

LAPORAN SKRIPSI

RANCANG BANGUN APLIKASI POSYANDU BERBASIS MOBILE MENGGUNAKAN FRAMEWORK REACT NATIVE DENGAN FITUR CHATBOT



Disusun Oleh :

**MUHAMMMAD HASAN FATHUL ARIFIN
215430043**

**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK DAN INFORMATIKA
UNIVERSITAS PANCA MARGA
2025**

LAPORAN SKRIPSI

RANCANG BANGUN APLIKASI POSYANDU BERBASIS MOBILE MENGGUNAKAN FRAMEWORK REACT NATIVE DENGAN FITUR CHATBOT

Diajukan kepada
Universitas Panca Marga Probolinggo
untuk memenuhi salah satu persyaratan
dalam menyelesaikan Program Sarjana Teknik
Program Studi Teknik Elektro

Disusun oleh :
MUHAMMAD HASAN FATHUL ARIFIN
215430043

**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK DAN INFORMATIKA
UNIVERSITAS PANCA MARGA PROBOLINGGO
2025**

HALAMAN PERSETUJUAN

1. Judul Skripsi : Rancang Bangun Aplikasi Posyandu
Berbasis Mobile Menggunakan
Framework React Native Dengan Fitur
Chatbot
2. Identitas Mahasiswa :
 - a. Nama Lengkap : Muhammad Hasan Fathul Arifin
 - b. NIM : 215430043
 - c. Program studi : Teknik Elektro
3. Dosen Pembimbing I :
 - a. Nama Lengkap dan Gelar : Ahmad Izzuddin, S.T., M.Kom.
 - b. NIDN : 0710108502
4. Dosen Pembimbing II :
 - a. Nama Lengkap dan Gelar : Nuzul Hikmah, S.T., M.Kom.
 - b. NIDN : 0718079002

Telah diperiksa dan disetujui sebagai persyaratan untuk mengikuti **Ujian Sidang Skripsi** pada tanggal 31 Juli 2025

Menyetujui

Dosen Pembimbing I

Dosen Pembimbing II

Ahmad Izzuddin, S.T., M.Kom.
NIDN 0710108502

Nuzul Hikmah, S.T., M.Kom.
NIDN 0718079002

LEMBAR PENGESAHAN SKRIPSI

RANCANG BANGUN APLIKASI POSYANDU BERBASIS MOBILE MENGUNAKAN FRAMEWORK REACT NATIVE DENGAN FITUR CHATBOT

Yang dipersiapkan dan disusun oleh :
MUHAMMAD HASAN FATHUL ARIFIN
215430043

Telah dipertahankan di depan Dewan Penguji
Pada tanggal : 31 Juli 2025
dan dinyatakan telah memenuhi syarat

Susunan Tim Penguji

Ketua Penguji

Dyah Ariyanti, S.Kom., M.Kom
NIDN 0719078402

Anggota Penguji I

Anggota Penguji II

Ira Aprilia, S.Pd., M.Si.
NIDN 0718048901

Nuzul Hikmah, S.T., M.Kom.
NIDN 0718079002

Probolinggo, 31 Juli 2025

Fakultas Teknik
Universitas Panca Marga
Dekan

Ahmad Izzuddin, S.T., M.Kom.
NIDN 0710108502

BERITA ACARA

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini :

Nama : Muhammad Hasan Fathul Arifin
NIM : 215430043
Program Studi : Teknik Elektro
Fakultas : Teknik dan Informatika

Menyatakan dengan sesungguhnya bahwa skripsi yang saya tulis ini benar benar tulisan saya, dan merupakan plagiasi baik sebagian atau seluruhnya.

Apabila kemudian hari terbukti atau dapat dibuktikan bahwa skripsi ini hasil plagiasi, baik sebagian atau seluruhnya, maka saya bersedia menerima sanksi atas perbuatan dengan ketentuan yang berlaku.

Probolinggo, 11 Agustus 2025

Yang membuat Pernyataan

Muhammad Hasan Fathul Arifin

RANCANG BANGUN APLIKASI POSYANDU BERBASIS MOBILE MENGGUNAKAN FRAMEWORK REACT NATIVE DENGAN FITUR CHATBOT

Nama Mahasiswa : Muhammad Hasan Fathul Arifin
NIM : 215430043
Pembimbing I : Ahmad Izzuddin, S.T., M.Kom
Pembimbing II : Nuzul Hikmah, S.T., M.Kom

ABSTRAK

Posyandu merupakan layanan kesehatan masyarakat yang memiliki peran penting dalam pemantauan tumbuh kembang anak serta penyediaan informasi kesehatan kepada masyarakat. Namun, proses pencatatan data dan penyampaian informasi di Posyandu masih dilakukan secara manual, sehingga menimbulkan risiko kehilangan data, keterlambatan pelaporan, serta terbatasnya akses terhadap informasi. Untuk mengatasi permasalahan tersebut, penelitian ini merancang aplikasi Posyandu berbasis mobile menggunakan *framework React Native* yang dilengkapi dengan fitur chatbot berbasis AI.

Aplikasi ini dikembangkan menggunakan metode *waterfall* yang mencakup tahapan analisis, perancangan, implementasi, pengujian, dan pemeliharaan. Fitur utama yang dikembangkan meliputi pencatatan data balita secara digital, pemantauan tumbuh kembang anak, serta chatbot yang dapat memberikan informasi seputar imunisasi, pencegahan stunting, dan konsultasi kesehatan lainnya. Chatbot dibangun dengan metode pemrosesan teks (*case folding, tokenizing, stopword removal, dan stemming*) serta pendekatan TF-IDF dan Cosine Similarity untuk menghasilkan respons berdasarkan kemiripan teks.

Hasil pengujian menunjukkan bahwa seluruh fitur aplikasi berhasil dijalankan sesuai dengan fungsinya (*Black Box Testing*). Pada tahap *User Acceptance Testing* (UAT), aplikasi memperoleh tingkat kepuasan pengguna sebesar 75,83%, dengan kategori "Setuju". Namun demikian, hasil evaluasi menunjukkan bahwa fitur chatbot mendapatkan nilai paling rendah dibandingkan fitur lainnya, karena banyak pengguna memberikan penilaian netral hingga tidak setuju terhadap akurasi dan kecepatan jawaban. Hal ini menunjukkan bahwa meskipun secara teknis chatbot telah berjalan dengan baik, masih diperlukan pengembangan dataset dan peningkatan pemahaman konteks agar chatbot dapat memberikan jawaban yang lebih tepat dan relevan.

Kata kunci : Posyandu, Aplikasi, Mobile, React Native, Chatbot, UAT

RANCANG BANGUN APLIKASI POSYANDU BERBASIS MOBILE MENGGUNAKAN FRAMEWORK REACT NATIVE DENGAN FITUR CHATBOT

By : Muhammad Hasan Fathul Arifin
Student Identity Number : 215430043
Advisor I : Ahmad Izzuddin, S.T., M.Kom
Advisor II : Nuzul Hikmah, S.T., M.Kom

ABSTRACT

Posyandu (Integrated Health Service Post) plays a crucial role in monitoring child development and providing health-related information to the community. However, data recording and information dissemination at Posyandu are still conducted manually, which can lead to data loss, delayed reporting, and limited access to information. To address these issues, this study designed a mobile-based Posyandu application using the React Native framework, equipped with an AI-based chatbot feature.

The application was developed using the waterfall methodology, including analysis, design, implementation, testing, and maintenance stages. The main features include digital recording of child health data, growth monitoring, and a chatbot that provides information on immunization, stunting prevention, and other health topics. The chatbot was built using text processing methods (case folding, tokenizing, stopword removal, and stemming) and applies the TF-IDF and Cosine Similarity approach to generate responses based on text similarity.

The test results showed that all application features functioned correctly (Black Box Testing). In the User Acceptance Testing (UAT) phase, the application received a user satisfaction score of 75.83%, categorized as "Agree". However, evaluation revealed that the chatbot feature received the lowest score compared to other features, as many users rated it neutral to disagree in terms of response accuracy and speed. This indicates that while the chatbot performs well technically, further improvements are needed, particularly in expanding the dataset and enhancing contextual understanding, so it can deliver more accurate and relevant responses.

Keyword : *Posyandu, Mobile Application, Mobile, React Native, Chatbot, UAT*

KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayah Nya sehingga penulis dapat menyelesaikan skripsi dengan judul "Rancang Bangun Aplikasi Posyandu Berbasis Mobile Menggunakan Framework React Native Dengan Fitur Chatbot" sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro Fakultas Teknik dan Informatika Universitas Panca Marga.

Penyusunan laporan skripsi ini tidak lepas dari bimbingan, arahan, dan bantuan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Bapak Ahmad Izzuddin, S.T., M.Kom., selaku Dekan Fakultas Teknik dan Informatika Universitas Panca Marga yang telah memberikan izin penelitian kepada penulis, serta selaku Dosen Pembimbing I yang telah memberikan bimbingan, arahan, dan masukan kepada penulis dalam menyelesaikan skripsi ini.
2. Ibu Nuzul Hikmah, S.T., M.Kom., selaku Ketua Program Studi Teknik Elektro Universitas Panca Marga yang telah memberikan arahan dan bimbingan selama proses perkuliahan, serta selaku Dosen Pembimbing II telah memberikan bimbingan, arahan, dan masukan kepada penulis dalam menyelesaikan skripsi ini.
3. Seluruh Dosen dan Staf Fakultas Teknik & Informatika Universitas Panca Marga yang telah memberikan ilmu dan bantuan selama proses perkuliahan.
4. Kedua orang tua, saudara, dan keluarga tercinta yang senantiasa memberikan doa, motivasi, dan dukungan kepada penulis.

