masuk ke aplikasi hanya dengan sekali memasukkan <i>email</i> dan <i>password</i>	√	√	√

Keterangan: √ = berhasil/scsuai

x = gagal/tidak sesuai

Fungsi *login* dan *register* 100% sudah berjalan sesuai dengan kebutuhan aplikasi.

Tabel 4.3 Pengujian berdasarkan hak akses

Fungsi	Aksi	Hasil yang diharapkan	Device		
			2	3	7
SOS (Hak akses pasien)	Menekan tombol SOS saat aplikasi gagal mendapatkan posisi pengguna	Gagal mengirim SOS dan memberikan notifikasi bahwa lokasi pengguna atau <i>internet</i> dan GPS tidak aktif	√	√	√
	Menekan tombol SOS saat GPS tidak aktif	Gagal mengirim SOS dan memberikan notifikasi bahwa lokasi pengguna atau <i>internet</i> dan GPS tidak aktif	√	√	√
	Menekan tombol SOS saat <i>internet</i> tidak aktif	Gagal mengirim SOS dan memberikan notifikasi bahwa	√	√	√

		lokasi pengguna atau <i>internet</i> dan GPS tidak aktif			
	Menekan tombol saat <i>internet</i> aktif dan posisi pengguna telah didapatkan	Berhasil mengirim posisi pasien kepada kerabat dan dokter	√	√	√
Penjadwalan (Hak akses pasien)	Data <i>input</i> tidak lengkap	Gagal menyimpan dan memberikan notifikasi data tidak lengkap	√	√	√
	<i>Input</i> tanggal sebelum hari ini	Berhasil menyimpan data	√	√	√
	Data <i>input</i> lengkap	Berhasil menyimpan data	√	√	√
Alarm (Hak akses pasien)	Aplikasi dihentikan paksa	Alarm tetap berfungsi	x	x	√
	<i>Device</i> dalam kondisi off	Alarm tetap berfungsi	x	x	x
	Keluar dari aplikasi	Alarm tetap berfungsi	√	√	√
	Aplikasi aktif dan sedang digunakan	Alarm tetap berfungsi	√	√	√
Push notification (Hak akses kerabat dan dokter)	Menekan <i>push notification</i>	Menampilkan peta, lokasi pasien, lokasi kerabat/dokter, dan rute	√	√	√
	Saat aplikasi ditutup paksa	Menerima <i>push notification</i>	x	x	√
	Saat aplikasi dibuka	Menerima <i>push notification</i>	√	√	√

Keterangan: √ = berhasil/sesuai

x = gagal/tidak sesuai

Pada pengujian fungsional berdasarkan hak akses penulis menyimpulkan bahwa 83.3% fungsi telah berjalan sesuai dengan dengan kebutuhan aplikasi.

4.2.2 Pengujian Kompabilitas

Pengujian kompabilitas dilakukan untuk menguji bagaimana kompabilitas aplikasi terhadap berbagai jenis *device Android*. Tabel 4.4 adalah pengujian kompabilitas aplikasi.

Tabel 4.4 Pengujian kompatibilitas

	Fungsi	Device									
		1	2	3	4	5	6	7	8	9	10
Halaman login	Halaman <i>login</i>	x	x	√	√	√	√	√	√	√	√
	Tombol registrasi sebagai pasien	x	x	√	√	√	√	√	√	√	√
	Tombol registrasi sebagai kerabat	x	x	√	√	√	√	√	√	√	√
Halaman register	Halaman <i>register</i>	x	x	√	√	√	√	√	√	√	√
	Tombol registrasi	x	x	√	√	√	√	√	√	√	√
	Tombol <i>login</i>	x	x	√	√	√	√	√	√	√	√
Hak akses pasien	Menu <i>SOS</i>	x	x	√	√	√	√	√	√	√	√
	Halaman <i>SOS</i>	x	x	√	√	√	√	√	√	√	√
	Tombol <i>SOS</i>	x	x	√	√	√	√	√	√	√	√
	Menu Penjadwalan	x	x	√	√	√	√	√	√	√	√
	Halaman Penjadwalan	x	x	√	√	√	√	√	√	√	√
	Tombol simpan	x	x	√	√	√	√	√	√	√	√
	Tombol kembali/batal	x	x	√	√	√	√	√	√	√	√
	Menu bantuan	x	x	√	√	√	√	√	√	√	√
	Halaman bantuan	x	x	√	√	√	√	√	√	√	√
	Menu tentang	x	x	√	√	√	√	√	√	√	√
	Profil penulis	x	x	√	√	√	√	√	√	√	√
Hak akses kerabat dan dokter	Menu beranda	x	x	√	√	√	√	√	√	√	√
	Halaman beranda	x	x	√	√	√	√	√	√	√	√
	Menu bantuan	x	x	√	√	√	√	√	√	√	√
	Halaman bantuan	x	x	√	√	√	√	√	√	√	√
	Menu tentang	x	x	√	√	√	√	√	√	√	√
	Gambar penulis	x	x	√	√	√	√	√	√	√	√

Keterangan: √ = berhasil/sesuai

x = gagal/tidak sesuai

Aplikasi tidak dapat dipasang pada *Android* dengan versi dibawah 4.0. Hal tersebut akibat penggunaan *Android SDK (Software Development Kit)* minimal versi 14 dan *library FCM (Firebase Cloud Messaging)* 9.4 yang tidak mendukung *Android* versi rendah.

4.2.3 Pengujian Akurasi Koordinat

Pengujian akurasi koordinat lokasi pasien bertujuan untuk mengetahui berapa besar selisih jarak antara posisi pasien yang sebenarnya terhadap posisi pada Google Maps V2 serta Waze. Pengujian akurasi koordinat seperti pada Tabel 4.5.

Tabel 4.5 Pengujian akurasi koordinat

No	Koodinat SOS	GM2	W	Σ
1	-7.911281, 112.629001	19	57	38
2	-7.917326, 112.629211	40	52	46
3	-7.925211, 112.625248	52	91	71.5
4	-7.944632, 112.619446	10	8	9
5	-7.634194, 112.905489	97	65	81
6	-7.645758, 112.901144	132	144	138
7	-7.647133, 112.903238	94	27	60.5
8	-7.645826, 112.901142	107	36	71.5
9	-7.672716, 112.911854	72	83	77.5
10	-7.749585, 112.993149	21	28	24.5
Σ tiap peta		64.4	59.1	61.75

Keterangan: GM2 = Selisih jarak pada *Google Maps V2* terhadap posisi sebenarnya dalam satuan meter.

W = Selisih jarak pada *Waze* terhadap posisi sbcnarnya dalam satuan meter.

Σ = Rata-rata selisih jarak yang merupakan penjumlahan antara GM2 dan W dibagi 2 dalam satuan meter.

Pada pengujian akurasi koordinat *SOS* (lokasi pasien) didapatkan selisih jarak rata-rata 64,4 meter pada *Google Maps V2* dan 59,1 meter pada *Waze*. Rata-rata akumulasi jarak pada *Google Maps V2* dan *Waze* adalah 61,75 meter.

4.2.4 Pengujian Waktu Respon *Push Notification SOS*

Pengujian waktu respon *push notification SOS* bertujuan untuk mengetahui berapa lama rata-rata waktu yang dibutuhkan *SOS* dari pasien sampai kepada kerabat atau dokter berdasarkan jenis jaringan yang didapatkan oleh *device* dokter

atau kerabat dalam satuan detik. Tabel 4.6 adalah tabel pengujian waktu respon *push notification SOS*.

Tabel 4.6 Pengujian waktu respon *push notification SOS*

Percobaan ke-	Jaringan			
	<i>EDGE</i>	<i>H+</i>	<i>4G</i>	<i>WIFI</i>
1	4	1	1	1
2	4	2	1	1
3	2	2	1	1
4	2	1	1	1
5	2	1	1	1
6	3	1	1	1
7	2	1	1	1
8	3	1	1	1
9	4	4	1	1
10	2	2	1	1
Σ waktu respon	2,8	1,6	1	1

Pada pengujian waktu respon *push notification SOS* didapatkan rata-rata waktu respon pada jaringan *EDGE* sebesar 2,8 detik, *H+* sebesar 1,6 detik, *4G* sebesar 1 detik, *WIFI* sebesar 1 detik, dan rata-rata secara keseluruhan adalah 1,6 detik.

4.2.5 Pengujian User

Pengujian *user* dilakukan terhadap 8 pasien TB. Tabel 4.7 adalah hasil pengujian terhadap *user*.

