

E-TOCOLOGIST
**(SISTEM INFORMASI KEBIDANAN UNTUK PELAYANAN
MONITORING KESEHATAN PADA IBU DAN ANAK BERBASIS
WEB DAN APLIKASI *MOBILE* DI KLINIK ROHAENI, S.ST)**

E-TOCOLOGIST
(*MIDWIFERY INFORMATION SYSTEM FOR HEALTH MONITORING SERVICES IN
MOTHERS AND CHILDREN BASED ON WEB AND MOBILE APPLICATIONS IN
CLINICS ROHAENI, S.ST*)

PROPOSAL PROYEK AKHIR

Diajukan sebagai syarat untuk mengambil Mata Kuliah Proyek akhir

oleh :

MUHAMMAD FAJAR NUGROHO ALAM
6705180059



D3 TEKNOLOGI TELEKOMUNIKASI
FAKULTAS ILMU TERAPAN
UNIVERSITAS TELKOM
2021

LEMBAR PENGESAHAN

Proposal Proyek Akhir dengan judul :

E-TOCOLOGIST

(SISTEM INFORMASI KEBIDANAN UNTUK PELAYANAN MONITORING
KESEHATAN PADA IBU DAN ANAK BERBASIS WEB DAN APLIKASI *MOBILE* DI
KLINIK ROHAENI, S.ST)

E-TOCOLOGIST

(*MIDWIFERY INFORMATION SYSTEM FOR HEALTH MONITORING SERVICES IN
MOTHERS AND CHILDREN BASED ON WEB AND MOBILE APPLICATIONS IN
CLINICS ROHAENI, S.ST*)

oleh :

MUHAMMAD FAJAR NUGROHO ALAM

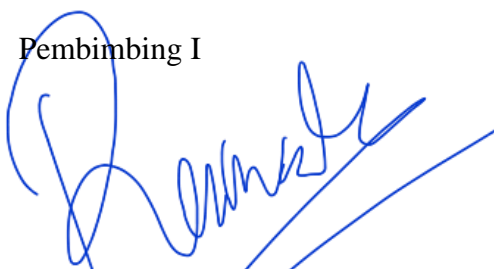
6705180059

Telah diperiksa dan disetujui untuk diajukan sebagai syarat mengambil
Mata Kuliah Proyek Akhir
pada Program Studi D3 Teknologi telekomunikasi Universitas Telkom

Bandung, 22 Januari 2021

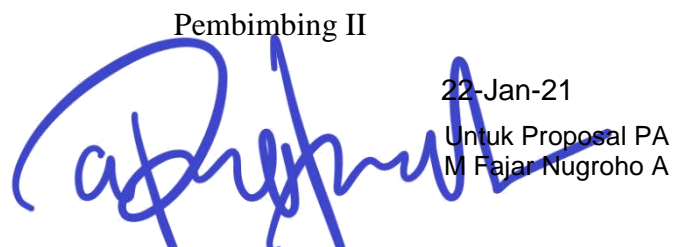
Menyetujui,

Pembimbing I



Dadan Nur Ramadhan, S.Pd., M.T.
NIP. 14820047

Pembimbing II



22-Jan-21
Untuk Proposal PA
M Fajar Nugroho A
Rohmat Tulloh, S.T., M.T.
NIP. 06830002

ABSTRAK

Perkembangan teknologi informasi dan komunikasi akhir-akhir ini telah berkembang pesat disegala bidang salah satunya pada sistem informasi dalam bentuk website dan Mobile Application yang banyak digunakan dalam sektor Industri, Hiburan maupun Kesehatan. Namun dalam sektor kebidanan, pengguna teknologi informasi ini masih jarang untuk diterapkan. Pada studi kasus di Klinik Rohaeni Budiman, S.ST, rancangan sistem masih tergolong konvensional, dikarenakan sistem administrasi dan pelayanan klinik yang terorganisir secara manual. Hal tersebut kurang efektif dan efisien yang menyebabkan beban kerja bertambah, proses mencari informasi yang cukup lama dan resiko hilangnya data-data penting akan sering terjadi. Sementara, pasien harus mendatangi bidan secara langsung ke tempat klinik apabila ingin mengetahui informasi perkembangan kehamilan dan anak balitanya.

Maka dirancanglah aplikasi E-TOCOLOGIST yang dapat menjadi solusi permasalahan tersebut. E-TOCOLOGIST terbagi menjadi 2 bagian, yaitu *website* dan aplikasi *mobile*. Pada bagian *website*, mempunyai hak akses utama yaitu *Owner* dan *Admin*. Seluruh layanan yang disediakan oleh klinik dapat diakses oleh *Owner* dan *Admin* hanya dapat mengelola bagian administrasi, penyimpanan obat dan data akun dari pengguna. Pada bagian aplikasi *mobile* mempunyai hak akses untuk bidan dan pasien. Bidan mendapatkan akses untuk memonitoring kesehatan pasien, input data pasien, memantau jadwal temu pasien dan membalas konsultasi chat dari pasien. Sementara pasien dapat melakukan monitoring kesehatan, layanan klinik, chat bidan untuk konsultasi dan *Find Us* digunakan sebagai sarana pencarian lokasi klinik dari tempat user berada.

Diharapkan dengan adanya aplikasi E-TOCOLOGIST dapat menjadi solusi dari permasalahan yang ada di klinik Bidan Rohaeni, S.ST, menjadikan bidan dan admin yang bekerja di klinik semakin efektif dan efisien serta membuat pasien lebih mudah dalam menggunakan layanan yang disediakan oleh klinik.

Kata kunci : E-Tocologist, Sistem Informasi, Kebidanan, Monitoring Kesehatan

DAFTAR ISI

LEMBAR PENGESAHAN.....	1
ABSTRAK.....	2
DAFTAR GAMBAR	4
DAFTAR TABEL.....	5
BAB I PENDAHULUAN	6
1.1 Latar Belakang	6
1.2 Tujuan dan Manfaat	6
1.3 Rumusan Masalah.....	7
1.4 Batasan Masalah	7
1.4 Metodologi.....	8
BAB II DASAR TEORI	11
2.1 Monitoring Kesehatan Ibu dan Anak.....	11
2.2 Framework.....	11
2.3 Laravel	11
2.4 PHP	11
2.5 React Native.....	12
2.6 Microservice Architecture.....	12
2.7 Node Js.....	13
2.8 MySQL Database.....	14
2.9 Android Studio.....	14
2.10 Redis.....	14
BAB III MODEL SISTEM.....	16
3.1 Blok Diagram Sistem.....	16
3.2 Flowchart Sistem Aplikasi.....	17
3.3 Use Case Diagram.....	20
3.4 Flowchart Pengerjaan Aplikasi	21
BAB IV BENTUK KELUARAN YANG DIHARAPKAN	23
4.1 Keluaran yang Diharapkan	23
4.2 Jadwal Pelaksanaan.....	23