5. Teman-teman seperjuangan yang telah memberikan semangat dan dukungan kepada penulis, khususnya teman-teman grup K... yang merupakan grup dari awal kuliah hingga saat ini.

Penulis menyadari bahwa proposal skripsi ini masih jauh dari sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun dari semua pihak demi perbaikan di masa yang akan datang. Semoga proposal skripsi ini dapat bermanfaat bagi penulis dan pembaca sekalian.

Probolinggo, 31 Juli 2025

Muhammad Hasan Fathul Arifin

DAFTAR ISI

LAPORAN SKRIPSI.....	i
HALAMAN PERSETUJUAN	ii
LEMBAR PENGESAHAN SKRIPSI	iii
BERITA ACARA.....	iv
PERNYATAAN KEASLIAN TULISAN	v
ABSTRAK.....	vi
ABSTRACT.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian	4
1.4 Batasan Masalah	4
1.5 Fokus Penelitian	5
1.6 Manfaat Penelitian	5
BAB II TINJAUAN PUSTAKA	6
2.1 Penelitian Terdahulu	6
2.2 Kerangka Pemikiran	11
2.3 Hipotesis	13
2.4 Definisi Operasional.....	13
2.4.1 Aplikasi Posyandu Berbasis <i>Mobile</i>	13
2.4.2 <i>React Native</i>	14
2.4.3 Chatbot Berbasis AI.....	14
2.4.4 Metode <i>Waterfall</i>	14
2.4.5 UML (<i>Unified Modeling Language</i>).....	17
BAB III METODOLOGI	21
3.1 Desain Penelitian.....	21
3.1.1 Alur Penelitian.....	21
3.1.2 Tempat Penelitian	32
3.1.3 Waktu dan Jadwal Penelitian.....	33
3.1.4 Bahan Penelitian	33
3.1.5 Alat Penelitian	33
3.1.6 Tatalaksana Penelitian	34
3.2 Variable Penelitian	34
3.2.1 Variable Bebas.....	35
3.2.2 Variabel Terkait.....	35
3.2.3 Variabel Kontrol.....	35
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA	36
4.1 Pengumpulan Data	36
4.2 Pengolahan Data	37
4.2.1 Penyajian Data.....	37
4.2.2 Analisa Kebutuhan	37
4.2.3 Data untuk Implementasi Fitur Penimbangan	38

4.2.4 Pemrosesan Data Chatbot	40
BAB V HASIL DAN PEMBAHASAN.....	53
5.1 Hasil Penelitian	53
5.1.1 Langkah Implementasi	53
5.1.2 Tampilan dan Fitur Aplikasi.....	56
5.2 Pembahasan	71
5.2.1 Kesesuaian Sistem dengan Spesifikasi.....	71
5.2.2 Pembahasan Akurasi Chatbot (Simulasi)	79
5.2.3 Implikasi dan Harapan	81
BAB VI KESIMPULAN DAN SARAN.....	83
6.1 Kesimpulan	83
6.2 Saran.....	84
DAFTAR PUSTAKA.....	86
LAMPIRAN	88

DAFTAR TABEL

Tabel 2.1 Perbandingan Penelitian Terdahulu	9
Tabel 2.2 Komponen <i>Use Case</i>	17
Tabel 2.3 Komponen <i>Activity Diagram</i>	18
Tabel 4.1 Ringkasan Data	37
Tabel 4.2 Kategori Status Gizi Berdasarkan Nilai BMI	38
Tabel 4.4 Kategori Status Pertumbuhan Berdasarkan Tinggi Badan	39
Tabel 4.5 Perhitungan (<i>TF</i>) <i>Term Frequency</i>	46
Tabel 4.6 Perhitungan <i>IDF</i> (<i>Inverse Document Frequency</i>)	47
Table 4.7 Perhitungan <i>Term Weighting</i>	48
Tabel 4.8 Implementasi <i>cosine similarity</i>	50
Tabel 5.1 Hasil Pengujian Sistem Admin dan Kader	72
Tabel 5.2 Hasil Pengujian Sistem Bidan	73
Tabel 5.3 Hasil Pengujian Sistem Orang Tua	73
Tabel 5.4 Skor Jawaban	75
Tabel 5.5 Kriteria Skor	75
Tabel 5.6 Kuisisioner Pengguna	75
Tabel 5.7 Hasil Simulasi pada Beberapa <i>Device</i>	77
Tabel 5.8 Dataset Dokumen Chatbot Berdasarkan Kategori	79
Tabel 5.9 <i>Query</i> (Pertanyaan Pengguna) untuk Uji Chatbot	80

DAFTAR GAMBAR

Gambar 2.1 Kerangka Pemikiran	11
Gambar 2.2 Metode <i>Waterfall</i> (Sumber : (Harry et al., 2024)).....	15
Gambar 3.1 <i>Use Case Diagram</i>	22
Gambar 3.2 <i>Activity Diagram</i> Login	25
Gambar 3.3 <i>Activity Diagram</i> Mengelola Pengguna/User.....	26
Gambar 3.4 <i>Activity Diagram</i> untuk Mengelola Data Posyandu	26
Gambar 3.5 <i>Activity Diagram</i> Bidan untuk Melihat Data Balita	27
Gambar 3.6 <i>Activity Diagram</i> Orang Tua untuk Melihat Data Anak.....	27
Gambar 3.7 <i>Class Diagram</i>	28
Gambar 3.8 Arsitektur Sistem Chatbot (Nuzul Hikmah et al., 2022).....	31
Gambar 4.1 Gambar code untuk proses <i>Case Folding</i>	41
Gambar 4.2 Gambar code untuk proses <i>Tokenizing</i>	41
Gambar 4.3 Gambar code untuk proses <i>Stopword Removal</i>	42
Gambar 4.4 Gambar code untuk proses <i>Stemming</i>	42
Gambar 4.5 Alur Kerja Chatbot Posyandu.....	43
Gambar 4.6 Gambar code TF-IDF.....	49
Gambar 4.7 Gambar code <i>Cosine Similarity</i>	52
Gambar 5.1 Halaman Welcome dan Login.....	57
Gambar 5.2 Dashboard Admin dan Monitoring bagian 1.....	58
Gambar 5.3 Dashboard Admin dan Monitoring bagian 2.....	59
Gambar 5.4 Dashboard Kader	60
Gambar 5.5 Dashboard Bidan	61
Gambar 5.6 Dashboard Orang Tua bagian 1	62
Gambar 5.7 Dashboard Orang Tua bagian 2	63
Gambar 5.8 Profil dan Ubah Password.....	64
Gambar 5.9 Pengelolaan Pengguna bagian 1	65
Gambar 5.10 Pengelolaan Pengguna bagian 2	66
Gambar 5.11 Pengelolaan Pengguna bagian 3	67
Gambar 5.12 Data Posyandu	68
Gambar 5.13 Simulasi Export Data bagian 1	69
Gambar 5.14 Simulasi Export Data bagian 2	70
Gambar 5.15 Fitur Chatbot.....	71

BAB I

PENDAHULUAN

1.1 Latar Belakang

Posyandu atau Pos Pelayanan Terpadu merupakan program kesehatan masyarakat yang penting di Indonesia, bertujuan meningkatkan kualitas kesehatan anak-anak, perempuan, dan lansia (Ramadhan Sukoco et al., 2022). Kegiatan posyandu meliputi pencatatan data balita, pemantauan tumbuh kembang anak, pemberian imunisasi, hingga edukasi kesehatan seperti pencegahan stunting. Di Desa Jabung Sisir, kegiatan Posyandu dikelola oleh 25 kader yang bertanggung jawab terhadap berbagai aspek layanan kesehatan masyarakat. Namun, dalam praktiknya, dalam pencatatan data dan penyampaian informasi serta pengumpulan laporan ke puskesmas masih menggunakan metode manual.

Metode pencatatan data di posyandu saat ini masih dilakukan secara manual menggunakan buku register atau Kartu Menuju Sehat (KMS). Selain itu, masyarakat, khususnya orang tua, juga masih mengandalkan buku manual seperti KMS untuk memantau tumbuh kembang anak di rumah. Informasi mengenai tinggi badan, berat badan, serta status gizi balita biasanya dicatat secara manual oleh kader dan dilihat oleh orang tua saat kegiatan posyandu berlangsung. Sementara itu, penyampaian informasi kesehatan seperti jadwal imunisasi, pentingnya pemantauan stunting, dan edukasi tumbuh kembang anak dilakukan secara langsung hanya pada saat kegiatan posyandu berlangsung.