Tabel 4.7 Pengujian terhadap *user*

No	Pertanyaan	Penilaian		
		Baik	Cukup	Kurang
1	Desain/tampilan aplikasi	4	3	1
2	Kemudahan penggunaan aplikasi	5	3	-
3	Fitur <i>alarm</i> minum obat dan periksa ke dokter	5	3	-
4	Fitur <i>SOS</i>	6	2	-
5	Manfaat aplikasi	6	2	-

Hasil pengujian *user* menunjukkan bahwa 50% pengguna menilai tampilan aplikasi sudah baik, 62.5% pengguna menilai aplikasi mudah digunakan, 62.5% pengguna menilai fitur *alarm* sudah baik, 75% pengguna menilai fitur *SOS* sudah baik, dan 75% pengguna menilai aplikasi ini bermanfaat.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil analisis, perancangan, dan pengujian penulis mengambil kesimpulan sebagai berikut:

1. Berdasarkan hasil pengujian fungsional, fungsi penjadwalan, yaitu proses simpan jadwal sampai dengan notifikasi alarm dan fungsi *SOS*, mulai dari pengiriman *SOS* sampai dengan *push notification*, dapat berjalan dengan lancar saat *device* dalam kondisi *on*.
2. Pada pengujian fungsional *login* dan *register*, fungsi 100% sudah sesuai dengan kebutuhan aplikasi.
3. Pada pengujian fungsional berdasarkan hak akses penulis menyimpulkan bahwa 83.3% fungsi telah sesuai dengan dengan kebutuhan aplikasi.
4. Berdasarkan hasil pengujian kompatibilitas, aplikasi tidak dapat dipasang pada *Android* dengan versi dibawah 4.0 akibat penggunaan *Android SDK* minimal versi 14 dan *library* FCM 9.4 yang tidak mendukung *Android* versi rendah.
5. Pada pengujian akurasi koordinat *SOS* (lokasi pasien) didapatkan selisih jarak rata-rata 64,4 meter pada Google Maps V2 dan 59,1 meter pada *Waze*. Rata-rata akumulasi jarak pada Google Maps V2 dan *Waze* adalah 61,75 meter.
6. Pada pengujian waktu respon *push notification SOS* didapatkan rata-rata waktu respon pada jaringan *EDGE* sebesar 2,8 detik, *H+* sebesar 1,6 detik, *4G* sebesar 1 detik, *WIFI* sebesar 1 detik, dan rata-rata secara keseluruhan adalah 1,6 detik.
7. Hasil pengujian *user* menunjukkan bahwa 50% pengguna menilai tampilan aplikasi sudah baik, 62.5% pengguna menilai aplikasi mudah digunakan, 62.5% pengguna menilai fitur *alarm* sudah baik, 75% pengguna menilai fitur *SOS* sudah baik, dan 75% pengguna menilai aplikasi ini bermanfaat.

LAMPIRAN-LAMPIRAN

Lampiran 1. Surat Pengantar Bimbingan Skripsi Dosen Pembimbing I



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

P.T. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 561431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karangrejo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 26 September 2016

Nomor : ITN-806/IX.INF/TA/2016
Lampiran : —
Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Ibu Suryo Adi Wibowo, ST, MT.
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : KURNIA ILHAM CIPTO HARTONO
Nim : 1318098
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

26 September 2016 S/D 25 Maret 2017

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.

Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1

Ketua,



Joseph Dedy Irawan, ST., MT.
NIP. 197404162005011002

Form 5-4a



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : J. Bencungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : J. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 26 September 2016

Nomor : ITN-806/IX.INF/TA/2016
Lampiran : --
Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Ibu Nurlaily Vendyansyah, ST
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

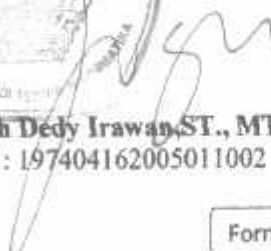
Nama : KURNIA ILHAM CIPTO HARTONO
Nim : 1318098
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

26 September 2016 S/D 25 Maret 2017

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.
Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua


Joseph Dedy Irawan, ST., MT.
NIP : 197404162005011002

Form S-4a

Malang, 26 September 2016

Lampiran : 1(Satu) berkas
Perihal : Kesiediaan sebagai Pembimbing Skripsi

Kepada : Yth. Bpk/Ibu Suryo Adi Wibowo, ST, MT.
Dosen Pembina Prodi Teknik Informatika S-1
Institut Teknologi Nasional
MALANG

Yang bertanda tangan dibawah ini:

Nama : KURNIA ILHAM CIPTO HARTONO
Nim : 1318098
Prodi : Teknik Informatika S-1

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing Utama / *Pendamping* *), untuk penyusunan Skripsi dengan judul (Proposal Terlampir) :

**Asisten Virtual Bagi Penderita TB (Tuberkolosis) Berbasis Android
Dengan Fitur SOS Menggunakan Metode LBS (Location Based Service)**

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik. Demikian permohonan kami dan atas kesiediaan Bapak/Ibu kami sampaikan terima kasih.

Prodi Teknik Informatika S-1
Institut Teknologi Nasional
MALANG



Joseph Dedy Irawan ST., MT.
NIP : 197404162005021002

Hormat Kami,



KURNIA ILHAM CIPTO HARTONO

Form S-3a

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : KURNIA ILHAM CIPTO HARTONO

Nim : 1318098

Program Studi : Teknik Informatika

Dengan ini menyatakan bersedia / ~~tidak bersedia~~ *) membimbing skripsi dari mahasiswa tersebut dengan judul :

Asisten Virtual Bagi Penderita TB (Tuberkolosis) Berbasis
Android Dengan Fitur SOS Menggunakan Metode LBS (Location
Based Service)

Demikian Surat Pernyataan ini kami buat agar dipergunakan seperlunya.

Malang, 27 September 2016

Hormat Kami,



Suryo Adi Wibowo, ST, MT.

Catatan :

Setelah disetujui agar formulir ini diserahkan mahasiswa/i
yg bersangkutan kepada Jurusan untuk diproses lebih lanjut
*) coret yang tidak perlu

Form S-3b

Malang, 26 September 2016

Lampiran : 1(Satu) berkas
Perihal : Kesiediaan sebagai Pembimbing Skripsi

Kepada : Yth. Bpk/Ibu Nurlaily Vendyansyah, ST
Dosen Pembina Prodi Teknik Informatika S-1
Institut Teknologi Nasional
MALANG

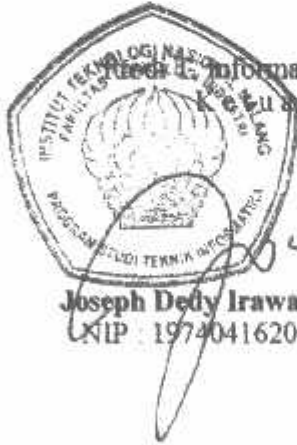
Yang bertanda tangan dibawah ini:

Nama : KURNIA ILHAM CIPTO HARTONO
Nim : 1318098
Prodi : Teknik Informatika S-1

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing (~~Utama~~ / Pendamping *), untuk penyusunan Skripsi dengan judul (Proposal Terlampir) :

**Asisten Virtual Bagi Penderita TB (Tuberkolosis) Berbasis Android
Dengan Fitur SOS Menggunakan Metode LBS (Location Based Service)**

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik. Demikian permohonan kami dan atas kesiediaan Bapak/Ibu kami sampaikan terima kasih.



Joseph Dedy Irawan, ST., MT.
NIP : 197404162005021002

Hormat Kami,

KURNIA ILHAM CIPTO HARTONO

Form S-3a

INSTITUT TEKNOLOGI NASIONAL
Jln. Bendungan Sigura-gura No. 2
Jln. Raya Karanglo Km2
MALANG

PERNYATAAN KESEDIAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : KURNIA ILHAM CIPTO HARTONO

Nim : 1318098

Program Studi : Teknik Informatika

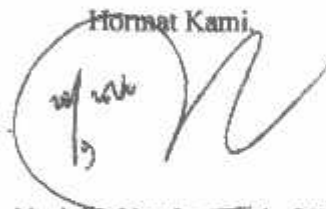
Dengan ini menyatakan bersedia / ~~tidak bersedia~~ *) membimbing skripsi dari mahasiswa tersebut dengan judul :

Asisten Virtual Bagi Penderita TB (Tuberkolosis) Berbasis
Android Dengan Fitur SOS Menggunakan Metode LBS (Location
Based Service)

Demikian Surat Pernyataan ini kami buat agar dipergunakan seperlunya.

Malang, 28 September 2016

Hormat Kami



Nurlaily Vendyansyah, ST

Catatan :

Setelah disetujui agar formulir ini diserahkan mahasiswa/i
yg bersangkutan kepada Jurusan untuk diproses lebih lanjut

*) coret yang tidak perlu

Form S-3b



BERITA ACARA SEMINAR JUDUL SKRIPSI PROGRAM STUDI TEKNIK INFORMATIKA S-1

No	Nama Mahasiswa : Kurnia Ilham C. H.	Nim	13.18.098
1	Keterangan	Tanggal 20-6-2016	Waktu
2	Pelaksanaan		Tempat Ruang Lab. Multimedid
3	Spesifikasi Judul (berilah tanda silang)**)		
	a. Jaringan komputer	c. Basis data	
	b. Multimedia	<input checked="" type="checkbox"/> d. Pemrograman & RPL	
		e. Lainnya	
4	Judul proposal yang diseminarkan mahasiswa	Asisten Virtual Bagi Penderita Tuberkulosis (TB) Berbasis Android Dengan Fitur SOS Menggunakan Metode Location Based Service (LBS)	
5	Perubahan judul yang diusulkan oleh kelompok dosen keahlian		
6	Catatan :		
7	Persetujuan Judul skripsi		
	Disetujui, Dosen keahlian I Karina A.	Disetujui, Dosen keahlian II Abdul Wahid	Disetujui, Dosen keahlian III
	Mengetahui, Ketua Prodi T.Informatika Joseph Dedy Irawan, ST.MT NIP. 19740416 200501 1 002	Moderator I Suryo Adi Wibowo	Moderator II




FORMULIR PERBAIKAN SEMINAR PROPOSAL /JUDUL

Dalam pelaksanaan Seminar Judul Jenjang Strata 1 Jurusan Teknik Informatika, maka perlu adanya perbaikan untuk mahasiswa :

Nama : Kurnia Ilham Cipto Hartono
NIM : 13.18.098

Perbaikan Meliputi :
① Tambahkan tinjauan pustaka tentang penelitian terkait penyusunan sistem penjadwalan berbasis android yg sudah ada seperti apa

Malang, 20-6-2016


(Karina Auliasari)



INSTITUT TEKNOLOGI NASIONAL MALANG
Fakultas Teknologi Industri
Program Studi Teknik Informatika S1

FORMULIR PERBAIKAN SEMINAR PROPOSAL /JUDUL

Dalam pelaksanaan Seminar Judul Jenjang Strata 1 Jurusan Teknik Informatika, maka perlu adanya perbaikan untuk mahasiswa :

Nama : Kurnia Ikhwan Cipto Hendono
NIM : 1318098
Perbaikan Meliputi : _____

1- Menyusun gambar & pengalihan gambar.