DAFTAR GAMBAR

Gambar 2 1 Arsitektur Microservices.....	13
Gambar 2 2 Cara Kerja Redis	15
Gambar 3 1 Arsitektur Rancangan Aplikasi E-TOCOLOGIST	16
Gambar 3 2 Flowchart perancangan E-TOCOLOGIST berbasis website	17
Gambar 3 3 Flowchart perancangan E-TOCOLOGIST berbasis mobile aplikasi.....	19
Gambar 3 4 Use Case Diagram	20
Gambar 3 5 Flowchart perancangan aplikasi E-TOCOLOGIST	22

DAFTAR TABEL

Tabel 4.1 Jadwal Pelaksanaan	23
------------------------------------	----

BAB I

PENDAHULUAN

1.1 Latar Belakang

Teknologi adalah sebuah alat yang digunakan untuk mempermudah dalam kebutuhan manusia. Perkembangan teknologi informasi dan komunikasi sendiri telah berkembang pesat di segala bidang salah satunya adalah dalam bidang Sistem Informasi dalam bentuk *Website* dan *Mobile Application* yang banyak digunakan dalam sektor Industri, Hiburan maupun Kesehatan. Namun dalam dunia Kebidanan, penggunaan teknologi ini masih sedikit penggunaannya.

Sementara pada Klinik Rohaeni Budiman, S.ST, rancangan sistem masih tergolong konvensional, dikarenakan dikelola dengan pencatatan manual. Hal tersebut kurang efektif dan efisien yang menyebabkan beban kerja bertambah, proses mencari informasi yang cukup lama dan resiko hilangnya data-data penting akan sering terjadi. Disisi lain, pasien ibu harus mendatangi bidan secara langsung apabila ingin melakukan beberapa pelayanan lalu mengetahui informasi perkembangan kehamilan dan anak balitanya. Oleh karena itu maka dibangun sistem informasi *E-TOCOLOGIST* yang dapat di akses oleh bidan dan orang tua untuk memudahkan beberapa sarana fasilitas yang disediakan oleh pihak klinik

E-TOCOLOGIST terintegrasi dengan 2 platform. Yaitu web untuk informasi pelayanan dari klinik dan pengolahan data pemasukan administrasi, lalu aplikasi mobile untuk monitoring kesehatan ibu dan anak.

1.2 Tujuan dan Manfaat

Adapun tujuan dari Proyek tingkat ini, sebagai berikut:

1. Dapat membuat sistem informasi berbasis website dan mobile aplikasi untuk pelayanan kebidanan di klinik Rohaeni, S.ST
2. Dapat membuat aplikasi mobile yang berjalan di 2 platform (Android dan IOS).
3. Membuat arsitektur sistem Microservice pada bagian website untuk fitur pelayanan kebidanan di klinik Rohaeni, S.ST

4. Membuat aplikasi mobile sebagai sarana monitoring kesehatan pasien ibu dan anak, *chatting* user ke bidan, layanan berdasarkan poli klinik dan pemesanan janji temu dari pasien ibu terhadap bidan.
5. Membuat sistem penyimpanan data utama dan cadangan pada sisi web dan aplikasi mobile.

Adapun manfaat dari Proyek tingkat ini, sebagai berikut:

1. Membantu bidan untuk lebih mudah mengelola serta memonitoring data dari pasien ibu dan anak.
2. Membantu admin dalam perihal administrasi klinik menjadi lebih efisien agar mengurangi resiko hilangnya data.
3. Membantu user agar dapat menggunakan layanan dari klinik semakin mudah dan efisien.
4. Membantu admin untuk pengecekan obat dari daftar penyimpanan barang.
5. Mengedukasi user agar lebih tahu tentang kesehatan ibu dan anak.

1.3 Rumusan Masalah

Adapun rumusan masalah dari Proyek tingkat ini, sebagai berikut:

1. Bagaimana membuat sistem informasi berbasis website dan mobile aplikasi untuk pelayanan kebidanan?
2. Bagaimana membuat mobile aplikasi yang berjalan di 2 platform?
3. Bagaimana membangun rancangan microservice untuk fitur pelayanan kebidanan?
4. Bagaimana dampak terhadap aplikasi jika *request* user terlalu banyak?
5. Bagaimana cara menghubungkan data user dan bidan secara bersamaan?
6. Bagaimana dampak terhadap aplikasi jika ada salah satu layanan terjadi gangguan?

1.4 Batasan Masalah

Dalam Proyek tingkat ini, dilakukan pembatasan masalah sebagai berikut:

1. Perancangan dan pembuatan aplikasi *E-Tocologist* hanya untuk digunakan di klinik Rohaeni, S.ST
2. *Cross platform* untuk aplikasi mobile dengan *framework* React Native.

3. Menggunakan MySQL sebagai penyimpanan data.
4. Menggunakan RedisDB sebagai penyimpanan data cadangan yang tersimpan di disk.
5. Arsitektur rancangan menggunakan microservice dan pembuatan API sebagai *endpoint* setiap layanan klinik.
6. Menggunakan bahasa pemrograman Javascript untuk aplikasi mobile dan bahasa pemrograman PHP untuk Web.

1.5 Metodologi

Metodologi pada penelitian ini, sebagai berikut:

1. Studi Literatur

Hal yang dilakukan adalah mencari informasi dan pendalaman materi-materi yang terkait melalui referensi yang tersedia di berbagai sumber.

2. Analisis Sistem

Pada tahap ini, sistem akan dianalisis bagaimana akan dijalankan nantinya. Hasil analisis berupa kelebihan dan kekurangan sistem, fungsi sistem, hingga pembaharuan yang dapat diterapkan.

Bagian ini termasuk dalam bagian perencanaan. Bagian lain yang termasuk dalam perencanaan ialah alokasi sumber daya, perencanaan kapasitas, penjadwalan proyek, estimasi biaya, dan penetapan.

Dengan demikian, hasil dari tahap perencanaan ialah rencana proyek, jadwal, estimasi biaya, dan ketentuan.

3. Perancangan Sistem

Setelah persyaratan dipahami, perancang dan pengembang dapat mulai mendesain *software*. Tahapan ini akan menghasilkan *prototype* dan beberapa *output* lain meliputi dokumen berisi desain, pola, dan komponen yang diperlukan untuk mewujudkan proyek tersebut.