Metode manual memiliki sejumlah kelebihan, seperti kemudahan dalam pelaksanaannya tanpa memerlukan pelatihan tambahan serta memungkinkan terjadinya komunikasi dua arah secara langsung antara kader posyandu dan masyarakat. Namun, metode ini juga memiliki beberapa kelemahan signifikan,

seperti tingginya risiko kehilangan data akibat pencatatan manual, potensi kesalahan dalam pencatatan, keterlambatan dalam pembuatan laporan, serta terbatasnya informasi yang dapat disampaikan kepada masyarakat. Selain itu, aksesibilitas menjadi terbatas karena informasi hanya dapat diberikan selama kegiatan posyandu berlangsung, dan terdapat ketergantungan pada kehadiran kader posyandu untuk menyampaikan informasi. Kelemahan-kelemahan ini menjadikan metode manual kurang efisien dan kurang dapat diandalkan untuk menjangkau masyarakat secara luas dan konsisten.

Banyak peneliti telah mengusulkan sistem berbasis aplikasi *mobile* ataupun berbasis website untuk memfasilitasi penyebaran informasi di bidang kesehatan. Aplikasi berbasis *mobile* memiliki kelebihan dalam mengotomatisasi pencatatan, mempercepat distribusi informasi, dan meningkatkan efisiensi layanan Posyandu (Anwar et al., 2023). Pendekatan pengembangan aplikasi *mobile* secara native menggunakan Java/Kotlin untuk Android dan Swift/Objective-C untuk iOS memungkinkan performa optimal karena setiap platform dikembangkan dengan teknologi spesifik. Namun, metode ini memerlukan sumber daya besar, baik dari segi waktu, biaya, maupun tenaga ahli, karena pengembangan untuk kedua platform harus dilakukan secara terpisah. Selain menguasai bahasa pemrograman seperti Java, Kotlin, Swift, dan Objective-C, pengembang juga harus memahami alat pengembangan seperti Android Studio untuk Android dan Xcode untuk iOS, yang menambah kompleksitas proses (Atmaji Perdana et al., 2023).

Penelitian ini mengusulkan rancang bangun aplikasi posyandu berbasis *mobile* menggunakan *framework React Native* yang dilengkapi dengan fitur chatbot berbasis AI. *Framework React Native* menjadi pilihan yang tepat untuk

pengembangan aplikasi ini karena kemampuannya menghasilkan aplikasi lintas platform (iOS dan Android) dengan satu basis kode. *Framework* ini memecahkan masalah pengembangan konvensional yang membutuhkan bahasa pemrograman terpisah seperti Swift untuk iOS dan Kotlin atau Java untuk Android, sehingga meningkatkan efisiensi dalam proses pengembangan (Harry et al., 2024).

Chatbot berbasis AI diintegrasikan sebagai media konseling untuk membantu dalam mengakses informasi kesehatan, khususnya mengenai stunting serta menyediakan panduan intervensi dini. Menurut Anindyati (2023), penggunaan chatbot berbasis AI dinilai efektif dalam meningkatkan pemahaman masyarakat terkait isu kesehatan seperti stunting.

Proses penelitian ini menggunakan metode *waterfall*. Metode *waterfall* merupakan pendekatan yang terstruktur dan fleksibel dalam pengembangan aplikasi. Prosesnya meliputi beberapa tahap seperti penelitian, analisis, desain, pengkodean, pengujian, implementasi, dan pemeliharaan, dengan keunggulan mudah dipahami dan diterapkan dalam pengembangan perangkat lunak (Santoso & Anggara, 2021).

Tujuan penelitian ini adalah untuk merancang dan mengembangkan aplikasi posyandu berbasis mobile yang tidak hanya mencatat data balita secara digital, tetapi juga menyediakan solusi inovatif berupa chatbot berbasis AI sebagai media konseling kesehatan. Sistem ini diharapkan mampu meningkatkan operasional posyandu terutama di posyandu desa Jabung Sisir serta memperluas cakupan edukasi kesehatan masyarakat, dan mendukung kader dalam memberikan informasi kesehatan yang lebih berkualitas dan membantu dalam pengumpulan laporan ke puskesmas secara digitalisasi. Selain itu, aplikasi ini juga bertujuan agar orang tua

dapat dengan mudah memantau tumbuh kembang anak secara mandiri melalui aplikasi kapan saja dan di mana saja, tanpa harus menunggu kegiatan posyandu berikutnya.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang dibuat, maka rumusan masalah pada penelitian ini ialah Bagaimana cara merancang aplikasi Posyandu berbasis *mobile* menggunakan *framework React Native* dengan fitur chatbot untuk mendukung layanan kesehatan posyandu di Desa Jabung Sisir?

1.3 Tujuan Penelitian

Adapun penelitian ini bertujuan untuk mengembangkan aplikasi Posyandu berbasis *mobile* menggunakan *React Native* untuk mengatasi masalah pencatatan dan pelaporan data posyandu secara manual, dan membantu orang tua dalam memantau tumbuh kembang anak, serta menerapkan fitur chatbot berbasis AI sebagai media konseling untuk meningkatkan interaksi serta komunikasi antara kader posyandu dan masyarakat, terutama dalam mengakses informasi kesehatan.

1.4 Batasan Masalah

Pada penelitian ini, batasan masalah yang dibuat yaitu sebagai berikut :

1. Aplikasi yang dikembangkan difokuskan untuk digunakan oleh Posyandu di Desa Jabung Sisir dan belum mencakup penerapan di wilayah lain.
2. Aplikasi *mobile* yang dikembangkan menggunakan *framework React Native* untuk mendukung lintas platform (Android dan iOS), dengan fitur chatbot berbasis AI dirancang sebagai alat untuk media konseling dengan memberikan informasi dasar terkait kesehatan anak dan terutama terkait pencegahan stunting.

1.5 Fokus Penelitian

Adapun penelitian berfokus pada :

1. Perancangan dan Implementasi Aplikasi Posyandu Berbasis *Mobile*

Fokus utama adalah pada pengembangan aplikasi *mobile* Posyandu menggunakan framework *React Native* yang dapat berjalan di *platform* iOS dan Android, dengan tujuan untuk meningkatkan kemudahan dalam pencatatan dan pengelolaan data Posyandu secara digital, serta meminimalkan risiko kesalahan entri data.

2. Penggunaan Chatbot Berbasis AI untuk Mendukung Informasi Kesehatan

Meneliti integrasi fitur chatbot berbasis AI untuk memberikan informasi terkait kesehatan, khususnya pengetahuan tentang stunting, dengan tujuan meningkatkan akses informasi kesehatan secara cepat dan mudah bagi pengguna (kader dan masyarakat).

1.6 Manfaat Penelitian

Adapun manfaat penelitian ini adalah sebagai berikut :

1. Memberikan akses lebih mudah terhadap informasi layanan Posyandu, seperti jadwal kegiatan, dan edukasi kesehatan. Serta memudahkan orang tua dalam memantau data anak
2. Memudahkan proses pencatatan data, sehingga lebih terorganisir
3. Mengurangi risiko kesalahan dalam pengelolaan data, sehingga informasi yang disimpan lebih terpercaya.
4. Mempermudah dalam membuat laporan.

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Dalam penelitian sebelumnya, telah dilakukan beberapa studi terkait dengan pengembangan aplikasi berbasis mobile dan implementasi chatbot. Beberapa penelitian terkait dengan topik ini adalah:

Penelitian yang dilakukan Fikran Anwar, Bahtiar Senung, A. Mulawati Mas Pratama, dan Abdul Rahman Ismail (2023) berjudul "Aplikasi Pos Pelayanan Terpadu (Posyandu) Berbasis Android" bertujuan untuk mengembangkan aplikasi Android di Puskesmas Buloila, Gorontalo Utara. Aplikasi ini dirancang untuk mempermudah kegiatan Posyandu, seperti penyampaian informasi jadwal dan pengelolaan data kesehatan anak serta ibu hamil, yang sebelumnya dilakukan secara manual. Hasil penelitian menunjukkan bahwa aplikasi ini mampu memberikan kemudahan bagi peserta Posyandu dalam mendapatkan informasi secara real-time melalui *smartphone*, serta membantu petugas dalam mencatat data kesehatan secara digital sehingga mengurangi risiko kehilangan dokumen. Pengujian *whitebox* membuktikan bahwa alur logika sistem berjalan efisien, sementara pengujian *blackbox* menunjukkan semua fitur seperti login, penjadwalan, pemeriksaan data, dan konsultasi berfungsi sesuai harapan. Fitur konsultasi juga memungkinkan interaksi langsung antara peserta dan petugas dengan menyimpan pertanyaan serta jawaban di database, memastikan informasi kesehatan selalu dapat diakses kapan saja. Dengan aplikasi ini, penyampaian informasi menjadi lebih cepat dan akurat, serta mendukung pembuatan laporan perkembangan kesehatan dengan lebih efisien. Penelitian ini menggunakan metode *Research and Development (R&D)*

dengan bahasa pemrograman Java untuk aplikasi Android, PHP untuk server, dan MySQL sebagai basis data (Anwar et al., 2023).