2- _____

Malang, 20-06-2016

(Abdul Wahid)

Lampiran 10. Validasi Jadwal Minum Obat Pasien TB (Tuberkulosis)
Panduan OAT yang digunakan di Indonesia (sesuai rekomendasi WHO)

1. OAT KDT Kategori I: 2(HRZE)/4(HR)3

Berat badan	Tahap Intensif tiap hari selama 56 hari RHZE (150/75/400/275)	Tahap Lanjutan 3 kali seminggu selama 16 minggu RH (150/150)
30 – 37 kg	2 tablet 4KDT	2 tablet 2KDT
38 – 54 kg	3 tablet 4KDT	3 tablet 2KDT
55 – 70 kg	4 tablet 4KDT	4 tablet 2KDT
≥ 71 kg	5 tablet 4KDT	5 tablet 2KDT

2. OAT Kombipak Kategori I: 2HRZE/4H3R3

Tahap Pengobatan	Lama Pengobatan	Dosis per hari / kali				Jumlah hari/kali menelan obat
		Tablet Isoniasid @ 300 mg	Kaplet Rifampisin @ 450 mg	Tablet Pirazinamid @ 500 mg	Tablet Etambutol @ 250 mg	
Intensif	2 Bulan	1	1	3	3	56
Lanjutan	4 Bulan	2	1	-	-	48

Menyatakan bahwa panduan OAT (Obat Anti Tuberkulosis) tersebut telah sesuai dengan rekomendasi WHO dan digunakan juga di Indonesia.

Pasuruan, 9 - November - 2016

Yang memvalidasi,


 (Dr. Moch. Zainun Sp.P.)



LEMBAR PEMANTAUAN SEMINAR PROGRES SKRIPSI

Nama : Kurnia Lham C.H.
NIM : 13.18.098
Tanggal : _____

Judul Skripsi : Asisten Virtual Bagi Penderita TB berbasis Android dengan fitur SOS menggunakan Metode LBS

Saran / Komentar :

1. jadwal periksa ke dokter
2. Review penggunaan API
3. Rute LBS

Mengetahui,
Program Studi T.Informatika
Ketua

Joseph Dedy Irawan, ST.MT
NIP. 19740416 200501 1 002

Pembimbing 1

(.....)

Pembimbing 2


NURLALY

(.....)

Penguat

(Josep Agus Pranoto)

Lampiran 12. Kuesioner Penggunaan Aplikasi

Form Kuesioner

**ASISTEN VIRTUAL BAGI PENDERITA TB (TUBERKULOSIS) BERBASIS ANDROID
DENGAN FITUR SOS MENGGUNAKAN METODE LBS (LOCATION BASED
SERVICE)**

Nama : Nuning Farida
Usia : 50 thn

Lakukan uji coba pada aplikasi dan tuliskan hasil uji coba pada tabel kuesioner, berikan tanda ceklis (v) pada kolom tabel yang dipilih :

Keterangan : B = Baik
C = Cukup
D = Kurang

Tabel Kuesioner

No	Pertanyaan	Penilaian		
		B	C	K
1	Bagaimana desain/tampilan aplikasi?		✓	
2	Bagaimana kemudahan dalam penggunaan aplikasi?	✓		
3	Bagaimana penilaian Anda terhadap fitur alarm minum obat dan periksa ke dokter?	✓		
4	Bagaimana penilaian Anda terhadap fitur SOS pada aplikasi?	✓		
5	Bagaimana penilaian Anda terhadap manfaat dari aplikasi?	✓		

Saran : Alarm kurang panjang

Nuning Farida

Form Kuesioner

**ASISTEN VIRTUAL BAGI PENDERITA TB (TUBERKULOSIS) BERBASIS ANDROID
DENGAN FITUR SOS MENGGUNAKAN METODE LBS (LOCATION BASED
SERVICE)**

Nama : AKHMAD KHOIRUL
Usia : 31

Lakukan uji coba pada aplikasi dan tuliskan hasil uji coba pada tabel kuesioner, berikan tanda ceklis (v) pada kolom tabel yang dipilih :

Keterangan : B = Baik
C = Cukup
D = Kurang

Tabel Kuesioner

No	Pertanyaan	Penilaian		
		B	C	K
1	Bagaimana desain/tampilan aplikasi?	✓	-	-
2	Bagaimana kemudahan dalam penggunaan aplikasi?	-	✓	-
3	Bagaimana penilaian Anda terhadap fitur alarm minum obat dan periksa ke dokter?	-	✓	-
4	Bagaimana penilaian Anda terhadap fitur SOS pada aplikasi?	-	✓	-
5	Bagaimana penilaian Anda terhadap manfaat dari aplikasi?	✓	-	-

Saran : ADA JADWAL DOKTERNYA.

Akhmad Khoirul

Form Kuesioner

**ASISTEN VIRTUAL BAGI PENDERITA TB (TUBERKULOSIS) BERBASIS ANDROID
DENGAN FITUR SOS MENGGUNAKAN METODE LBS (LOCATION BASED
SERVICE)**

Nama : M. MUMIR
Usia : 23

Lakukan uji coba pada aplikasi dan tuliskan hasil uji coba pada tabel kuesioner, berikan tanda ceklis (✓) pada kolom tabel yang dipilih :

Keterangan : B = Baik
C = Cukup
D = Kurang

Tabel Kuesioner

No	Pertanyaan	Penilaian		
		B	C	K
1	Bagaimana desain/tampilan aplikasi?		✓	
2	Bagaimana kemudahan dalam penggunaan aplikasi?	✓		
3	Bagaimana penilaian Anda terhadap fitur alarm minum obat dan periksa ke dokter?		✓	
4	Bagaimana penilaian Anda terhadap fitur SOS pada aplikasi?	✓		
5	Bagaimana penilaian Anda terhadap manfaat dari aplikasi?	✓		

Saran : Klo bisa. aplikasi bisa kirim pesan. ✓ hal 2 orgen

Mumir

Form Kuesioner

**ASISTEN VIRTUAL BAGI PENDERITA TB (TUBERKULOSIS) BERBASIS ANDROID
DENGAN FITUR SOS MENGGUNAKAN METODE LBS (LOCATION BASED
SERVICE)**

Nama : ZAINAL ARIFIN
Usia : 35 thn

Lakukan uji coba pada aplikasi dan tuliskan hasil uji coba pada tabel kuesioner, berikan tanda ceklis (✓) pada kolom tabel yang dipilih :

Keterangan : B = Baik
C = Cukup
D = Kurang

Tabel Kuesioner

No	Pertanyaan	Penilaian		
		B	C	K
1	Bagaimana desain/tampilan aplikasi?	✓		
2	Bagaimana kemudahan dalam penggunaan aplikasi?	✓		
3	Bagaimana penilaian Anda terhadap fitur alarm minum obat dan periksa ke dokter?	✓		
4	Bagaimana penilaian Anda terhadap fitur SOS pada aplikasi?		✓	
5	Bagaimana penilaian Anda terhadap manfaat dari aplikasi?		✓	

Saran : Aplikasinya bisa di pakai tanpa Inter net

ZAINAL A.

Form Kuesioner

**ASISTEN VIRTUAL BAGI PENDERITA TB (TUBERKULOSIS) BERBASIS ANDROID
DENGAN FITUR SOS MENGGUNAKAN METODE LBS (LOCATION BASED
SERVICE)**

Nama : Dwi Susi Rohani
Usia : 33 thn

Lakukan uji coba pada aplikasi dan tuliskan hasil uji coba pada tabel kuesioner, berikan tanda ceklis (v) pada kolom tabel yang dipilih :

Keterangan : B = Baik
C = Cukup
D = Kurang

Tabel Kuesioner

No	Pertanyaan	Penilaian		
		B	C	K
1	Bagaimana desain/tampilan aplikasi?		✓	
2	Bagaimana kemudahan dalam penggunaan aplikasi?		✓	
3	Bagaimana penilaian Anda terhadap fitur alarm minum obat dan periksa ke dokter?	✓		
4	Bagaimana penilaian Anda terhadap fitur SOS pada aplikasi?	✓		
5	Bagaimana penilaian Anda terhadap manfaat dari aplikasi?	✓		

Saran : Waktu SOS, HP pasien bunyi juga. Biar orang dekat pasien tahu.

(Dwi Susi R.)