Setelah spesifikasi, kemudian dilakukan perancangan sistem sebagai tahapan kelanjutannya. Tahap ini ialah tahap di mana seluruh hasil analisis dan pembahasan tentang spesifikasi sistem diterapkan menjadi rancangan atau cetak biru sebuah sistem.

Tahap ini disebut sebagai cetak biru, di mana sistem sudah siap untuk dikembangkan mulai dari implementasi, analisis sistem, hingga tenaga pendukung sistem yang akan dikembangkan.

4. Pembangunan Sistem

Pengembangan sistem ialah tahap di mana rancangan mulai dikerjakan, dibuat, atau diimplementasikan menjadi sistem yang utuh dan dapat digunakan. Jika diibaratkan bangunan, tahap ini merupakan tahap membangun.

Tahap ini memakan waktu cukup lama karena akan muncul kendala-kendala baru yang mungkin dapat menghambat jalannya pengembangan sistem. Pada tahapan ini, perancangan bisa saja berubah karena satu atau banyak hal.

Tahap selanjutnya ialah memproduksi perangkat lunak di bawah proses pengembangan. Menurut metodologi yang sudah digunakan, tahap ini dapat dilakukan dengan cepat. *Output* yang dihasilkan pada tahap ini ialah perangkat lunak yang telah berfungsi dan siap diuji.

5. Pengujian Sistem dan Troubleshooting

Sesudah sistem selesai dikembangkan, sistem harus melalui pengujian sebelum digunakan atau dikomersialisasikan. Tahap pengujian sistem harus dijalankan untuk mencoba apakah sistem yang dikembangkan dapat bekerja optimal atau tidak.

Pada tahap ini, ada beberapa hal yang harus diperhatikan, seperti kemudahan penggunaan sampai pencapaian tujuan dari sistem yang sudah disusun sejak perancangan sistem dilakukan. Jika ada kesalahan, tahap pertama hingga keempat harus diperbarui, diulangi, atau pun dirombak total.

Tahap tes aplikasi ialah bagian paling penting dalam rangkaian pembuatan sebuah perangkat lunak. Karena sangat tidak mungkin mempublikasikan sebuah *software* tanpa melalui pengujian terlebih dahulu.

Beberapa pengujian yang harus dilewati, antara lain kualitas kode, tes fungsional, tes integrasi, tes performa, dan tes keamanan.

Dari tahap ini, akan dihasilkan perangkat lunak yang telah dites dan siap untuk disebarkan ke dalam proses produksi.

6. Implementasi

Implementasi dan pemeliharaan merupakan tahap akhir dalam pembuatan Proyek akhir. Di tahap ini sistem sudah dibuat, diuji coba, dan dipastikan dapat bekerja optimal.

Setelah tahap pembuatan selesai, dilakukan implementasi dan pemeliharaan oleh pengguna. Pemeliharaan sangat penting untuk memastikan sistem bekerja dengan optimal setiap saat.

Untuk implementasi, langkah yang harus dilakukan adalah sebagai berikut.

- Menganalisis dan mempelajari sistem yang sudah ada dan sedang berjalan.
- Melakukan pemecahan masalah dalam pengembangan sistem.
- Menentukan penggunaan *hardware* dan *software* yang tepat.
- Merancang dan mengembangkan sistem baru.
- Memelihara dan meningkatkan sistem yang baru jika diperlukan.

Fase ini disebut juga sebagai tahap penyebaran. Pada tahap ini, *software* disebarakan setelah melewati proses yang melibatkan beberapa persetujuan manual. Tahap ini dilakukan sebelum menurunkan *software* ke produksi.

7. Survei dan Dokumentasi

Dilakukan survei untuk mencari tahu seberapa layak dan cocoknya sistem yang sudah dikembangkan, Agar pengembang tahu bagaimana langkah selanjutnya yang akan diambil untuk menunjang kebutuhan user.

Untuk dokumentasi langkah yang harus dilakukan ialah pembuatan laporan Proyek akhir dan pengambilan gambar yang cukup untuk dijadikan suatu dokumentasi.

BAB II

DASAR TEORI

2.1 Monitoring Kesehatan Ibu dan Anak

Monitoring dapat menjadi sebuah pengambilan keputusan terhadap pasien untuk menindaklanjuti kejadian sebelumnya pada pasien seperti pemantauan untuk perkembangan janin yang memudahkan untuk mendeteksi apabila ada kemungkinan perkembangan yang tidak normal [6].

2.2 Framework

Framework adalah sebuah arsitektur yang terbuka yang dibuat berdasarkan pada standar pengembangan perangkat lunak yang diterima secara umum. Dengan menggunakan *framework* bukan berarti kita akan terbebas dari pengkodean, karena kita harus menggunakan fungsi-fungsi yang tersedia di dalam sebuah *framework* tersebut [7].

2.3 Laravel

Laravel adalah kerangka aplikasi web dengan sintaks yang ekspresif dan elegan. Kerangka kerja web menyediakan struktur dan titik awal untuk membuat aplikasi, memungkinkan untuk fokus pada pembuatan sesuatu yang luar biasa. Laravel berusaha untuk memberikan pengalaman pengembang yang luar biasa, sambil menyediakan fitur-fitur canggih seperti injeksi ketergantungan menyeluruh, lapisan abstraksi basis data ekspresif, antrean dan tugas terjadwal, pengujian unit dan integrasi, dan banyak lagi [10].

Ada berbagai alat dan kerangka kerja yang tersedia untuk Anda saat membangun aplikasi web. Namun, Laravel adalah pilihan terbaik untuk membangun aplikasi web full-stack yang modern [10].

2.4 PHP

PHP adalah bahasa *server side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Karena PHP merupakan *server side scripting* maka sintak dan hasilnya akan dikirimkan ke *browser* dengan format HTML [8].

Server side scripting merupakan sebuah teknologi *scripting* atau pemrograman *website* dimana *script* (program) dikompilasi atau diterjemahkan ke *server*. Dengan *server side scripting*, memungkinkan untuk menghasilkan halaman *website* yang dinamis [12].