Penelitian yang dilakukan oleh Briyan Gifari Aji dan Muhamad Awiet Wiedanto Prasetyo (2024) berjudul "Perancangan Sistem Informasi Posyandu Berbasis Web dengan Metode *Extreme Programming* pada Desa Candinata" bertujuan untuk mengembangkan sistem informasi posyandu berbasis web di Posyandu Melati, Desa Candinata. Posyandu ini memiliki peran penting dalam melayani kesehatan ibu dan anak di lingkungan desa, namun pencatatan data yang dilakukan secara manual seringkali menyebabkan keterlambatan dalam distribusi informasi kepada masyarakat. Sistem informasi yang dikembangkan menggunakan pendekatan *Extreme Programming* (XP), sebuah metode yang cocok untuk tim kecil dengan siklus pengembangan cepat, yang juga memastikan sistem dapat merespon kebutuhan pengguna secara efektif. Penggunaan diagram *Unified Modeling Language* (UML) seperti use case dan class diagram membantu dalam merancang sistem yang sederhana dan mudah digunakan. Bahasa pemrograman yang digunakan dalam implementasi sistem ini adalah PHP, dengan basis data MySQL untuk pengelolaan data kesehatan balita. Berdasarkan hasil pengujian menggunakan metode *Black Box Testing*, sistem ini terbukti mampu menjalankan fungsi-fungsi yang diperlukan dengan baik, mulai dari pencatatan data balita hingga penyebaran informasi kesehatan. Adanya sistem ini mempermudah kader dalam mengelola data balita secara digital dan memungkinkan orang tua untuk menerima informasi kesehatan anak mereka secara real-time. Dengan demikian, penelitian ini menunjukkan bahwa digitalisasi pencatatan posyandu melalui pendekatan XP tidak

hanya meningkatkan efisiensi kerja kader, tetapi juga mempercepat akses informasi bagi masyarakat (Gifari Aji & Prasetyo, 2024).

Penelitian yang dilakukan oleh Putu Ika Farmani, I Nyoman Mahayasa Adiputra, dan Putu Ayu Laksmini (2021) berjudul "Perancangan Sistem Informasi Posyandu Sebagai Upaya Digitalisasi Data Posyandu di UPTD Puskesmas II Dinas Kesehatan Kecamatan Denpasar Timur" bertujuan untuk merancang sistem informasi berbasis web yang dapat membantu mengatasi masalah pencatatan dan pelaporan data posyandu yang sebelumnya dilakukan secara manual. Studi ini menggunakan metode *Waterfall*, dengan tahapan mulai dari identifikasi kebutuhan, desain sistem, implementasi, pengujian unit, hingga integrasi. Hasil penelitian menunjukkan bahwa sistem informasi ini mampu mengurangi pencatatan data berulang yang dilakukan oleh kader, meningkatkan ketepatan dan keseragaman pelaporan, serta membantu petugas puskesmas dalam pengintegrasian data posyandu yang dapat digunakan untuk pelaporan surveilans gizi. Dengan adanya sistem ini, kegiatan posyandu dapat lebih efektif dan efisien, serta memudahkan dalam penyimpanan data secara digital. Penelitian ini berperan penting dalam mendukung upaya digitalisasi data posyandu sebagai solusi pencatatan dan pelaporan bagi kader dan petugas puskesmas di UPTD Puskesmas II Denpasar Timur (Farmani et al., 2021).

Penelitian yang dilakukan oleh Dwi Yuli Prasetyo, Fitri Yunita, dan Ricky Setiawan (2021) berjudul "Aplikasi ePosyandu Kesehatan Ibu dan Anak di Kecamatan Kempas Jaya" bertujuan untuk mengembangkan sistem informasi posyandu berbasis web di Posyandu Srikandi II. Posyandu ini melayani kesehatan ibu dan anak di wilayah Kelurahan Kempas Jaya, Kecamatan Kempas. Penelitian

ini dilatarbelakangi oleh proses manajemen data yang masih manual, yang menyebabkan kesulitan dalam pencatatan dan pelaporan kegiatan posyandu. Sistem informasi yang dikembangkan menggunakan metode *Unified Modeling Language* (UML) dan diimplementasikan dengan bahasa pemrograman PHP serta basis data MySQL. Berdasarkan hasil pengujian dengan metode *black box* dan *white box*, sistem ini mampu memfasilitasi pengelolaan data ibu dan anak secara efisien serta mempermudah akses data di seluruh lingkungan posyandu. Adanya sistem ini juga membantu kader posyandu dalam menghasilkan laporan kunjungan secara berkala sehingga meningkatkan efektivitas dalam memberikan layanan kesehatan bagi masyarakat setempat (Prasetyo et al., 2021).

Berdasarkan beberapa penelitian terdahulu yang dikutip, diketahui bahwa penelitian-penelitian tersebut berfokus pada perancangan atau pengembangan sistem, baik berbasis website maupun aplikasi Android. Tujuan utama dari penelitian-penelitian ini adalah untuk menggantikan metode pencatatan, pelaporan, dan pemeritahuan jadwal posyandu secara manual dengan metode digital yang terintegrasi.

Tabel 2.1 Perbandingan Penelitian Terdahulu

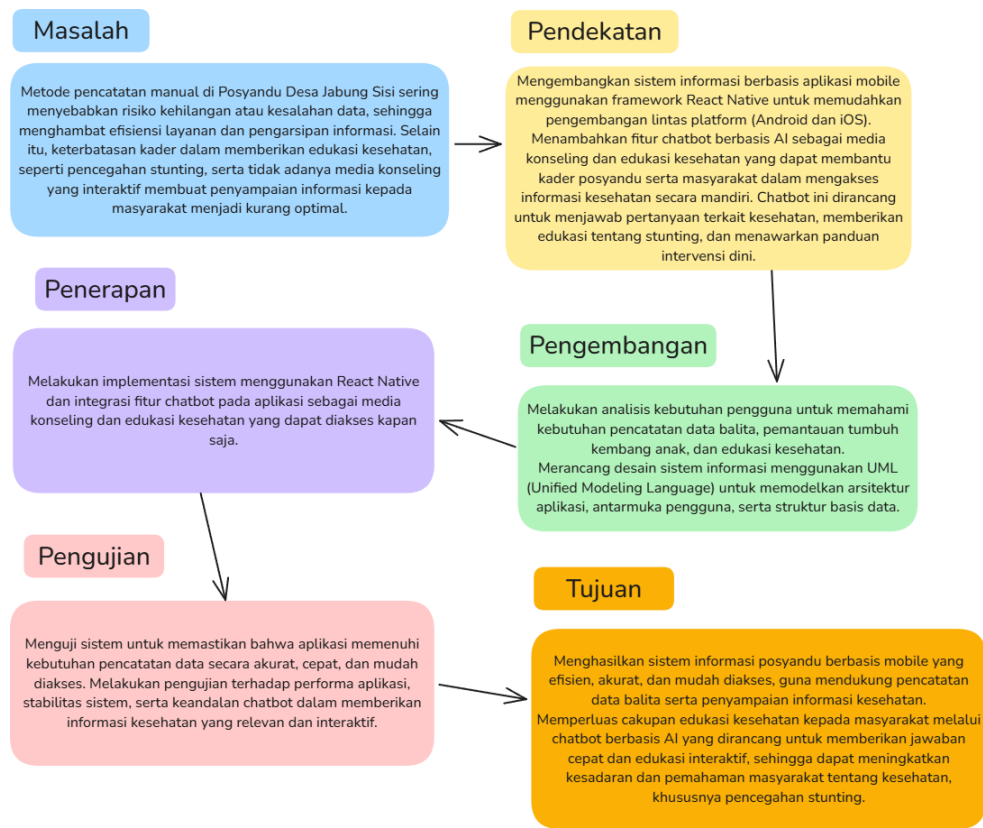
NO	Penelitian	Gap Penelitian		Fokus Penelitian	Media Konseling berbasis AI
		Versi Aplikasi	Metode Pengembangan		
1	Anwar et al., 2023	Android / Mobile	Research and Development (R&D)	Pengembangan aplikasi pengelolaan data posyandu dan penyampaian jadwal berbasis android di puskesmas buloila	Tidak ada

Tabel 2.1 Lanjutan

NO	Penelitian	Gap Penelitian		Fokus Penelitian	Media Konseling berbasis AI
		Versi Aplikasi	Metode Pengembangan		
2	Gifari Aji & Prasetyo, 2024	Website	Extreme Programming	Perancangan system pencatatan dan pengolahan data balita secara real-time di Posyandu Melati, Desa Candinata,	Tidak ada
3	Farmani et al., 2021	Website	<i>Waterfall</i>	Pengembangan sistem informasi berbasis web untuk digitalisasi pencatatan dan pelaporan data posyandu di UPTD Puskesmas II Denpasar Timur.	Tidak ada
4	(Prasetyo et al., 2021)	Website	<i>Waterfall</i>	Pengembangan aplikasi ePosyandu berbasis web untuk mengelola data kesehatan ibu dan anak di Posyandu Srikandi II, Kelurahan Kempas Jaya, Kecamatan Kempas.	Tidak ada
5	Penelitian yang diusulkan	Android/mobile	<i>Waterfall</i>	Pengembangan aplikasi pencatatan dan pengelolaan data Posyandu berbasis android yang dilengkapi fitur chatbot sebagai media konseling di Posyandu Jabung Sisir.	Ada

Berdasarkan tabel perbandingan penelitian terdahulu, dapat diidentifikasi bahwa perbedaan utama penelitian ini dibandingkan dengan penelitian sebelumnya adalah **ketersediaan fitur chatbot berbasis AI**, yang belum diterapkan pada penelitian terdahulu.