Form Kuesioner

**ASISTEN VIRTUAL BAGI PENDERITA TB (TUBERKULOSIS) BERBASIS ANDROID
DENGAN FITUR SOS MENGGUNAKAN METODE LBS (LOCATION BASED
SERVICE)**

Nama : Siti Jamila Safitri
Usia : 23 thn

Lakukan uji coba pada aplikasi dan tuliskan hasil uji coba pada tabel kuesioner, berikan tanda ceklis (v) pada kolom tabel yang dipilih :

Keterangan : B = Baik
C = Cukup
D = Kurang

Tabel Kuesioner

No	Pertanyaan	Penilaian		
		B	C	K
1	Bagaimana desain/tampilan aplikasi?			✓
2	Bagaimana kemudahan dalam penggunaan aplikasi?		✓	
3	Bagaimana penilaian Anda terhadap fitur alarm minum obat dan periksa ke dokter?	✓		
4	Bagaimana penilaian Anda terhadap fitur SOS pada aplikasi?	✓		
5	Bagaimana penilaian Anda terhadap manfaat dari aplikasi?		✓	

Saran : Aktifasinya harusnya ada pilihan dg fitur suara

(Siti Jamila Safitri)

Form Kuesioner

**ASISTEN VIRTUAL BAGI PENDERITA TB (TUBERKULOSIS) BERBASIS ANDROID
DENGAN FITUR SOS MENGGUNAKAN METODE LBS (LOCATION BASED
SERVICE)**

Nama : Urwati Chasanah
Usia : 31 tahun

Lakukan uji coba pada aplikasi dan tuliskan hasil uji coba pada tabel kuesioner, berikan tanda ceklis (v) pada kolom tabel yang dipilih :

Keterangan : B = Baik
C = Cukup
D = Kurang

Tabel Kuesioner

No	Pertanyaan	Penilaian		
		B	C	K
1	Bagaimana desain/tampilan aplikasi?	✓		
2	Bagaimana kemudahan dalam penggunaan aplikasi?	✓		
3	Bagaimana penilaian Anda terhadap fitur alarm minum obat dan periksa ke dokter?		✓	
4	Bagaimana penilaian Anda terhadap fitur SOS pada aplikasi?	✓		
5	Bagaimana penilaian Anda terhadap manfaat dari aplikasi?	✓		

Saran : alarnya rumah sakit terdekat

Urwati Chasanah

Form Kuesioner

**ASISTEN VIRTUAL BAGI PENDERITA TB (TUBERKULOSIS) BERBASIS ANDROID
DENGAN FITUR SOS MENGGUNAKAN METODE LBS (LOCATION BASED
SERVICE)**

Nama : M. Alfin R.
Usia : 24

Lakukan uji coba pada aplikasi dan tuliskan hasil uji coba pada tabel kuesioner, berikan tanda ceklis (v) pada kolom tabel yang dipilih :

Keterangan : B = Baik
C = Cukup
D = Kurang

Tabel Kuesioner

No	Pertanyaan	Penilaian		
		B	C	K
1	Bagaimana desain/tampilan aplikasi?	✓		
2	Bagaimana kemudahan dalam penggunaan aplikasi?	✓		
3	Bagaimana penilaian Anda terhadap fitur alarm minum obat dan periksa ke dokter?	✓		
4	Bagaimana penilaian Anda terhadap fitur SOS pada aplikasi?	✓		
5	Bagaimana penilaian Anda terhadap manfaat dari aplikasi?	✓		

Saran : lebih dikembangkan lagi !

M. Alfin R.



FORMULIR BIMBINGAN SKRIPSI

Nama : Kurnia Ilham Cipto Hartono
 NIM : 13.18.098
 Masa Bimbingan : 26 September 2016 s/d 25 Maret 2017
 Judul Skripsi : Asisten Virtual Bagi Penderita TB (Tuberkulosis) Berbasis *Android* Dengan Fitur *SOS* Menggunakan Metode *LBS (Location Based Service)*

No	Tanggal	Uraian	Paraf Pembimbing
1	17/10/2016	Latar belakang, rumusan masalah, batasan masalah, tujuan	
2	03/11/2016	Tinjauan pustaka, penelitian terkait	
3	10/11/2016	Analisis sistem saat ini, sistem yang akan dibangun, perancangan <i>mockup</i> ,	
4	17/11/2016	kebutuhan fungsional, kebutuhan non fungsional	
5	23/11/2016	Demo aplikasi <i>Android</i> , menambah hak akses dokter, <i>flowchart</i> , <i>data flow diagram</i> .	
6	29/11/2016	Demo aplikasi <i>Android</i> , cetak <i>italic</i> pada istilah asing, penulisan jurnal	
7	05/12/2016	Demo aplikasi <i>Android</i> , revisi <i>flowchart</i> dan <i>dfd</i>	
8	14/12/2016	pengujian fungsional, komparabilitas, akurasi koordinat	
9	04/01/2017	Abstrak, revisi tujuan dan manfaat	
10	10/01/2017	Demo aplikasi <i>Android</i> , revisi halaman bantuan	

Malang, 11 Januari 2017
 Dosen Pembimbing

(Suryo Adi Wobowo., ST. MT.)

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

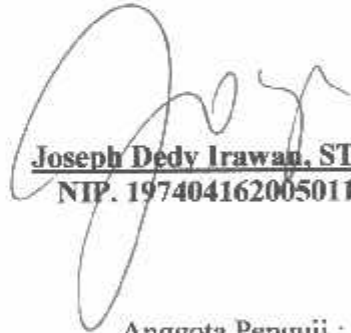
NAMA : Kurnia Ilham Cipto Hartono
NIM : 1318190
JURUSAN : Teknik Informatika S-1
JUDUL : Asisten Virtual Bagi Penderita TB (Tuberkulosis) Berbasis *Android*
Dengan Fitur *SOS* Menggunakan Metode *LBS (Location Based Service)*

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari : Selasa
Tanggal : 17 Januari 2017
Nilai : 91 (A)

Panitia Ujian Skripsi :

Ketua Majelis Penguji



Joseph Dedy Irawan, ST. MT.
NIP. 197404162005011002

Anggota Penguji :

Dosen Penguji I



Yosep Agus Pranoto, ST. MT.
NIP. P. 1031000432

Dosen Penguji II





Sandy Nataly Mantja, S.Kom.
NIP. P. 1030800418

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata I Program Studi Teknik Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : Kurnia Ilham Cipto Hartono
 NIM : 1318190
 JURUSAN : Teknik Informatika S-1
 JUDUL : Asisten Virtual Bagi Penderita TB (Tuberkulosis) Berbasis *Android* Dengan Fitur *SOS* Menggunakan Metode *LBS (Location Based Service)*

No	Penguji	Tanggal	Uraian	Paraf
1.	Penguji I	17 Januari 2017	1. Program ok 2. Revisi tabel dan gambar di bab II harus ada sumber (sitasi) 3. Revisi urutan nama gambar	
2.	Penguji II	17 Januari 2017	1. Program ok 2. Revisi metodologi penelitian	

Dosen Penguji I



Yosep Agus Pranoto, ST. MT.
NIP. P. 1031000432

Dosen Penguji II



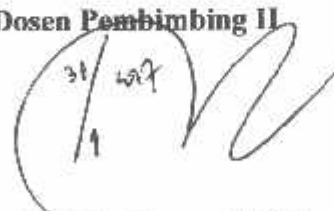
Sandy Nataly Mantja, S.Kom.
NIP. P. 1030800418

Dosen Pembimbing I



Suryo Adi Wibowo, ST. MT.
NIP. P. 1031000438

Dosen Pembimbing II



Nurlaily Vandyansyah, ST.



FORMULIR PERBAIKAN UJIAN SKRIPSI


Dalam pelaksanaan Ujian Skripsi Jenjang Strata 1 Jurusan Teknik Informatika, maka perlu adanya perbaikan untuk mahasiswa :

Nama : Kurnia Uham Cipto Hartono
NIM : 13.18.090
Perbaikan Meliputi : _____

1. Tabel dan gambar di bab II harus ada sumber (sitas)

2. Urutan nama gambar.

Malang, 17 Januari 2017


(Yusap Agus P)