2.5 React Native

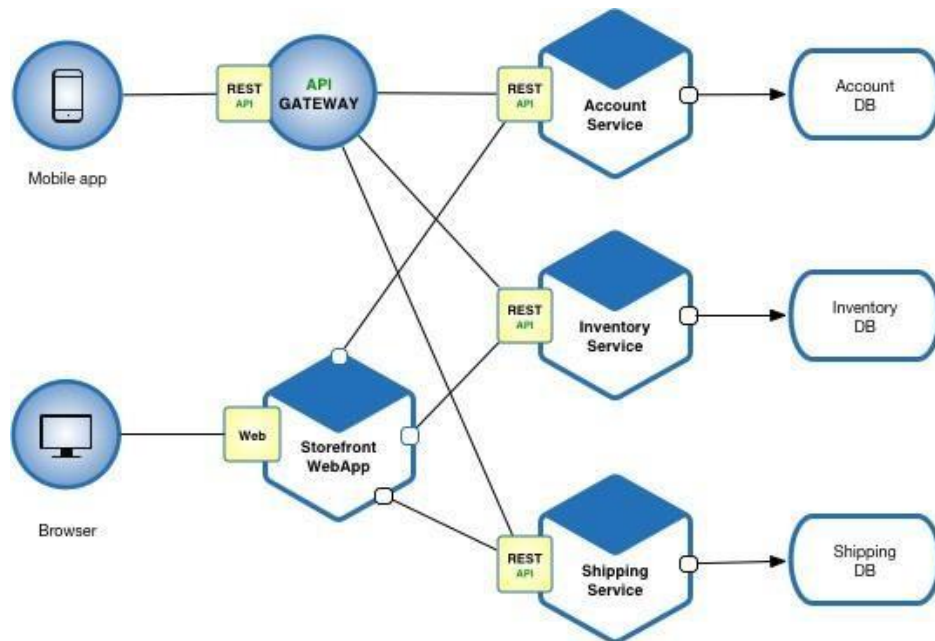
React Native adalah framework Javascript yang digunakan untuk membuat aplikasi native yang mampu berjalan di platform Android dan iOS. Framework ini berbasis React JS yang merupakan framework javascript buatan dari Facebook yang digunakan untuk membuat tampilan aplikasi yang berfokus pada aplikasi mobile. React Native dapat berjalan dengan baik tanpa terlihat perbedaan yang signifikan dibandingkan aplikasi asli, baik dari sisi tampilan maupun performa [4].

2.6 Microservice Architecture

Microservice adalah komponen kecil perangkat lunak yang khusus dalam satu tugas tertentu dan bekerja sama untuk mencapai tugas tingkat tinggi [5].

Aplikasi diatur sedemikian rupa sehingga saling terpisah menjadi *service-service* kecil yang independent, berfungsi spesifik (*high cohesion*) dan tidak saling bergantung pada komponen program lainnya (*loose coupling*), dengan antarmuka API (*Application Programming Interface*) [9].

Fungsi-fungsi yang terdapat pada arsitektur bisa dibuat dengan menggunakan teknologi stack yang sesuai kebutuhan tiap fungsi dan bisa berbeda satu sama lain. Setiap fungsi bisa dibangun oleh satu tim tersendiri dengan code base tersendiri dan bisa dilakukan testing secara independent [1]. Gambaran arsitektur *Microservices* bisa dilihat pada Gambar 2.1



Gambar 2 1 Arsitektur Microservices

2.7 Node Js

Sebagai *runtime* JavaScript berbasis peristiwa asinkron, Node.js dirancang untuk membangun aplikasi jaringan yang dapat diskalakan. Ini berbeda dengan model konkurensi yang lebih umum saat ini, di mana utas OS digunakan. Jaringan berbasis thread relatif tidak efisien dan sangat sulit digunakan. Selain itu, pengguna Node.js bebas dari kekhawatiran proses *dead-locking*, karena tidak ada kunci. Hampir tidak ada fungsi di Node.js yang secara langsung melakukan I / O, sehingga proses tersebut tidak pernah diblokir. Karena tidak ada yang menghalangi, sistem yang dapat diskalakan sangat masuk akal untuk dikembangkan di Node.js [2].

Node.js memiliki desain yang mirip dan dipengaruhi oleh sistem seperti Ruby Event Machine dan Python's Twisted. Node.js mengambil model acara sedikit lebih jauh. Ini menyajikan event loop sebagai konstruksi *runtime*, bukan sebagai perpustakaan. Di sistem lain, selalu ada panggilan pemblokiran untuk memulai event-loop. Di Node.js, tidak ada panggilan start-the-event-loop seperti itu. Node.js

cukup memasuki loop acara setelah menjalankan skrip input. Node.js keluar dari loop peristiwa ketika tidak ada lagi callback untuk dijalankan. Perilaku ini seperti JavaScript browser - event loop disembunyikan dari pengguna [2].

2.8 MySQL Database

Sistem database MySQL menggunakan arsitektur klien-server yang memiliki kendali pusat di server. Server tersebut merupakan sebuah program yang dapat memanipulasi *database*. Program klien tidak melakukannya secara langsung, tetapi ia mengkomunikasikan tujuan pengguna kepada server dengan cara menuliskan *query* dengan bahasa SQL (*Structured Query Language*). Program klien diinstall secara lokal di mesin di tempat dimana pengguna mengakses MySQL. Server dapat diinstall dimana saja, sepanjang klien dapat berhubungan dengannya. MySQL secara *inheren* merupakan sistem dengan database jaringan, sehingga setiap klien dapat berkomunikasi dengan server yang dijalankan secara lokal pada mesin pengguna [3].

2.9 Android Studio

Android Studio adalah lingkungan Pengembang Terpadu, IDE (*Integrated Development Environment*) untuk pengembangan aplikasi Android. Berdasarkan **IntelliJ IDEA**. Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur yang lebih banyak untuk meningkatkan produktivitas saat membuat aplikasi Android [13].