2.2 Kerangka Pemikiran



Gambar 2.1 Kerangka Pemikiran

1. Masalah

Metode pencatatan manual di Posyandu Desa Jabung Sisi sering menyebabkan risiko kehilangan atau kesalahan data, sehingga menghambat efisiensi layanan dan pengarsipan informasi. Selain itu, keterbatasan kader dalam memberikan edukasi kesehatan, seperti pencegahan stunting, serta tidak adanya media konseling yang interaktif membuat penyampaian informasi kepada masyarakat menjadi kurang optimal.

2. Pendekatan

Mengembangkan aplikasi berbasis *mobile* menggunakan *framework React Native* untuk memudahkan pengembangan lintas platform (Android dan iOS). Menambahkan fitur chatbot berbasis AI sebagai media konseling dan

edukasi kesehatan yang dapat membantu kader posyandu serta masyarakat dalam mengakses informasi kesehatan secara mandiri. Chatbot ini dirancang untuk menjawab pertanyaan terkait kesehatan, memberikan edukasi tentang stunting, dan menawarkan panduan intervensi dini.

3. Pengembangan

Melakukan analisis kebutuhan pengguna untuk memahami kebutuhan pencatatan data balita, pemantauan tumbuh kembang anak, dan edukasi kesehatan. Merancang desain sistem informasi menggunakan UML (*Unified Modeling Language*) untuk memodelkan arsitektur aplikasi, antarmuka pengguna, serta struktur basis data.

4. Penerapan

Melakukan implementasi sistem dengan memanfaatkan *framework React Native* untuk membangun aplikasi lintas platform, serta mengintegrasikan fitur chatbot sebagai sarana konseling dan edukasi kesehatan yang memungkinkan pengguna mengakses informasi secara fleksibel, kapan pun dibutuhkan.

5. Pengujian

Menguji sistem untuk memastikan bahwa aplikasi memenuhi kebutuhan pencatatan data secara akurat, cepat, dan mudah diakses. Melakukan pengujian terhadap performa aplikasi, stabilitas sistem, serta keandalan chatbot dalam memberikan informasi kesehatan yang relevan dan interaktif.

6. Tujuan

Menghasilkan aplikasi posyandu berbasis *mobile* yang efisien, akurat, dan mudah diakses, guna mendukung pencatatan data balita serta penyampaian

informasi kesehatan. Memperluas cakupan edukasi kesehatan kepada masyarakat melalui chatbot berbasis AI yang dirancang untuk memberikan jawaban cepat dan edukasi interaktif, sehingga dapat meningkatkan kesadaran dan pemahaman masyarakat tentang kesehatan, khususnya pencegahan stunting.

2.3 Hipotesis

Berdasarkan topik penelitian yang berfokus pada "Rancang Bangun Aplikasi Posyandu Berbasis *Mobile* Menggunakan *Framework React Native* dengan Fitur Chatbot", berikut adalah hipotesis yang dihasilkan :

1. Hipotesis Alternatif (H1)

Penggunaan aplikasi berbasis *React Native* dengan chatbot AI dapat meningkatkan efisiensi pencatatan dan penyampaian informasi di Posyandu.

2. Hipotesis Nol (H0)

Penggunaan aplikasi berbasis *React Native* dengan chatbot AI tidak memberikan peningkatan efisiensi dalam pencatatan dan penyampaian informasi di Posyandu.

2.4 Definisi Operasional

Berikut definisi operasional pada penelitian yang mengenai "Rancang Bangun Aplikasi Posyandu Berbasis *Mobile* Menggunakan *Framework React Native* dengan Fitur Chatbot".

2.4.1 Aplikasi Posyandu Berbasis *Mobile*

Aplikasi Posyandu berbasis *mobile* adalah aplikasi yang dirancang untuk memfasilitasi layanan Posyandu dengan menyediakan fitur-fitur seperti pencatatan data kesehatan dan pemberian informasi kepada pengguna. Aplikasi ini dibuat dengan tujuan untuk mendukung pengelolaan layanan Posyandu, terutama dalam

hal mencatat data balita, memberikan akses informasi jadwal Posyandu, serta memudahkan navigasi bagi para pengguna. Keberhasilan aplikasi ini dapat diukur melalui berbagai aspek, termasuk jumlah fitur yang berhasil diimplementasikan dan sejauh mana aplikasi mempermudah pengguna dalam mengakses informasi yang relevan dengan layanan Posyandu.

2.4.2 React Native

React Native adalah *framework* yang memungkinkan pengembangan aplikasi mobile lintas platform, seperti Android dan iOS, dengan menggunakan satu basis kode JavaScript (Harry et al., 2024). Keberhasilan pengembangan aplikasi dengan *React Native* diukur dari seberapa kompatibel aplikasi tersebut di kedua platform, serta seberapa baik kinerja aplikasi tersebut secara teknis dalam mendukung penggunaannya.

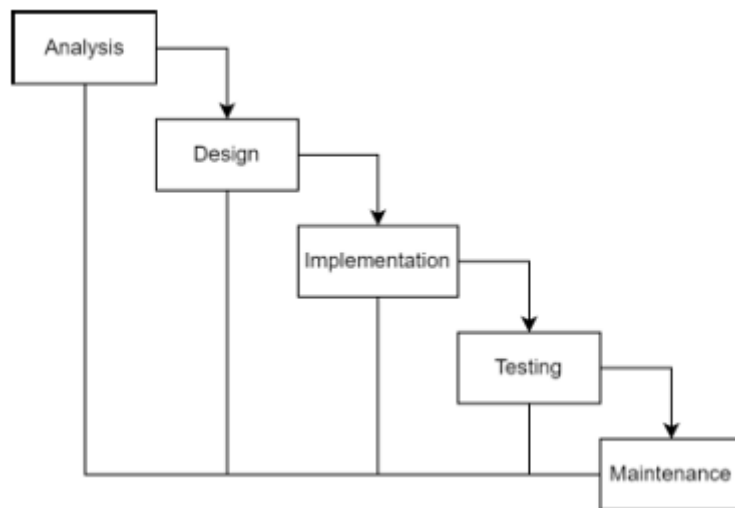
2.4.3 Chatbot Berbasis AI

ChatBot adalah layanan percakapan berbasis kecerdasan buatan (AI) yang dirancang untuk meniru interaksi manusia, memahami, memproses permintaan pengguna, dan memberikan respons yang sesuai dengan cepat. Dalam aplikasi Posyandu, chatbot AI berfungsi untuk memberikan informasi serta menjawab pertanyaan terkait layanan Posyandu, khususnya tentang stunting, dengan kinerjanya diukur berdasarkan akurasi jawaban dan kecepatan respons terhadap setiap permintaan pengguna.

2.4.4 Metode Waterfall

Metode *Waterfall* adalah pendekatan yang digunakan dalam pengembangan sistem dengan proses yang dilakukan secara berurutan, mulai dari tahap analisis kebutuhan, desain, pengkodean, pengujian, hingga implementasi dan pemeliharaan (Santoso & Anggara, 2021). Keberhasilan penerapan metode ini diukur berdasarkan

seberapa baik setiap tahapan dilaksanakan sesuai dengan urutan yang ditetapkan dan apakah setiap tahapan dapat diselesaikan tepat waktu sesuai dengan jadwal yang telah direncanakan.



Gambar 2.2 Metode *Waterfall* (Sumber : (Harry et al., 2024))

Gambar 2.2 menjelaskan langkah –langkah penerapan Metode *Waterfall*, dimana salah satu metodologi pengembangan perangkat lunak yang paling populer dan dianggap mudah untuk diimplementasikan. Metode ini menggambarkan pendekatan linier dan berurutan yang mirip dengan air terjun yang mengalir dari atas ke bawah.

1. *Analysis*

Tahap ini bertujuan untuk mengidentifikasi kebutuhan sistem dari para pemangku kepentingan, seperti kader posyandu, tenaga kesehatan, dan masyarakat. Data dikumpulkan melalui wawancara mendalam dengan kader dan pengguna potensial, serta penyebaran kuesioner untuk memahami kendala yang dihadapi. Observasi juga dilakukan terhadap proses pencatatan manual dan penyampaian informasi di posyandu untuk

memahami alur kerja yang ada. Hasil analisis ini dirangkum dalam dokumen spesifikasi kebutuhan sistem.

2. *Design*

Pada tahap ini, desain sistem dikembangkan meliputi arsitektur aplikasi, antarmuka pengguna, dan basis data. Komponen utama aplikasi, seperti fitur pencatatan data balita dan chatbot berbasis AI, dirancang untuk mendukung efisiensi proses. Desain antarmuka dibuat sederhana dan intuitif agar mudah digunakan oleh kader posyandu dan masyarakat. Skema basis data dirancang untuk menyimpan informasi secara terstruktur dan mendukung akses cepat.

3. *Implementation*

Tahap implementasi mencakup pengembangan kode untuk seluruh komponen aplikasi menggunakan framework *React Native* agar dapat berjalan di perangkat Android dan iOS. Fitur-fitur utama, seperti pencatatan data balita, monitoring tumbuh kembang anak, dan chatbot untuk edukasi kesehatan atau media konseling, diintegrasikan ke dalam sistem untuk memastikan fungsionalitasnya.

4. *Testing*

Setelah implementasi, dilakukan pengujian menyeluruh untuk memastikan semua fitur bekerja dengan baik dan sistem bebas dari kesalahan. Uji coba meliputi pengujian fungsionalitas aplikasi, performa chatbot berbasis AI, keandalan sistem, serta keamanan data pengguna.