Lampiran 17. Source Code

Source code dapat diunduh di: goo.gl/E0BftK dan <https://goo.gl/6gelp6>

1. Source code file .java

1.1 Folder utils

a. Constants.java

```
package dev.elte.dokterretus.utills;

/**
 * Created by LT on 10/23/2016.
 */
public class Constants {
    public static final String IP =
"http://elte.dev.xyz/";
    public static final String LOGIN_URL = IP +
"login.php";
    public static final String URL_REGISTER_PASIEN = IP +
"register_pasien.php";
    public static final String URL_REGISTER_KERABAT = IP +
"register_kerabat.php";
    public static final String URL_SEND_MULTIPLE_PUSH = IP +
"sendMultiplePush.php";
    public static final String URL_FETCH_DEVICES = IP +
"GetRegisteredDevices.php";
    public static final String KODE_KERABAT_URL = IP +
"ambil_kode_kerabat.php";
    public static final String KODE_PASIEN_URL = IP +
"ambil_kode_pasien.php";
    public static final String ROLE_URL = IP +
"ambil_role.php";

    public static final String JSON_ARRAY = "result";
    public static final String KEY_ID_PASIEN = "id_pasien";
    public static final String KEY_ID_KERABAT = "id_kerabat";
    public static final String KEY_NAMA = "nama";
    public static final String KEY_EMAIL = "email";
    public static final String KEY_PASSWORD = "password";
    public static final String KEY_ROLE = "role";
    public static final String KEY_STATUS = "status";
    public static final String KEY_ID1 = "id1";
    public static final String KEY_ID2 = "id2";
    public static final String KEY_TOKEN = "token";

    //key for scheduling
    public static final String KEY_INTENSITAS = "intensitas";
    public static final String KEY_PESAN_OBAT = "Waktunya minum
obat....";
    public static final String KEY_PESAN_PERIKSA= "Waktunya periksa ke
dokter....";
    public static final String KEY_PESAN_PERIKSA2= "Hari ini adalah
jadwal periksa Anda";

    //rule minum obat
    //jenis obat KDT1 tahap Intensif
    public static final String jenisObatKDT1Intensif1 = "2 tablet 4KDT";
    public static final String jenisObatKDT1Intensif2 = "3 tablet 4KDT";
    public static final String jenisObatKDT1Intensif3 = "4 tablet 4KDT";
    public static final String jenisObatKDT1Intensif4 = "5 tablet 4KDT";

    //jenis obat KDT1 tahap Lanjutan
    public static final String jenisObatKDT1lanjutan1 = "2 tablet 2KDT";
    public static final String jenisObatKDT1lanjutan2 = "3 tablet 2KDT";
    public static final String jenisObatKDT1lanjutan3 = "4 tablet 2KDT";
    public static final String jenisObatKDT1lanjutan4 = "5 tablet 2KDT";

    //jenis obat Kombipaki tahap Intensif dan lanjutan
```

```

public static final String jenisObatKombipaklintensif = "1 Tablet
isoniasid @ 300 mgr \n1 Kaplet Rifampisin @450 mgr \n3 Tablet Pirazinamid
@500 mgr \n3 Tablet Etambutol @ 250 mgr";
public static final String jenisObatKombipakllanjutan = "2 Tablet
isoniasid @ 300 mgr \n1 Kaplet Rifampisin @450 mgr";
}

```

b. MyVolley.java

```

package dev.elte.dokterretus.utils;

import android.content.Context;

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.toolbox.Volley;

/**
 * Created by Belal on 13/10/16.
 */

public class MyVolley {

    private static MyVolley mInstance;
    private RequestQueue mRequestQueue;
    private static Context mCtx;

    private MyVolley(Context context) {
        mCtx = context;
        mRequestQueue = getRequestQueue();
    }

    public static synchronized MyVolley getInstance(Context context) {
        if (mInstance == null) {
            mInstance = new MyVolley(context);
        }
        return mInstance;
    }

    public RequestQueue getRequestQueue() {
        if (mRequestQueue == null) {
            // getApplicationContext() is key, it keeps you from leaking
            // the
            // Activity or BroadcastReceiver if someone passes one in.
            mRequestQueue =
Volley.newRequestQueue(mCtx.getApplicationContext());
        }
        return mRequestQueue;
    }

    public <T> void addToRequestQueue(Request<T> req) {
        getRequestQueue().add(req);
    }
}

```

1.2 Folder services

a. MyNotificationManager.java

```

package dev.elte.dokterretus.services;

//Created by Belal on 03/11/16.
//Created by Ravi on 31/03/15.

import android.app.Notification;
import android.app.NotificationManager;

```

```

import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.net.Uri;
import android.support.v4.app.NotificationCompat;
import android.text.Html;

import java.io.IOException;
import java.io.InputStream;
import java.net.HttpURLConnection;
import java.net.URL;

import dev.elte.dokterretus.R;

public class MyNotificationManager {

    public static final int ID_SMALL_NOTIFICATION = 235;

    private Context mContext;
    Uri u= Uri.parse("android.resource://dev.elte.dokterretus/raw/sos" );

    public MyNotificationManager(Context mContext) {
        this.mContext = mContext;
    }

    public void showSmallNotification(String title, String message,
Intent intent) {
        PendingIntent resultPendingIntent =
            PendingIntent.getActivity(
                mContext,
                ID_SMALL_NOTIFICATION,
                intent,
                PendingIntent.FLAG_UPDATE_CURRENT
            );

        NotificationCompat.Builder mBuilder = new
NotificationCompat.Builder(mContext);
        Notification notification;
        notification =
mBuilder.setSmallIcon(R.mipmap.ic_launcher).setTicker(title).setWhen(0)
            .setSound(u)
            .setAutoCancel(true)
            .setContentIntent(resultPendingIntent)
            .setContentTitle(title)
            .setSmallIcon(R.mipmap.ic_launcher)

        .setLargeIcon(BitmapFactory.decodeResource(mContext.getResources(),
R.mipmap.ic_launcher))
            .setContentText(message)
            .build();

        notification.flags |= Notification.FLAG_AUTO_CANCEL;

        NotificationManager notificationManager = (NotificationManager)
mContext.getSystemService(Context.NOTIFICATION_SERVICE);
        notificationManager.notify(ID_SMALL_NOTIFICATION, notification);
    }
}

```

1.3 Folder receivers

a. AlarmReceiver.java

```

package dev.elte.dokterretus.receivers;
import android.app.AlarmManager;

```

```

import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.support.v4.content.LocalBroadcastManager;
import android.support.v4.content.WakefulBroadcastReceiver;

import dev.elte.dokterretus.database.DatabaseHelper;
import dev.elte.dokterretus.models.Reminder;
import dev.elte.dokterretus.services.MyFirebaseMessagingService;
import dev.elte.dokterretus.utils.AlarmUtil;
import dev.elte.dokterretus.utils.NotificationUtil;

public class AlarmReceiver extends WakefulBroadcastReceiver {

    @Override
    public void onReceive(Context context, Intent intent) {
        DatabaseHelper database = DatabaseHelper.getInstance(context);
        Reminder reminder =
database.getNotification(intent.getIntExtra("NOTIFICATION_ID", 0));
        reminder.setNumberShown(reminder.getNumberShown() + 1);
        database.addNotification(reminder);

        NotificationUtil.createNotification(context, reminder);

        AlarmManager am = (AlarmManager)
context.getSystemService(Context.ALARM_SERVICE);
        PendingIntent pi = PendingIntent.getService(context, 0, new
Intent(context, MyFirebaseMessagingService.class),
PendingIntent.FLAG_UPDATE_CURRENT);
        am.setInexactRepeating(AlarmManager.RTC_WAKEUP,
System.currentTimeMillis() + 0, 0, pi);

        // Check if new alarm needs to be set
        if (reminder.getNumberToShow() > reminder.getNumberShown() ||
Boolean.parseBoolean(reminder.getForeverState())) {
            AlarmUtil.setNextAlarm(context, reminder, database);
        }

        Intent updateIntent = new Intent("BROADCAST_REFRESH");
LocalBroadcastManager.getInstance(context).sendBroadcast(updateIntent);

        database.close();
    }
}

```

1.4 Folder fragments

a. Maps.java

```

package dev.elte.dokterretus.fragments;

import android.Manifest;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Color;
import android.location.Criteria;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.net.Uri;
import android.os.Bundle;
import android.support.v4.app.ActivityCompat;
import android.support.v7.app.AppCompatActivity;
import android.view.MenuItem;

```

```

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;
import com.google.android.gms.appindexing.Action;
import com.google.android.gms.appindexing.AppIndex;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;

/**
 * Created by LT on 9/13/2016.
 */
public class Maps extends AppCompatActivity implements
    OnMapReadyCallback, LocationListener,
    ConnectivityReceiver.ConnectivityReceiverListener {

    private GoogleApiClient googleApiClient;
    private GoogleMap mMap;
    private Criteria criteria;
    private LocationManager locationManager;
    private String provider;
    private Location location;
    private Intent intent;
    LatLng latLng, positionPasien;
    double latPasien, longPasien;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);

        if(MainActivity2.fa != null){
            MainActivity2.fa.finish();
        }

        if (getSupportActionBar() != null){
            getSupportActionBar().setDisplayHomeAsUpEnabled(true);
            getSupportActionBar().setDisplayShowHomeEnabled(true);
        }

        SupportMapFragment mapFragment = (SupportMapFragment)
        getSupportFragmentManager()
            .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);

        googleApiClient = new GoogleApiClient.Builder(this)
            .addApi(LocationServices.API)
            .addApi(AppIndex.API).build();
        if (ActivityCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_FINE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED &&
        ActivityCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_COARSE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED) {
            return;
        }
    }
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {

```

```

        if (item.getItemId() == android.R.id.home) {
            onBackPressed();
        }

        return super.onOptionsItemSelected(item);
    }

    @Override
    public void onLocationChanged(Location location) {
    }

    @Override
    public void onMapReady(GoogleMap googleMap) {

        String posisi = getIntent().getStringExtra("msg");

        String[] latLngPasien = posisi.split(",");
        latPasien = Double.parseDouble(latLngPasien[0]);
        longPasien = Double.parseDouble(latLngPasien[1]);

        mMap = googleMap;

        googleMap.setMyLocationEnabled(true);

        // Getting LocationManager object from System Service
        LOCATION_SERVICE
        criteria = new Criteria();
        locationManager = (LocationManager)
        getSystemService(Context.LOCATION_SERVICE);
        provider = locationManager.getBestProvider(criteria, true);
        if (ActivityCompat.checkSelfPermission(this,
        android.Manifest.permission.ACCESS_FINE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED &&
        ActivityCompat.checkSelfPermission(this,
        android.Manifest.permission.ACCESS_COARSE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED) {
            return;
        }
        location =
        locationManager.getLastKnownLocation(LocationManager.NETWORK_PROVIDER);

        if (location != null) {
            onLocationChanged(location);
        }

        locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER,
        500, 0, this);

        latLng = new LatLng(location.getLatitude(),
        location.getLongitude());
        positionPasien = new LatLng (latPasien, longPasien);

        getDirection();

        mMap.addMarker(new MarkerOptions().position(latLng).
        icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_azure)).
            title("Posisi Anda"));
        mMap.addMarker(new MarkerOptions().position(positionPasien).
        icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED)).
            title("Posisi Pasien"));
    }