2.10 Redis

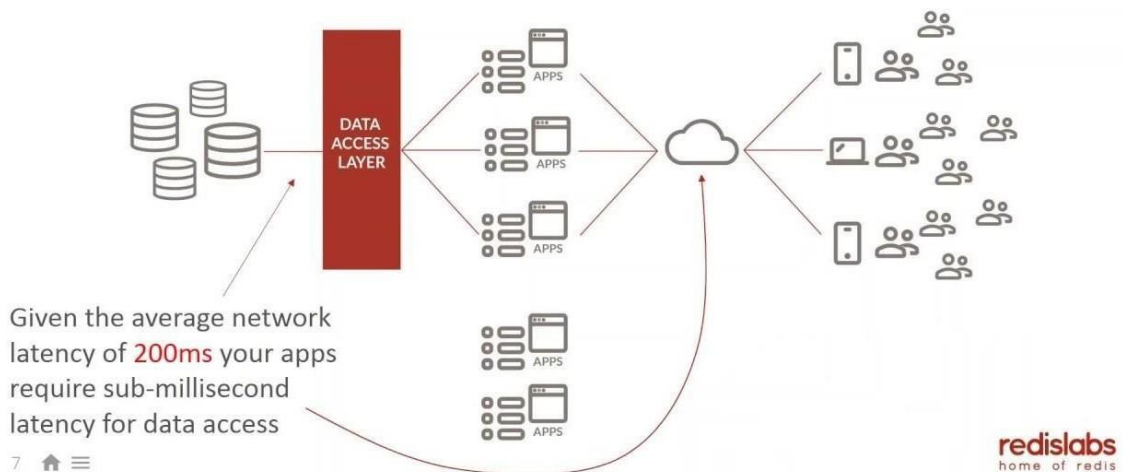
Redis, singkatan dari Remote Dictionary Server, adalah penyimpanan data nilai utama di dalam memori yang super cepat dengan sumber terbuka untuk digunakan sebagai database, cache, broker pesan, dan antrean. Proyek ini dimulai ketika Salvatore Sanfilippo, pengembang awal Redis, mencoba meningkatkan skalabilitas startup Italia miliknya. Redis kini memberikan respons dalam waktu di bawah satu milidetik yang memungkinkan jutaan permintaan per detik untuk aplikasi real-time pada Permainan, Ad-Tech, Layanan Finansial, Layanan Kesehatan, dan IoT. Redis adalah pilihan populer untuk caching, manajemen sesi, permainan, papan

peringkat, analisis real-time, geospasial, tumpangan berkendara, obrolan/perpesanan, streaming media, dan aplikasi pub/sub [11].

Semua data Redis terdapat di dalam memori, berbeda dengan database yang menyimpan data di disk atau SSD. Dengan menghilangkan kebutuhan untuk mengakses disk, penyimpanan data di dalam memori seperti Redis menghindari penundaan waktu pencarian dan dapat mengakses data dalam mikrodetik. Redis dilengkapi dengan struktur data serba guna, ketersediaan yang sangat baik, geospasial, Lua scripting, transaksi, persistensi di disk, dan dukungan klaster yang membuatnya lebih mudah untuk membuat aplikasi skala internet secara real-time [11]. Gambaran cara kerja redis dapat dilihat pada Gambar 2.2

Challenge #3

How to keep your applications responsive?



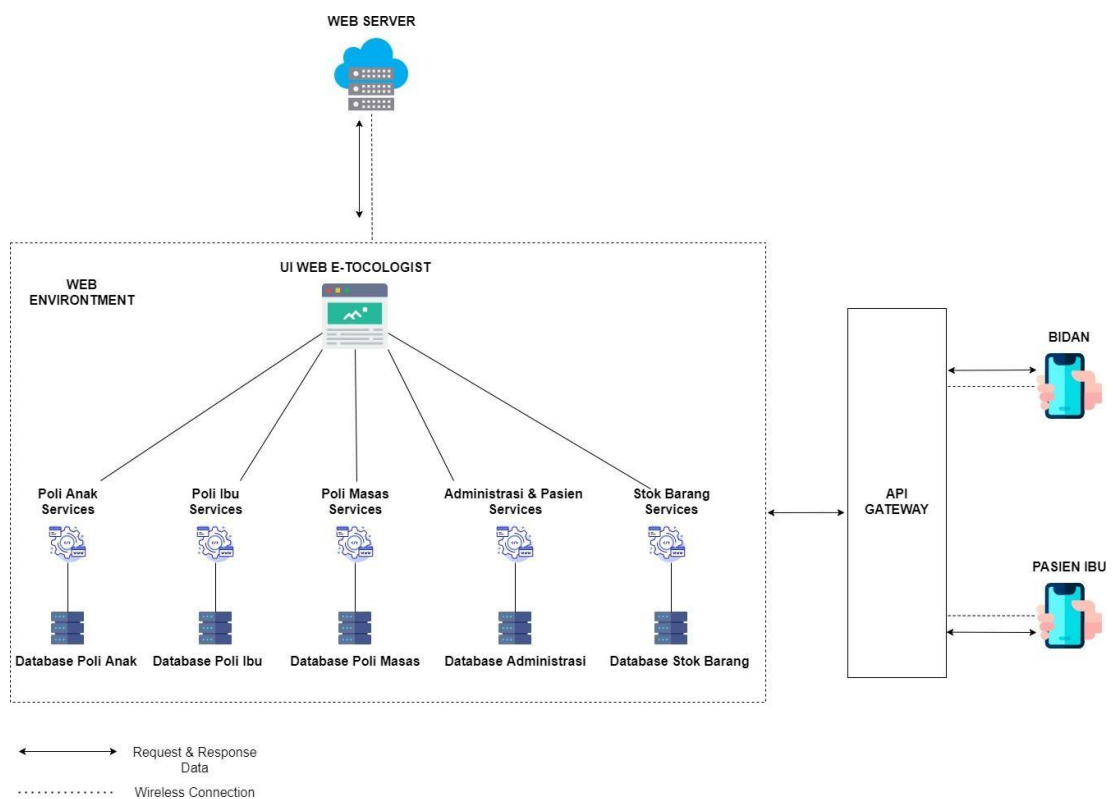
Gambar 2 2 Cara Kerja Redis

BAB III

MODEL SISTEM

3.1 Blok Diagram Sistem

Berikut adalah blok diagram keseluruhan dari aplikasi E-TOCOLOGIST berbasis *website* dan *mobile* aplikasi pada proyek akhir ini:



Gambar 3 1 Arsitektur Rancangan Aplikasi E-TOCOLOGIST

Arsitektur rancangan E-TOCOLOGIST pada gambar 3.1 terbagi menjadi 2 bagian, yaitu bagian web dan aplikasi. Pada platform web dirancang untuk digunakan oleh admin, dimana admin dapat mengelola informasi layanan dan mengelola data dari klinik tersebut. Sementara pada platform mobile yang terbagi menjadi 2 user, yaitu Bidan dan pasien ibu. Sistem monitoring kesehatan ibu dan anak dapat diakses oleh Bidan dan Pasien Ibu pada tampilan mobile aplikasi.

Bagian Mobile Aplikasi menerima sebuah response status sesuai dengan *endpoint* setiap layanan yang akan ditampilkan pada UI aplikasi untuk kebutuhan *Input* maupun *Output* pada user.

Berikut merupakan *flowchart* dari perancangan aplikasi E-TOCOLOGIST digital berbasis *website* pada proyek akhir kali ini:

Gambar 3 2 Flowchart perancangan E-TOCOLOGIST berbasis website

Dari *flowchart* diatas dapat diketahui bahwa website E-TOCOLOGIST mempunyai 2 akses, yaitu sebagai Owner dan Admin.