5. *Maintenance*

Tahap terakhir adalah pemeliharaan sistem yang mencakup perbaikan bug yang ditemukan selama penggunaan, pengembangan fitur baru sesuai kebutuhan pengguna, dan pemantauan berkala untuk memastikan kinerja aplikasi tetap optimal.

2.4.5 UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah sebuah bahasa visual yang digunakan untuk memodelkan dan mengomunikasikan informasi sistem melalui diagram dan skrip pendukung (Fatahila et al., 2023). UML dirancang sebagai bahasa pemodelan yang tidak bergantung pada bahasa pemrograman tertentu atau proses rekayasa spesifik. Selain itu, UML menyediakan model yang dapat digunakan secara konsisten dan mengadopsi praktik terbaik yang telah ditetapkan dalam pemodelan.

Ada 3 jenis diagram yang bias digunakan dalam praktik yaitu *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram*,

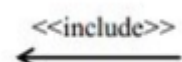
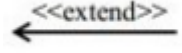
1. *Use Case Diagram*

Use case diagram adalah diagram yang digunakan untuk memodelkan sistem informasi yang akan diimplementasikan. Diagram ini berfungsi untuk menggambarkan bagaimana satu atau lebih aktor berinteraksi dengan sistem informasi dalam menjalankan berbagai fungsi atau proses yang disediakan oleh sistem.

Tabel 2.2 Komponen *Use Case*

Simbol	keterangan
	Actor : mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case

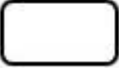
Tabel 2.2 Lanjutan Komponen Use Case

	Use case : abstraksi dan interaksi antara sistem dan actor.
	Association : abstraksi dari penghubung antara aktor dengan use case
	Generalisasi : menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case
	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya
	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainya jika suatu kondisi terpenuhi

2. Activity Diagram

Activity diagram adalah representasi visual yang digunakan untuk menunjukkan rangkaian aktivitas dalam suatu sistem, seperti sebuah program. Proses, keputusan, dan jalur yang dipertimbangkan dalam melaksanakan berbagai aktivitas sistem ditampilkan dalam diagram ini.

Tabel 2.3 Komponen Activity Diagram

Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan	Penggabungan dimana ada lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

3. *Class Diagram*

Class diagram adalah diagram yang digunakan untuk menggambarkan struktur sebuah sistem dengan mendefinisikan komponen utama yang akan dibuat selama pengembangan sistem. Diagram ini menjelaskan karakteristik, metode, dan hubungan antar kelas untuk memberikan pemahaman yang jelas tentang arsitektur sistem.

Class Diagram terdiri dari beberapa komponen utama, yaitu :

- a. Kelas (*Class*) : Kelas merepresentasikan entitas dalam sistem yang memiliki atribut dan metode. Digambarkan sebagai persegi panjang yang terdiri dari nama kelas, atribut, dan metode.
- b. Atribut (*Attribute*) : Atribut adalah properti atau data dalam kelas, memiliki nama, tipe data, dan visibilitas. Contohnya, username: String dalam kelas "User".
- c. Metode (*Methods*) : Metode adalah fungsi atau perilaku yang dapat dilakukan oleh kelas. Misalnya, metode `login()` dalam kelas "User" untuk proses masuk ke sistem.
- d. Visibilitas (*Visibility/Access Modifier*) : Menentukan hak akses atribut atau metode. Public (+) dapat diakses dari luar kelas, Private (-) hanya dalam kelas, dan Protected (#) dalam kelas serta subclass.
- e. Hubungan antar Kelas (*Relationships*) : Hubungan dalam class diagram meliputi **Asosiasi** (hubungan antar kelas), **Agregasi** (hubungan "has-a" yang tidak bergantung), **Komposisi** (hubungan kuat di mana kelas bergantung sepenuhnya pada kelas lain), dan

Pewarisan (kelas mewarisi atribut dan metode dari superclass).

- f. Kardinalitas (*Multiplicity*) : Menunjukkan jumlah objek dalam hubungan antar kelas, seperti 1 (satu objek), 0..1 (nol atau satu objek), dan * (banyak objek).

BAB III METODOLOGI

3.1 Desain Penelitian

Desain penelitian adalah kerangka kerja yang disusun untuk melaksanakan penelitian secara sistematis. Kerangka ini berfungsi sebagai panduan bagi peneliti dalam menyusun langkah-langkah yang diperlukan untuk mengumpulkan, menganalisis, dan menginterpretasikan data guna menjawab permasalahan penelitian atau menguji hipotesis yang ada.

Berikut ini adalah desain penelitian yang diterapkan dalam rancang bangun aplikasi posyandu berbasis mobile menggunakan framework *React Native*, yang dilengkapi dengan fitur tambahan berupa chatbot.

3.1.1 Alur Penelitian

Penelitian ini mengikuti alur metode *waterfall* yang terdiri dari beberapa tahapan, yaitu analisis kebutuhan, perancangan sistem, implementasi, testing, dan maintenance.

3.1.1.1 Analisis Kebutuhan

Tahap ini bertujuan untuk menggali dan memahami kebutuhan sistem dari berbagai pemangku kepentingan, seperti kader posyandu, tenaga kesehatan, dan masyarakat. Pengumpulan data dilakukan melalui wawancara mendalam dengan kader serta calon pengguna guna mengidentifikasi kendala yang dihadapi. Selain itu, observasi terhadap proses pencatatan manual dan penyampaian informasi di posyandu dilakukan untuk memahami alur kerja yang sedang berjalan. Hasil dari analisis ini kemudian disusun dalam dokumen spesifikasi kebutuhan sistem.

3.1.1.2 Perancangan Sistem

Pada tahap ini, desain sistem dikembangkan menggunakan UML (*Unified Modeling Language*) untuk memodelkan arsitektur aplikasi, antarmuka pengguna, serta struktur basis data. Diagram UML, seperti *use case diagram*, *class diagram*, dan *activity diagram*, digunakan untuk menggambarkan interaksi pengguna, alur kerja sistem, serta hubungan antar komponen.

3.1.1.2.1 Use Case Diagram



Gambar 3.1 Use Case Diagram

Berdasarkan gambar use case diagram tersebut, berikut adalah penjelasan untuk setiap *use case* :

1. Login

Semua aktor (Admin, Kader, Bidan, dan Orang Tua) harus login untuk mengakses fitur yang sesuai dengan perannya. Proses ini memastikan keamanan dan akses terbatas berdasarkan hak pengguna.

2. Kelola User

Fitur ini hanya dapat diakses oleh Admin. Admin dapat menambah, mengubah, atau menghapus data pengguna dalam sistem, termasuk kader, bidan, dan orang tua.

3. Kelola data Posyandu

Admin juga memiliki wewenang untuk mengelola data posyandu, seperti menambah jadwal kegiatan posyandu, lokasi, dan data lainnya.

Kader juga memiliki wewenang dalam data posyandu seperti mencatat data balita, seperti berat badan, tinggi badan, dan status imunisasi. Data ini kemudian disimpan di sistem untuk dianalisis atau digunakan oleh aktor lainnya.

4. Memantau Data Balita

Admin dapat memantau seluruh data balita yang terdaftar di sistem, termasuk data kesehatan, imunisasi, dan perkembangan anak. Fitur ini mempermudah admin dalam mengawasi kegiatan posyandu secara keseluruhan.

5. Membuat Laporan

Kader/Petugas memiliki kemampuan untuk membuat laporan berdasarkan data balita yang sudah dimasukkan. Laporan ini digunakan untuk keperluan evaluasi atau pelaporan kepada pihak terkait, seperti dinas kesehatan.

6. Lihat Data Kesehatan Balita

Bidan memiliki akses untuk melihat data kesehatan balita yang terdaftar, seperti berat badan, tinggi badan, dan riwayat imunisasi. Informasi ini membantu bidan dalam memberikan masukan atau penanganan kesehatan lebih lanjut.

7. Konsultasi Chatbot

Orang tua dapat menggunakan fitur chatbot untuk mendapatkan informasi atau berkonsultasi tentang kesehatan anak mereka. Chatbot memberikan jawaban otomatis mengenai jadwal imunisasi, tips kesehatan anak, dan informasi posyandu lainnya.

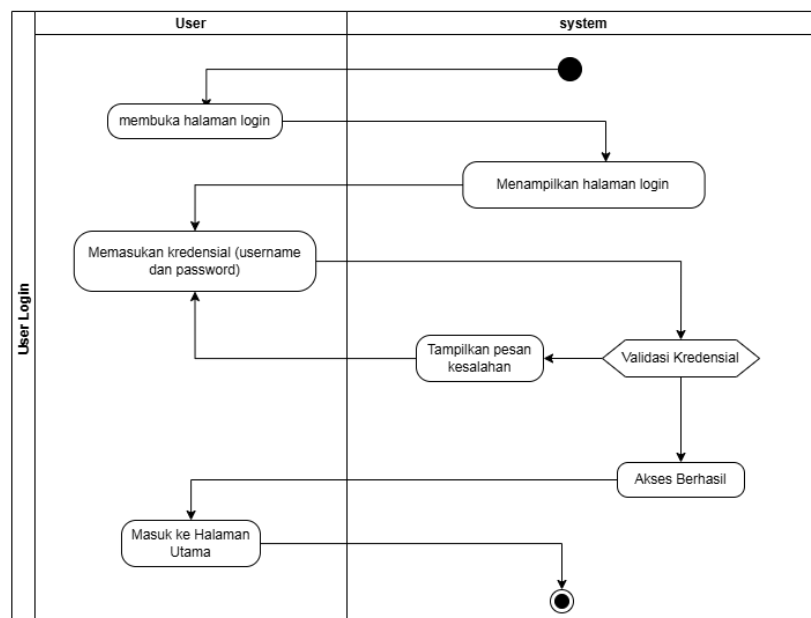
8. Lihat Data Anak dan Jadwal Posyandu

Orang tua dapat mengakses data perkembangan anak mereka, seperti berat badan, tinggi badan, dan jadwal kegiatan posyandu berikutnya. Fitur ini membantu orang tua dalam memastikan anak mereka mendapat pelayanan kesehatan yang sesuai.