```

```

        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
        mMap.animateCamera(CameraUpdateFactory.zoomTo(12));
    }

    @Override
    protected void onStart() {
        googleApiClient.connect();
        super.onStart();
        Action viewAction = Action.newAction(
            Action.TYPE_VIEW, // TODO: choose an action type.
            "Maps Page", // TODO: Define a title for the content
            Uri.parse("http://host/path"),
            // TODO: Make sure this auto-generated app deep link URI
            // is correct.
            Uri.parse("android-
app://dev.elte.dokterretus/http/host/path")
        );
        AppIndex.AppIndexApi.start(googleApiClient, viewAction);
    }

    private void getDirection() {
        String url = makeURL(location.getLatitude(),
location.getLongitude(), latPasien, longPasien);

        final ProgressDialog loading = ProgressDialog.show(this, "",
"Mendapatkan lokasi pasien...", false, false);

        StringRequest stringRequest = new StringRequest(url,
            new Response.Listener<String>() {
                @Override
                public void onResponse(String response) {
                    loading.dismiss();
                    drawPath(response);
                }
            },
            new Response.ErrorListener() {
                @Override
                public void onErrorResponse(VolleyError error) {
                    loading.dismiss();
                }
            }
        );

        RequestQueue requestQueue = Volley.newRequestQueue(this);
        requestQueue.add(stringRequest);
    }
}

```

1.5 Folder database

a. DatabaseHelper.java

```

package dev.elte.dokterretus.database;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import java.util.ArrayList;
import java.util.List;

import dev.elte.dokterretus.R;
import dev.elte.dokterretus.models.Reminder;

```

```

private static RemindMe mInstance;
@Override
public void onCreate() {
    super.onCreate();
    mInstance = this;
}
public static synchronized RemindMe getInstance() {
    return mInstance;
}
}

```

1.7 Folder activities

a. CreateSchedule.java

```

package dev.elte.dokterretus.activities;

import android.app.DatePickerDialog;
import android.app.TimePickerDialog;
import android.content.Intent;
import android.graphics.Color;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;

import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;

public class CreateSchedule extends AppCompatActivity implements
    AdvancedRepeatSelector.AdvancedRepeatSelectionListener,
    DaysOfWeekSelector.DaysOfWeekSelectionListener,
    RepeatSelector.RepeatSelectionListener {

    private String icon;
    private String colour;
    private Calendar calendar;
    private boolean[] daysOfWeek = new boolean[?];
    private int timesShown = 0;
    private int timesToShow = 1;
    private int repeatType;
    private int id;
    private int interval = 1;

    private Spinner spinJenis, spinTahap, spinBerat, spinPola;
    private LinearLayout spin1Row, spin2Row, spin3Row, spin4Row,
    repeat_row, time_row;

    private String pesan, tanggal, jam;
    private int intensitas, temp_intensitas;
    private Intent intent;
    private Calendar nowCalendar;
    private DatabaseHelper database;
    private Reminder reminder;
    private Bundle bundle;
    private Bundle b;

    public static String[] namesOfDays = {"Minggu", "Senin", "Selasa",
    "Rabu", "Kamis", "Jumat", "Sabtu"};
    public static String[] namesOfMonths = new String[] { "Januari",
    "Februari", "Maret", "April", "Mei", "Juni", "Juli", "Agustus",
    "September", "Oktober", "November", "Desember" };

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_create_schedule);
    }
}

```

```

ButterKnife.bind(this);

spinJenis = (Spinner) findViewById(R.id.spin_jenis);
spinTahap = (Spinner) findViewById(R.id.spin_tahap);
spinBerat = (Spinner) findViewById(R.id.spin_berat);
spinPola = (Spinner) findViewById(R.id.spin_pola);

spin1Row = (LinearLayout) findViewById(R.id.spin1_row);
spin2Row = (LinearLayout) findViewById(R.id.spin2_row);
spin3Row = (LinearLayout) findViewById(R.id.spin3_row);
spin4Row = (LinearLayout) findViewById(R.id.spin4_row);
repeat_row = (LinearLayout) findViewById(R.id.repeat_row);
time_row = (LinearLayout) findViewById(R.id.time_row);

pesan = txtPesan.getText().toString().trim();
tanggal = dateText.getText().toString().trim();
jam = timesText.getText().toString().trim();

setSupportActionBar(toolbar);
toolbar.setNavigationIcon(R.drawable.ic_arrow_back_white_24dp);
if (getActionBar() != null)
getActionBar().setDisplayHomeAsUpEnabled(true);
if (getSupportActionBar() != null)
getSupportActionBar().setTitle(null);

calendar = Calendar.getInstance();
icon = getString(R.string.default_icon_value);
colour = getString(R.string.default_colour_value);

spin3Row.setVisibility(View.GONE);
spin4Row.setVisibility(View.GONE);
repeat_row.setVisibility(View.GONE);

id = getIntent().getIntExtra("NOTIFICATION_ID", 0);

if (id == 0) {
    DatabaseHelper database = DatabaseHelper.getInstance(this);
    id = database.getLastNotificationId() + 1;
    database.close();
} else {
    assignReminderValues();
}

bundle = getIntent().getExtras();
intensitas = bundle.getInt(Constants.KEY_INTENSITAS);
Toast.makeText(getApplicationContext(),
String.valueOf(intensitas), Toast.LENGTH_LONG).show();
if(intensitas==99 || intensitas==98 || intensitas==97 ||
intensitas==96){

    dateText.setText(R.string.date_today2);
    timeText.setText(R.string.time_now2);
    txtJudul.setText(Constants.KEY_PESAN_PERIKSA);
    txtPesan.setText(Constants.KEY_PESAN_PERIKSA2);
    repeatType = Reminder.WEEKLY;
    calendar.set(Calendar.DAY_OF_MONTH,
calendar.get(Calendar.DAY_OF_MONTH) + 14);
    temp_intensitas=intensitas-10;//93,91,89,87
    txtUlang.setText("5");
    timesToShow=5;
    onAdvancedRepeatSelection(3, 2, "Repeats every:");
}if(intensitas==95){
    dateText.setText(R.string.date_today2);
    timeText.setText(R.string.time_now2);
    txtJudul.setText(Constants.KEY_PESAN_PERIKSA);
    txtPesan.setText(Constants.KEY_PESAN_PERIKSA2);
}

```

```

        repeatType = Reminder.WEEKLY;
        calendar.set(Calendar.DAY_OF_MONTH,
calendar.get(Calendar.DAY_OF_MONTH) + 14);
        temp_intensitas=intensitas-10; //93,91,89,87
        txtUlang.setText("9");
        timesToShow=9;
        onAdvancedRepeatSelection(3, 2, "Repeats every:");
    }
    spinJenis.setOnItemSelectedListener(spinnerListener);
    spinTahap.setOnItemSelectedListener(spinnerListener);
    spinBerat.setOnItemSelectedListener(spinnerListener);
    spinPola.setOnItemSelectedListener(spinnerListener);
    spinJenis.setOnItemSelectedListener(spinnerListener);
    spinTahap.setOnItemSelectedListener(spinnerListener);
    spinBerat.setOnItemSelectedListener(spinnerListener);
    spinPola.setOnItemSelectedListener(spinnerListener);
}

private AdapterView.OnItemClickListener spinnerListener = new
AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parentView, View
selectedItemView, int position, long id) {

        if (spinTahap.getSelectedItemPosition() == 1 &&
spinJenis.getSelectedItemPosition() == 1){
            repeat_row.setVisibility(View.GONE);
            repeatType = Reminder.DAILY;
            timesToShow=56;
            txtUlang.setText("56");
            txtPesan.setText("");
            if (spinBerat.getSelectedItemPosition() == 1){
                txtPesan.setText("");
            }
            txtPesan.setText(pesan+""+Constants.jenisObatKDT1Intensif1);
            temp_intensitas=intensitas-1; //99
        }else if (spinBerat.getSelectedItemPosition() == 2){
            txtPesan.setText("");
        }
        txtPesan.setText(pesan+""+Constants.jenisObatKDT1Intensif2);
            temp_intensitas=intensitas-2; //98
        }else if (spinBerat.getSelectedItemPosition() == 3){
            txtPesan.setText("");
        }
        txtPesan.setText(pesan+""+Constants.jenisObatKDT1Intensif3);
            temp_intensitas=intensitas-3; //97
        }else if (spinBerat.getSelectedItemPosition() == 4){
            txtPesan.setText("");
        }
        txtPesan.setText(pesan+""+Constants.jenisObatKDT1Intensif4);
            temp_intensitas=intensitas-4; //96
        }
    }
}

@Override
public void onNothingSelected(AdapterView<?> parentView) {
}

};

public void saveNotification() {
    DatabaseHelper database = DatabaseHelper.getInstance(this);
    Reminder reminder = new Reminder()
        .setId(id)
        .setTitle(txtJudul.getText().toString())
}