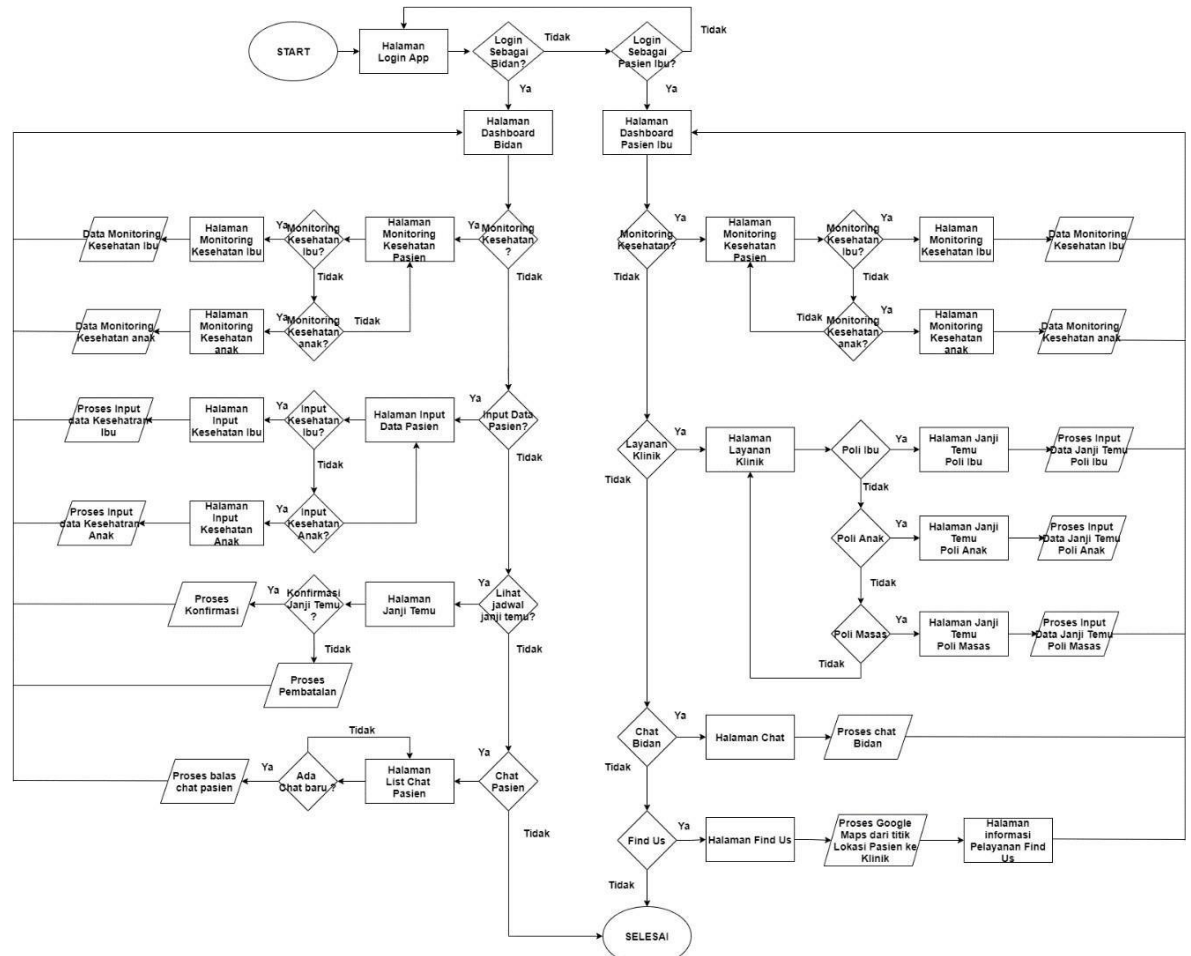
Pada akses Owner dapat melakukan semua fitur meliputi:

- Menu Registrasi Akun: Dimulai dari membuat, mengubah, melihat serta menghapus proses registrasi akun untuk bidan dan admin.
- Menu Service Layanan Klinik: Layanan klinik dibagi menjadi 3 Poli yaitu Poli Anak, Poli Ibu dan Poli Masas. Owner dapat membuat, mengubah, mengedit serta menghapus layanan berdasarkan setiap masing-masing Poli.
- Menu Administrasi Klinik: Dimulai dari membuat, mengubah, mengedit serta menghapus proses data administrasi klinik dan menghasilkan keluaran berupa print laporan.
- Menu Storage Obat: Dimulai dari membuat, mengubah, melihat serta menghapus proses data penyimpanan obat dan menghasilkan keluaran berupa print laporan.
- Menu Konfirmasi Akun: Owner dapat mengkonfirmasi akun ibu yang sudah terdaftar agar dapat digunakan apabila telah memenuhi seluruh persyaratan.

Pada akses admin hanya dapat melakukan beberapa fitur meliputi:

- Menu Administrasi Klinik: Dimulai dari membuat, mengubah, melihat serta menghapus proses data administrasi klinik dan menghasilkan keluaran berupa print laporan.
- Menu Storage Obat: Dimulai dari membuat, mengubah, mengedit serta menghapus proses data penyimpanan obat dan menghasilkan keluaran berupa print laporan.
- Menu Konfirmasi Akun: Admin dapat mengkonfirmasi akun ibu yang sudah terdaftar agar dapat digunakan apabila telah memenuhi seluruh persyaratan.

Berikut merupakan *flowchart* dari perancangan aplikasi E-TOCOLOGIST digital berbasis Mobile Aplikasi pada proyek akhir kali ini:



Gambar 3 3 Flowchart perancangan E-TOCOLOGIST berbasis mobile aplikasi

Dari *flowchart* diatas dapat diketahui bahwa Mobile Aplikasi E-TOCOLOGIST mempunyai 2 akses, yaitu Bidan dan Pasien Ibu.