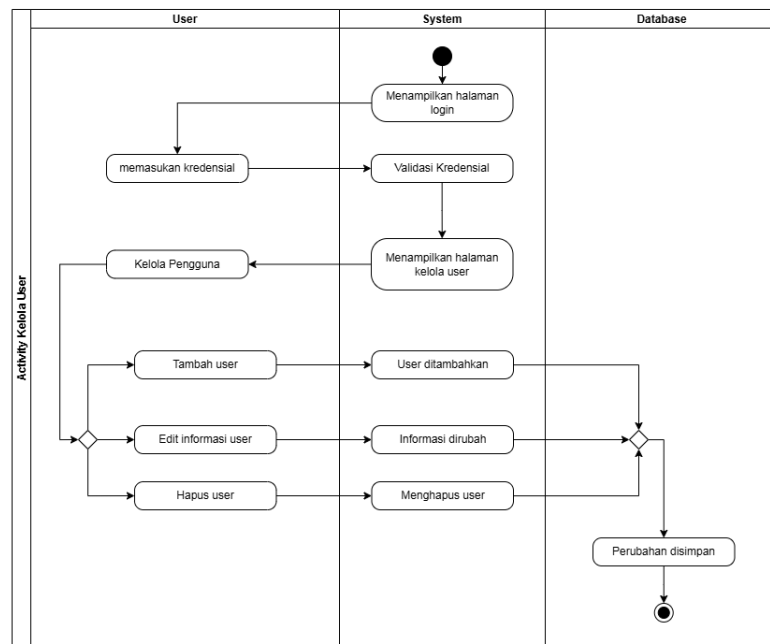
Use case diagram ini memberikan gambaran umum tentang fungsionalitas utama yang harus disediakan oleh sistem, serta interaksi antara pengguna atau aktor dengan sistem tersebut.

3.1.1.2.2 Activity Diagram

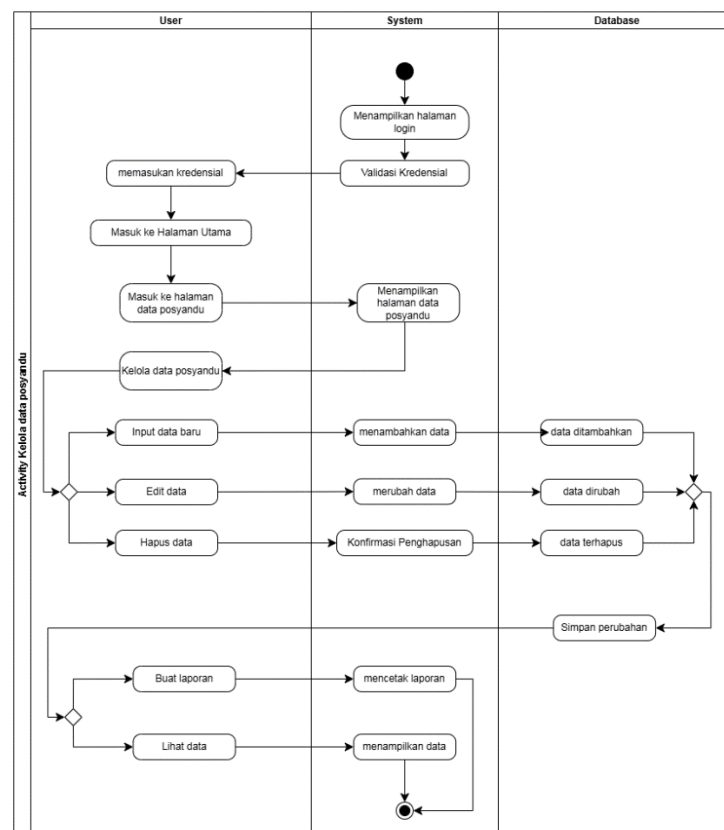
Activity Diagram digunakan untuk menggambarkan alur kerja atau aktivitas dalam aplikasi Posyandu. Diagram ini memvisualisasikan urutan langkah-langkah atau proses yang terjadi dalam aplikasi, seperti pencatatan data balita, pengelolaan informasi kesehatan, hingga interaksi pengguna dengan fitur chatbot berbasis AI untuk mendapatkan edukasi kesehatan.



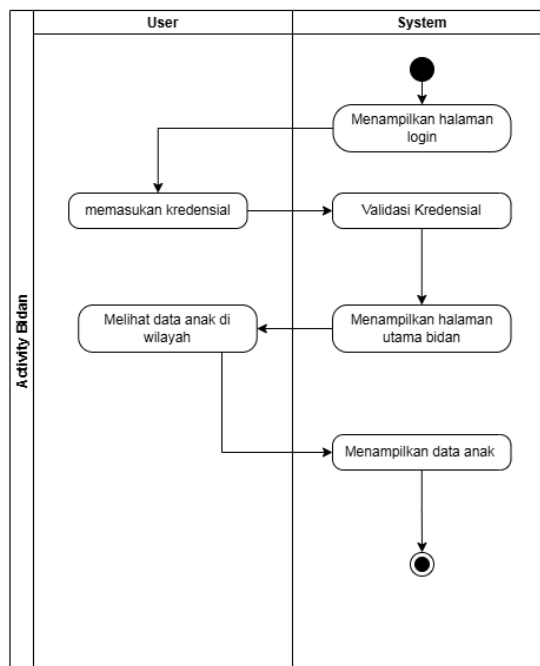
Gambar 3.2 Activity Diagram Login



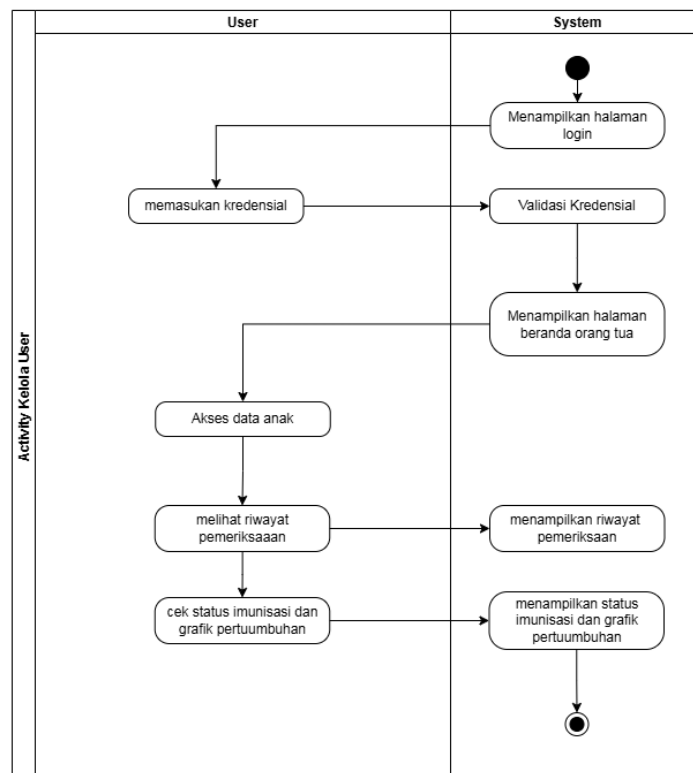
Gambar 3.3 Activity Diagram Mengelola Pengguna/User