```

```

        .setContent(txtPesan.getText().toString())

        .setDateAndTime(DateAndTimeUtil.toStringDateAndTime(calendar))
        .setRepeatType(repeatType)

        .setForeverState(Boolean.toString(foreverSwitch.isChecked()))
        .setNumberToShow(timesToShow)
        .setNumberShown(timesShown)
        .setIcon(icon)
        .setColour(colour)
        .setInterval(interval);
        database.addNotification(reminder);
        if (repeatType == Reminder.SPECIFIC_DAYS) {
            reminder.setDaysOfWeek(daysOfWeek);
            database.addDaysOfWeek(reminder);
        }
        database.close();
        Intent alarmIntent = new Intent(this, AlarmReceiver.class);
        calendar.set(Calendar.SECOND, 0);
        AlarmUtil.setAlarm(this, alarmIntent, reminder.getId(),
calendar);
        if (temp_intensitas==99 || temp_intensitas==98 ||
temp_intensitas==97 || temp_intensitas==96 || temp_intensitas==95){
            intent = new Intent(getApplicationContext(),
CreateSchedule.class);
            b = new Bundle();
            b.putInt(Constants.KEY_INTENSITAS, temp_intensitas);
            //b.putString(Constants.KEY_PESAN,
Constants.KEY_PESAN_PERIKSA);
            intent.putExtras(b);
            startActivity(intent);
        }else{
            intent = new Intent(getApplicationContext(),
MainActivity.class);
            startActivity(intent);
        }
        finish();
    }
}

```

2. Source code layout file .xml

a. activity_create_schedule.xml

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    tools:ignore="ContentDescription"
    android:id="@+id/create_coordinator"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/background">

    <LinearLayout
        xmlns:app="http://schemas.android.com/apk/res-auto"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <LinearLayout
            android:id="@+id/header"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="?attr/colorPrimary"
            android:orientation="vertical">

```

```

<android.support.v7.widget.Toolbar
    android:id="@+id/toolbar"
    android:layout_width="match_parent"
    android:layout_height="?attr/actionBarSize"
    android:theme="@style/Toolbar"
    android:background="?attr/colorPrimary"
    app:popupTheme="@style/AppTheme.PopupOverlay" />

<LinearLayout
    android:visibility="gone"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="?attr/colorPrimary"
    android:orientation="vertical"
    android:paddingBottom="14dp"
    android:paddingLeft="@dimen/title_padding_left"
    android:paddingRight="@dimen/title_padding_right">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/title"

        android:textAppearance="?android:attr/textAppearanceSmall"
        android:textColor="@android:color/white" />

    <EditText
        android:id="@+id/txt_judul"
        android:text="Waktunya minum obat..."
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textCapSentences|textAutoCorrect"
        android:textColor="@android:color/white"
        android:textColorHighlight="@color/colorAccent"
        android:textSize="28sp"
        android:theme="@style/TitleEditText" />

</LinearLayout>
</LinearLayout>

<View
    android:id="@+id/toolbar_shadow"
    android:layout_width="match_parent"
    android:layout_height="@dimen/toolbar_shadow_height"
    android:layout_marginBottom="@dimen/toolbar_shadow_adjust"
    android:background="@drawable/toolbar_shadow" />

<ScrollView
    android:layout_width="match_parent"
    android:paddingLeft="@dimen/create_scroll_view_padding"
    android:paddingStart="@dimen/create_scroll_view_padding"
    android:layout_height="match_parent">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:animateLayoutChanges="true">

        <LinearLayout
            android:visibility="gone"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"

            android:paddingBottom="@dimen/table_row_default_horizontal_padding"

```

```

        <LinearLayout
            android:id="@+id/spin2_row"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="?attr/selectableItemBackground"

            android:paddingBottom="@dimen/table_row_default_horizontal_padding"

            android:paddingEnd="@dimen/table_row_default_padding_side"

            android:paddingRight="@dimen/table_row_default_padding_side"

            android:paddingTop="@dimen/table_row_default_horizontal_padding"
            android:weightSum="1">

            <Spinner
                android:id="@+id/spin_jenis"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:entries="@array/isi_paket oat"
                android:spinnerMode="dialog" />

        </LinearLayout>

        <LinearLayout
            android:id="@+id/spin3_row"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="?attr/selectableItemBackground"

            android:paddingBottom="@dimen/table_row_default_horizontal_padding"

            android:paddingEnd="@dimen/table_row_default padding_side"

            android:paddingRight="@dimen/table_row_default_padding_side"

            android:paddingTop="@dimen/table_row_default_horizontal_padding"
            android:weightSum="1">

            <Spinner
                android:id="@+id/spin_berat"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:entries="@array/isi_berat"
                android:spinnerMode="dialog" />

        </LinearLayout>

        <View
            android:layout_width="match_parent"
            android:layout_height="@dimen/line_separator_height"

            android:layout_marginLeft="@dimen/line_separator_margin_side_large"

            android:layout_marginStart="@dimen/line_separator_margin_side_large"
            android:background="@color/lightGray" />

        <LinearLayout
            android:id="@+id/time_row"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="?attr/selectableItemBackground"

```

```

android:paddingBottom="@dimen/table_row_default_horizontal_padding"
android:paddingEnd="@dimen/table_row_default_padding_side"
android:paddingRight="@dimen/table_row_default_padding_side"
android:paddingTop="@dimen/table_row_default_horizontal_padding"
    android:weightSum="1">
    <ImageView
        android:id="@+id/time_image"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
android:layout_marginLeft="@dimen/image_margin_side"
android:layout_marginStart="@dimen/image_margin_side"
        android:tint="@color/iconTint"
android:src="@drawable/ic_access_time_white_24dp"/>
    <TextView
        android:id="@+id/txt_jam"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
android:layout_marginLeft="@dimen/text_margin_side"
android:layout_marginStart="@dimen/text_margin_side"
        android:text="@string/time_now"
        android:textColor="@android:color/black"
        android:textSize="16sp" />
    </LinearLayout>
    <View
        android:layout_width="match_parent"
        android:layout_height="@dimen/line_separator_height"
android:layout_marginLeft="@dimen/line_separator_margin_side_large"
android:layout_marginStart="@dimen/line_separator_margin_side_large"
        android:background="@color/lightGray" />
    <LinearLayout
        android:id="@+id/date_row"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="?attr/selectableItemBackground"
android:paddingBottom="@dimen/table_row_default_horizontal_padding"
android:paddingEnd="@dimen/table_row_default_padding_side"
android:paddingRight="@dimen/table_row_default_padding_side"
android:paddingTop="@dimen/table_row_default_horizontal_padding"
        android:weightSum="1">
        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
android:layout_marginLeft="@dimen/image_margin_side"

```

```

android:layout_marginStart="@dimen/image_marqin_side"
        android:tint="@color/icorTint"
        android:src="@drawable/ic_today_white_24dp" />

        <TextView
            android:id="@+id/date"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"

android:layout_marginLeft="@dimen/text_margin_side"

android:layout_marginStart="@dimen/text_margin_side"
        android:text="@string/date_today"
        android:textColor="@android:color/black"
        android:textSize="16sp" />
    </LinearLayout>
</ScrollView>
</LinearLayout>
</android.support.design.widget.CoordinatorLayout>

```

b. activity_maps.xml

```

<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".fragments.Maps"
    android:background="@color/background">

    <fragment
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:id="@+id/map"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        class="com.google.android.gms.maps.SupportMapFragment" />

</RelativeLayout>

```

3. Manifest

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="dev.elte.dokterretus">

    <uses-permission android:name="android.permission.VIBRATE" />
    <uses-permission
android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
    <uses-permission android:name="android.permission.WAKE_LOCK"/>

    <uses-permission android:name="android.permission.GET_ACCOUNTS" />
    <uses-permission android:name="android.permission.READ_PROFILE" />
    <uses-permission android:name="android.permission.READ_CONTACTS" />

    <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission
android:name="com.google.android.providers.gsf.permission.READ_GSERVICES"
/>
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE"
/>

```

```

<permission
  android:name="dev.elte.dokterretus.permission.MAPS_RECEIVE"
  android:protectionLevel="signature" />

<permission
  android:name="dev.elte.dokterretus.permission.ACCESS_COARSE_LOCATION"
  android:protectionLevel="signature" />
<permission
  android:name="dev.elte.dokterretus.permission.ACCESS_FINE_LOCATION"
  android:protectionLevel="signature" />

<uses-feature
  android:glEsVersion="0x00020000"
  android:required="true"/>

<application
  android:allowBackup="true"
  android:name=".application.RemindMe"
  android:icon="@drawable/logo"
  android:label="@string/app_name"
  android:supportsRtl="true"
  android:theme="@style/AppTheme">
  <activity
    android:name=".activities.Login"
    android:screenOrientation="portrait">
    <intent-filter>
      <action android:name="android.intent.action.MAIN" />

      <category android:name="android.intent.category.LAUNCHER"
/>
      </intent-filter>
    </activity>

    <activity
      android:name=".activities.Register"
      android:screenOrientation="portrait"
      android:theme="@style/AppTheme"/>
    <activity
      android:name=".activities.MainActivity"
      android:screenOrientation="portrait"
      android:theme="@style/AppTheme.NoActionBar"/>

    <activity
      android:name=".activities.MainActivity2"
      android:screenOrientation="portrait"
      android:theme="@style/AppTheme.NoActionBar"/>

    <activity
      android:name=".fragments.Maps"
      android:screenOrientation="portrait"/>

    <service android:name=".services.MyFirebaseInstanceIdService">
      <intent-filter>
        <action
  android:name="com.google.firebase.INSTANCE_ID_EVENT" />
        </intent-filter>
      </service>
      <service android:name=".services.MyFirebaseMessagingService">
        <intent-filter>
          <action
  android:name="com.google.firebase.MESSAGING_EVENT" />
          </intent-filter>
        </service>