Pada akses Bidan dapat melakukan beberapa fitur sebagai berikut:

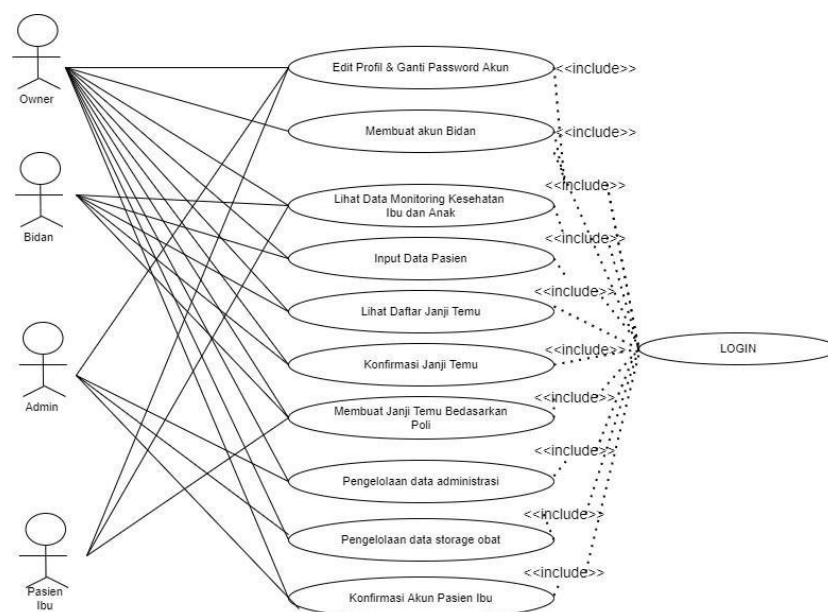
- Menu Monitoring Kesehatan: Monitoring Kesehatan dibagi menjadi 2 bagian, yaitu Kesehatan Ibu dan Anak. Bidan dapat melihat perkembangan data Ibu dan Anak melalui keluaran berupa grafik.
- Menu Input Data Pasien: Mulai dari membuat, mengubah, melihat serta menghapus proses dari setiap pasien ibu yang telah melaksanakan *check up*.

- Menu Lihat Jadwal Janji Temu: Bidan dapat melihat list seluruh janji temu pasien ibu yang akan melakukan kunjungan. Sebelum itu bidan harus mengkonfirmasi jadwal janji temu terlebih dahulu agar kunjungan dapat dilakukan.
- Menu Chat Pasien: Digunakan untuk melayani pasien yang akan konsultasi melalui fitur chat yang sudah disediakan.

Pada akses Pasien Ibu dapat melakukan beberapa fitur sebagai berikut:

- Menu Monitoring Kesehatan: Monitoring Kesehatan dibagi menjadi 2 bagian, yaitu Kesehatan Ibu dan Anak. Bidan dapat melihat perkembangan data Ibu dan Anak melalui keluaran berupa grafik.
- Menu Layanan Klinik: Layanan klinik dibagi menjadi 3 Poli yaitu, Poli Ibu, Poli Anak dan Poli Masas. Pasien ibu memilih layanan berdasarkan Poli yang sudah disediakan, setelah itu melakukan proses janji temu kepada bidan agar dapat melakukan kunjungan.
- Menu Chat Bidan: Layanan chat bidan digunakan untuk konsultasi pasien dengan bidan.
- Menu Find Us: Layanan ini digunakan untuk menemukan templat klinik bidan dari titik koordinat pasien yang dapat menentukan estimasi jarak dan waktu.

3.3 Use Case Diagram



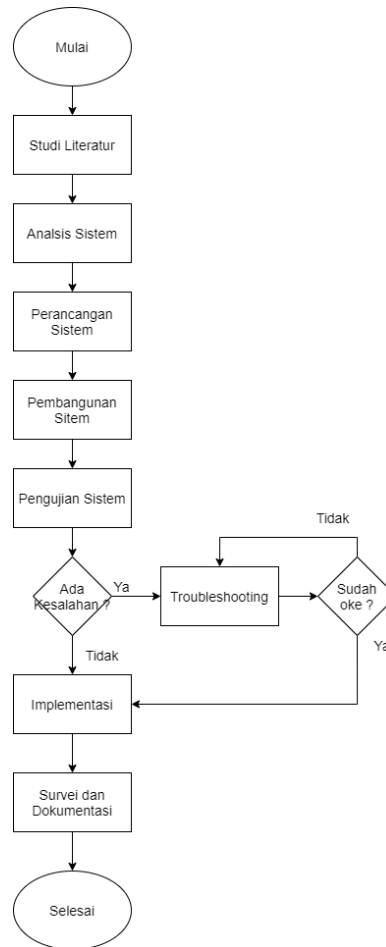
Gambar 3 4 Use Case Diagram

Alir dari aplikasi ini digambarkan dalam *Use Case Diagram*. Adapun kebutuhan aplikasi ini adalah sebagai berikut:

1. Akun Bidan dan Admin diharuskan didaftarkan terlebih dahulu oleh owner sebelum melakukan *login*.
2. *Login* diharuskan sesuai dengan kategori akun yaitu, Owner, Admin, Bidan dan Pasien Ibu.
3. Apabila Owner, Admin, Bidan dan Pasien Ibu sudah login, maka dapat mengakses fitur sesuai dengan kategori akun.
4. Akun Pasien Ibu dapat menggunakan layanan klinik apabila sudah dikonfirmasi oleh akun Owner dan Admin.
5. Owner dapat menggunakan seluruh fitur yang terdapat pada aplikasi E-TOCOLOGIST.
6. Pasien ibu dapat melakukan janji temu apabila Bidan telah mengkonfirmasi jadwal yang telah diajukan.

3.4 Flowchart Pengerjaan Aplikasi

Berikut merupakan *flowchart* dari perancangan aplikasi E-TOCOLOGIST berbasis *website* dan *mobile* aplikasi pada proyek akhir ini:



Gambar 3 5 Flowchart perancangan aplikasi E-TOCOLOGIST

Pada *flowchart* diatas dapat diketahui bahwa tahapan pembuatan aplikasi E-TOCOLOGIST berbasis *website* dan *mobile* aplikasi ini dimulai dari studi literatur untuk meneliti dan mencari informasi dari berbagai sumber. Lalu analisis sistem diperlukan untuk perencanaan pembuatan aplikasi kedepannya. Setelah itu tahap perancangan sistem dilakukan dimana tahap ini dilakukan untuk membuat *prototype* dari aplikasi. Selanjutnya adalah tahap pembangunan sistem, tahap ini merupakan tahap membangun aplikasi. Langkah selanjutnya merupakan tahap pengujian sistem dimana aplikasi dilakukan pengecekan apakah terdapat kesalahan atau tidak, jika terdapat kesalahan maka akan dilakukan perbaikan sampai aplikasi layak untuk dipergunakan. Jika aplikasi sudah layak, maka langkah selanjutnya ke tahap implementasi dan tahap terakhir yaitu tahap survei dan dokumentasi untuk uji kelayakan aplikasi E-TOCOLOGIST.