```

```

<meta-data
    android:name="com.google.android.maps.v2.API_KEY"
    android:value="AIzaSyBB9C3YeCbUJHRgiLhJsn4KA8nSbUn0" />
<meta-data
    android:name="com.google.android.gms.version"
    android:value="@integer/google_play_services_version" />

<activity
    android:name=".activities.Scheduling"
    android:label="@string/app_name"
    android:screenOrientation="portrait"
    android:theme="@style/AppTheme.NoActionBar"/>

<activity
    android:name=".activities.CreateSchedule"
    android:theme="@style/AppThemeLight"
    android:screenOrientation="portrait"
    android:configChanges="orientation|screenSize"/>

<activity
    android:name=".activities.ViewActivity"
    android:screenOrientation="portrait"
    android:theme="@style/AppThemeLight"/>

<activity
    android:name=".activities.PreferenceActivity"
    android:label="@string/settings"
    android:theme="@style/AppTheme"/>

<activity
    android:name=".activities.SnoozeDialogActivity"
    android:taskAffinity=""
    android:theme="@style/AppThemeTransparent"
    android:excludeFromRecents="true"/>

<receiver android:name=".receivers.AlarmReceiver" />

<receiver android:name=".receivers.SnoozeActionReceiver" />

<receiver android:name=".receivers.SnoozeReceiver" />

<receiver android:name=".receivers.DismissReceiver" />

<receiver android:name=".receivers.NagReceiver" />

<receiver
    android:name=".receivers.BootReceiver"
    android:enabled="true" >
    <intent-filter>
        <action
            android:name="android.intent.action.BOOT_COMPLETED" />
        <action
            android:name="android.intent.action.MY_PACKAGE_REPLACED" />
        <category android:name="android.intent.category.DEFAULT"
/>
    </intent-filter>
</receiver>

<receiver
    android:name=".receivers.ConnectivityReceiver"
    android:enabled="true">
    <intent-filter>
        <action
            android:name="android.net.conn.CONNECTIVITY_CHANGE" />
    </intent-filter>

```

```
        </receiver>
    </application>
</manifest>
```

4. Gradle (module)

```
apply plugin: 'com.android.application'
apply plugin: 'com.neenbedankt.android-apt'

android {
    compileSdkVersion 24
    buildToolsVersion "24.0.0"

    defaultConfig {
        applicationId "dev.site.doxterretus"
        minSdkVersion 14
        targetSdkVersion 24
        versionCode 1
        versionName "1.0"
    }
    buildTypes {
        release {
            minifyEnabled true
            shrinkResources true
            proguardFiles getDefaultProguardFile('proguard-android.txt'),
'proguard-rules.pro'
        }
        debug {
            minifyEnabled true
            shrinkResources true
        }
    }
}

dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    testCompile 'junit:junit:4.12'
    compile 'com.android.support:appcompat-v7:24.2.1'
    compile 'com.android.support:design:24.2.1'
    compile 'com.android.support:support-v4:24.2.1'
    compile 'com.android.support:cardview-v7:24.2.1'

    compile 'com.github.markushi:circlebutton:1.1'
    compile 'com.jakewharton:butterknife:8.1.0'
    apt 'com.jakewharton:butterknife-compiler:8.1.0'
    compile 'com.afollestad.material-dialogs:commons:0.8.6.1'
    compile 'com.jpardogo.materialtabstrip:library:1.1.0'

    compile 'com.mcxiaoke.volley:library:1.0.19'

    compile 'com.google.firebase:firebase-messaging:9.4.0'
    compile 'com.google.android.gms:play-services:9.4.0'
    compile 'com.google.android.gms:play-services-maps:9.4.0'
    compile 'com.google.maps.android:android-maps-utils:0.4+'
}
apply plugin: 'com.google.gms.google-services'
```

5. Gradle (project)

```
buildscript {
    repositories {
        jcenter()
    }
    dependencies {
        classpath 'com.android.tools.build:gradle:2.1.2'
        classpath 'com.google.gms:google-services:3.0.0'
        classpath 'com.neenbedankt.gradle.plugins:android-apt:1.8'
```

```

    }
}

allprojects {
    repositories {
        jcenter()
    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}

```

6. Source code .php

a. ambil token.php

```

<?php
if($_SERVER['REQUEST_METHOD']=='GET'){

    include("koneksi.php");

    $sql = "SELECT * from devices";

    $res = mysqli_query($con,$sql);

    $result = array();

    while($row = mysqli_fetch_array($res)){
        array_push($result,
            array('id'=>$row['id'],
                'email'=>$row['email'],
                'token'=>$row['token']
            ));
    }

    echo json_encode(array("result"=>$result));
    mysqli_close($con);
}

```

b. Config.php

```

<?php
define('DB_USERNAME','root');
define('DB_PASSWORD','');
define('DB_NAME','dr_retus');
define('DB_HOST','localhost');

define('FIREBASE_API_KEY',
'AAAAKuNkI3c:APA91bGhLÜrcAM192iX2NiECVHNjLj--
Qg3Qgj7hYXDo6pWSvWm6CztKBFONDh2m-
SrAplxguaj_XSPuOeZafNZDmW46ztOqQn7e99yJWFX_yuRAcr02e26JEdb8coV8Md6QFHMS8-
zsZ8qaMqLDlvsVtEzY939dqw');

```

c. DbConnect.php

```

<?php
//Class DbConnect
class DbConnect
{
    private $con;

    function __construct()
    {

    }

    function connect()

```

```

    }
    include_once dirname(__FILE__) . '/Config.php';

    //connecting to mysql database
    $this->con = new mysqli(DB_HOST, DB_USERNAME, DB_PASSWORD,
DB_NAME);
    if (mysqli_connect_errno()) {
        echo "Failed to connect to MySQL: " . mysqli_connect_error();
    }
    return $this->con;
}
}

```

d. DbOperation.php

```

<?php
class DbOperation
{
    //Database connection link
    private $con;

    //Class constructor
    function __construct()
    {
        //Getting the DbConnect.php file
        require_once dirname(__FILE__) . '/DbConnect.php';

        //Creating a DbConnect object to connect to the database
        $db = new DbConnect();

        //Initializing our connection link of this class
        //by calling the method connect of DbConnect class
        $this->con = $db->connect();
    }

    //storing token in database
    public function registerDevice($email,$password,$token,$role){
        if(!$this->isEmailExist($email)){
            $stmt = $this->con->prepare("INSERT INTO devices (email,
password, token, role) VALUES (?, ?, ?, ?) ");
            $stmt->bind_param("ssss", $email, $password, $token, $role);
            if($stmt->execute())
                return 0; //return 0 means success
            return 1; //return 1 means failure
        }else{
            return 2; //returning 2 means email already exist
        }
    }

    //storing data pasien
    public function registerPasien($id_pasien,$email,$nama){
        if($this->isEmailExist($email)){
            $stmt = $this->con->prepare("INSERT INTO pasien (id_pasien,
email, nama) VALUES (?, ?, ?) ");
            $stmt->bind_param("sss", $id_pasien, $email, $nama);
            if($stmt->execute())
                return 0; //return 0 means success
            return 1; //return 1 means failure
        }else{
            return 2; //returning 2 means email already exist
        }
    }

    //storing data pasien
    public function registerKerabat($id_kerabat,$email,$id_pasien,$nama){
        if($this->isEmailExist($email)){

```

```

        $stmt = $this->con->prepare("INSERT INTO kerabat (id_kerabat,
email, id_pasien, nama) VALUES (?, ?, ?, ?) ");
        $stmt->
>bind_param("ssss", $id_kerabat, $email, $id_pasien, $nama);
        if($stmt->execute())
            return 0; //return 0 means success
            return 1; //return 1 means failure
        }else{
            return 2; //returning 2 means email already exist
        }
    }

    //the method will check if email already exist
    private function isEmailExist($email){
        $stmt = $this->con->prepare("SELECT id FROM devices WHERE email =
?");
        $stmt->bind_param("s", $email);
        $stmt->execute();
        $stmt->store_result();
        $num_rows = $stmt->num_rows;
        $stmt->close();
        return $num_rows > 0;
    }

    //getting all tokens to send push to all devices
    public function getAllTokens(){
        $stmt = $this->con->prepare("SELECT token FROM devices where
role='kerabat' or role='dokter'");
        $stmt->execute();
        $result = $stmt->get_result();
        $tokens = array();
        while($token = $result->fetch_assoc()){
            array_push($tokens, $token['token']);
        }
        return $tokens;
    }

    //getting a specified token to send push to selected device
    public function getTokenByEmail($email){
        $stmt = $this->con->prepare("SELECT token FROM devices WHERE
email = ?");
        $stmt->bind_param("s", $email);
        $stmt->execute();
        $result = $stmt->get_result()->fetch_assoc();
        return array($result['token']);
    }

    //getting all the registered devices from database
    public function getAllDevices(){
        $stmt = $this->con->prepare("SELECT * FROM devices");
        $stmt->execute();
        $result = $stmt->get_result();
        return $result;
    }
}

```

e. GetRegisteredDevices.php

```

<?php
require_once 'DbOperation.php';
$db = new DbOperation();
$devices = $db->getAllDevices();
$response['devices'] = array();

while($device = $devices->fetch_assoc()){
    $temp = array();
    $temp['id'] = $device['id'];
    $temp['email'] = $device['email'];
    $temp['token'] = $device['token'];
}

```