BAB IV

BENTUK KELUARAN YANG DIHARAPKAN

4.1 Keluaran yang Diharapkan

Perancangan Proyek tingkat akan dibuat sistem informasi untuk klinik kebidanan pada aplikasi E-TOCOLOGIST dengan spesifikasi sebagai berikut:

- a) Dapat merancang sistem aplikasi yang sesuai dengan peforma, kualitas dan keamanan yang baik agar dapat meningkatkan kualitas pelayanan.
- b) Dapat menjalankan sistem aplikasi sesuai dengan permintaan studi kasus, Klinik Bidan Rohaeni. S,ST.
- c) Dapat mengintegrasikan sistem layanan *website* dengan *mobile* aplikasi melalui API.

4.2 Jadwal Pelaksanaan

Adapun jadwal pengerjaan Proyek akhir bisa dilihat pada tabel 4.1 sebagai berikut :

Tabel 4 1 Jadwal Pelaksanaan

Judul Kegiatan	Waktu						
	Nov	Des	Jan	Feb	Mar	Apr	Mei
Studi Literatur							
Analisis Sistem							
Perancangan Sistem							
Pembangunan Sistem							
Pengujian Sistem dan Troubleshooting							
Implementasi							
Survei dan Dokumentasi							

DAFTAR PUSTAKA

- [1] al, M. V. (2015). Evaluating the Monolithic and the Microservice Architecture Pattern to Deploy Web Applications in the Cloud. *IEEE*, 583-590.
- [2] Dahl, R. (2009). *About*. Retrieved from Node Js: <https://id.wikipedia.org/wiki/Node.js>
- [3] Dr. Eng, R. H. (n.d.). *Pemograman Database Menggunakan MySQL*. PENERBIT ANDI.
- [4] Dwi Wijonarko, R. F. (2018). PERBANDINGAN PHONEGAP DAN REACTNATIVE SEBAGAI FRAMEWORK PENGEMBANGAN APLIKASI MOBILE. *Jurnal Manajemen Informatika & Sistem Informasi*, 2614-1701.
- [5] Gonzalez. (2016). *Developing Microservices with Node.js*. Mumbai: PACKT.
- [6] Lili Rusdiana, H. S. (2018). PERANCANGAN APLIKASI MONITORING KESEHATAN IBUHAMIL BERBASIS MOBILE ANDROID. *Jurnal SISTEMASI*, 197–203.
- [7] Luthfi, F. (2017). Penggunaan Framework Laravel Dalam Rancang Bangun Modul Back-End Artikel Website Bisnisbisnis.ID. *JISKA (Jurnal Inform. Sunan Kalijaga)*, 34-41.
- [8] Muhammad Aji Eko Syahputro, E. H. (2018). APLIKASI PENJUALAN DAN PEMESANAN BERBASIS WEB DI SATE GULE KAMBING PAK NI PONOROGO. *e-Proceeding of Applied Science*, 1737.
- [9] Newman, S. (2015). *Building Microservices Designing Fine Grained Systems*. United States of America: O'REILLY.
- [10] Otwell, T. (2021). *Documentation*. Retrieved from Laravel: <https://laravel.com/>
- [11] Sanfilippo, S. (2021). *Redis*. Retrieved from Aws Amazon: <https://aws.amazon.com/id/redis/>
- [12] Solichin, A. (2016). *Pemograman Web dengan PHP dan MySQL*. Universitas Budi Luhur.
- [13] Suryana, D. (2018). *Android Studio: Belajar Bahasa Android Studio*. Books on Google Play.

LAMPIRAN

Surat Kerjasama dengan Klinik Bidan Rohaeni, S.ST,

KLINIK BIDAN ROHAENI BUDIMAN.AMKEB.SST

Alamat : Jl. Encep Kartawiria No.39 A, RT.02/RW.03, Citeureup, Kec. Cimahi Utara, Kota Cimahi, Jawa Barat 40512 Telp/Hp : 0812-2461-2309 Website : <https://bidan-rohaeni.business.site/>

Cimahi, 15 Januari 2021

SURAT REKOMENDASI

Saya yang bertanda tangan di bawah ini :

Jabatan : Pemilik Klinik
NIP : 3277036512700010
Nama : Rohaeni Budiman.Amkeb.SST

Memberikan rekomendasi kepada :

Nama : Muhammad Fajar Nugroho Alam
NIM : 6705180059
Instansi : Telkom University

Untuk memenuhi kebutuhan dengan membuat suatu layanan sistem informasi berbasis Web dan Aplikasi demi menunjang pelayanan administrasi, stok obat, praktek bidan, imunisasi dll. Maka surat ini kami buat dengan bekerja sama dengan saudara Muhammad Fajar Nugroho Alam.

Demikian Surat Kerjasama ini saya buat dengan sebenar-benarnya untuk dipergunakan seperlunya.

Cimahi, 15 Januari 2021

Pemilik Klinik





UNIVERSITAS TELKOM
FAKULTAS ILMU TERAPAN
KARTU KONSULTASI
SEMINAR PROPOSAL PROYEK TINGKAT

NAMA / PRODI : Muhammad Fajar Nugroho Alam/ D3TT


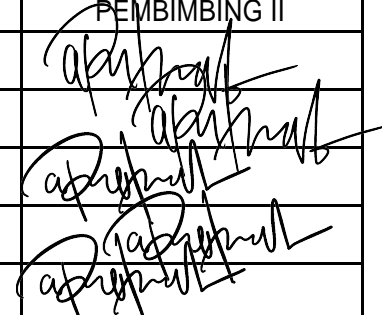
NIM : 6705180059

JUDUL PROYEK TINGKAT :

E-TOCOLOGIST (SISTEM INFORMASI KEBIDANAN UNTUK PELAYANAN MONITORING KESEHATAN
PADA IBU DAN ANAK BERBASIS WEB DAN APLIKASI *MOBILE* DI KLINIK ROHAENI, S.ST)

CALON PEMBIMBING : I. Dadan Nur Ramadhan, S.Pd., M.T

II. Rohmat Tulloh, S.T., M.T

NO	TANGGAL	CATATAN HASIL KONSULTASI	TANDA TANGAN CALON PEMBIMBING I
1		BAB 1 (SELESAI)	
2		BAB 2 (SELESAI)	
3		BAB 3 (SELESAI)	
4		BAB 4 (SELESAI)	
5		FINALISASI PROPOSAL	
6			
7			
8			
9			
10			
NO	TANGGAL	CATATAN HASIL KONSULTASI	TANDA TANGAN CALON PEMBIMBING II
1		BAB 1 (SELESAI)	
2		BAB 2 (SELESAI)	
3		BAB 3 (SELESAI)	
4		BAB 4 (SELESAI)	
5		FINALISASI PROPOSAL	
6			
7			
8			
9			
10			