Gambar 3.4 Activity Diagram untuk Mengelola Data Posyandu



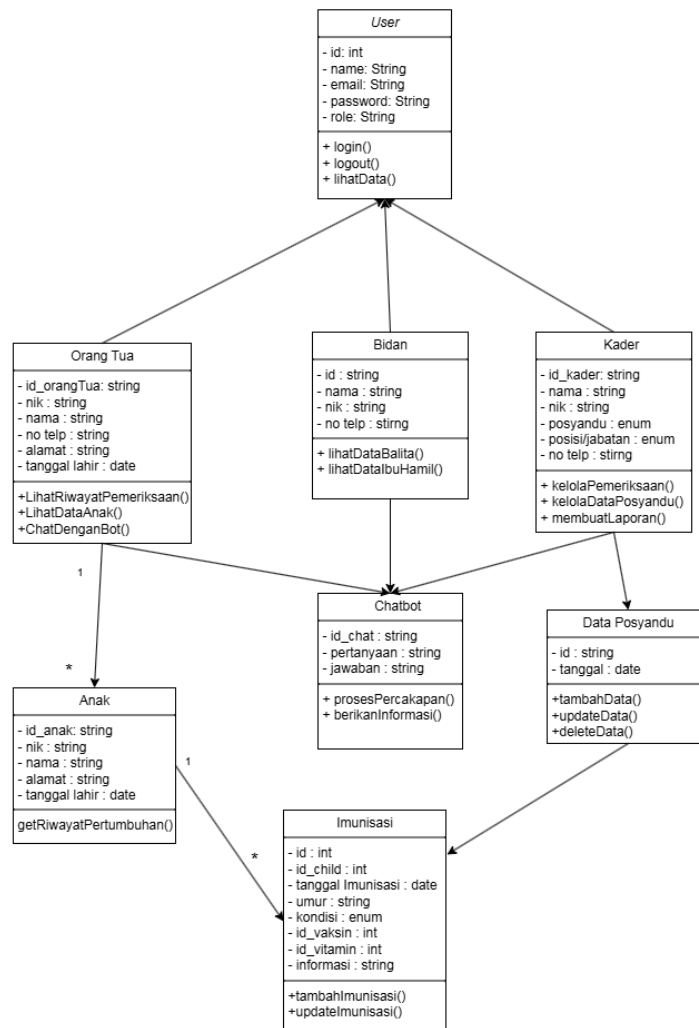
Gambar 3.5 Activity Diagram Bidan untuk Melihat Data Balita



Gambar 3.6 Activity Diagram Orang Tua untuk Melihat Data Anak

3.1.1.2.3 Class Diagram

Class Diagram digunakan untuk menggambarkan struktur objek dalam sistem informasi Posyandu berbasis *mobile*. *Class Diagram* ini memetakan kelas-kelas yang terlibat dalam sistem, atribut yang dimiliki setiap kelas, serta hubungan antar kelas. Diagram ini membantu memvisualisasikan bagaimana data seperti informasi balita, jadwal Posyandu, pengguna (admin, kader, bidan, orang tua), serta fitur chatbot AI diatur dan saling terhubung dalam aplikasi. Berikut diagramnya terlampir pada Gambar 3.7 :



Gambar 3.7 Class Diagram

3.1.1.2.4 Proses Pembuatan Chatbot

Dalam pengembangan chatbot pada sistem informasi Posyandu berbasis *mobile*, digunakan beberapa tahapan pemrosesan teks dan metode pencocokan teks untuk meningkatkan akurasi chatbot dalam memberikan jawaban yang relevan.

1. *Text Processing*

Proses pemrosesan teks dilakukan untuk menyiapkan data sebelum diolah lebih lanjut dalam sistem chatbot. Tahapan ini meliputi:

- a. *Case Folding*: Mengubah seluruh huruf dalam teks menjadi huruf kecil. Hal ini dilakukan agar variasi penggunaan huruf besar atau kecil tidak mempengaruhi analisis, sehingga kata “Posyandu” dan “posyandu” dianggap sama.
- b. *Tokenizing*: Memecah teks menjadi potongan-potongan yang lebih kecil atau token, seperti kata-kata, sehingga setiap unit teks dapat dianalisis secara terpisah. Tokenisasi memungkinkan sistem untuk mengidentifikasi kata kunci dari setiap kalimat.
- c. *Stopword Removal*: Menghapus kata-kata umum yang tidak memberikan informasi penting atau kontekstual, seperti “dan”, “atau”, “di”, serta kata-kata lain yang tidak berkontribusi pada makna inti. Penghilangan *stopword* membantu mempercepat proses pencocokan kata yang lebih relevan.
- d. *Stemming*: Mengembalikan kata-kata ke bentuk dasar atau akar katanya. Misalnya, kata “mencatat”, “pencatatan”, dan

“mencatatkan” disederhanakan menjadi bentuk dasar “catat”. Hal ini bertujuan untuk menyamakan variasi kata sehingga sistem dapat mengenali kesamaan makna.

2. TF-IDF dan VSM

Chatbot menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF) untuk menghitung bobot setiap kata dalam dokumen berdasarkan seberapa sering kata tersebut muncul, serta seberapa unik kata tersebut dalam keseluruhan dataset. Selanjutnya, *Vector Space Model* (VSM) digunakan untuk merepresentasikan dokumen dalam bentuk vektor, sehingga memungkinkan pencocokan pertanyaan dengan jawaban yang paling relevan (Nuzul Hikmah et al., 2022).

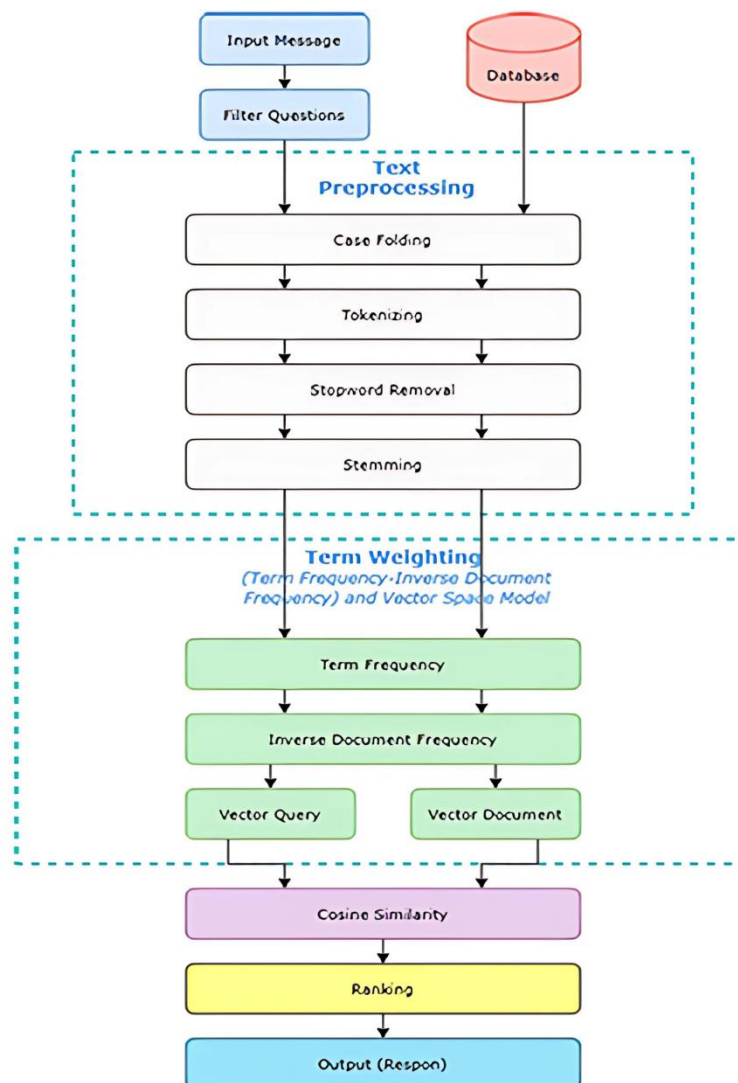
3. Cosine Similarity

Metode *Cosine Similarity* digunakan untuk menghitung tingkat kesamaan antara pertanyaan pengguna dan dokumen yang tersedia dalam database chatbot. Semakin tinggi nilai kemiripan yang diperoleh, semakin besar kemungkinan chatbot memberikan jawaban yang sesuai dengan pertanyaan pengguna.

Gambar 3.8 menggambarkan arsitektur sistem chatbot, di mana pesan yang diterima dari pengguna pertama-tama diolah melalui proses text preprocessing, kemudian dilakukan pembobotan kata menggunakan metode TF-IDF dan VSM untuk menentukan signifikansi setiap kata, dilanjutkan dengan penghitungan tingkat kemiripan antara pesan dan dokumen melalui *cosine similarity*, dan akhirnya hasilnya diranking berdasarkan nilai bobot

tertinggi untuk menentukan respons yang paling relevan. Adapun lingkup pertanyaan yang bias dijawab oleh chatbot adalah :

1. Menjawab pertanyaan terkait informasi jadwal imunisasi, edukasi kesehatan, dan kegiatan Posyandu.
2. Menyediakan informasi tentang layanan kesehatan seperti tips pencegahan stunting, serta saran-saran intervensi dini.



Gambar 3.8 Arsitektur Sistem Chatbot
(Nuzul Hikmah et al., 2022)

3.1.1.3 Implementasi

Pada tahap ini, pengembangan kode dilakukan untuk membangun seluruh komponen aplikasi menggunakan framework *React Native*, sehingga aplikasi dapat berjalan di perangkat Android dan iOS. Fitur utama, seperti pencatatan data balita, pemantauan pertumbuhan anak, serta chatbot untuk edukasi kesehatan dan layanan konseling, diintegrasikan ke dalam sistem guna memastikan fungsionalitasnya berjalan dengan optimal.

3.1.1.4 Testing

Setelah implementasi, dilakukan pengujian menyeluruh menggunakan metode *Black Box Testing* untuk memastikan setiap fitur berfungsi sesuai dengan kebutuhan tanpa memeriksa kode sumber. Pengujian ini mencakup verifikasi fungsionalitas aplikasi, evaluasi performa chatbot berbasis AI, serta uji keandalan sistem dan keamanan data pengguna. Fokus utama pengujian adalah memastikan bahwa setiap input menghasilkan output yang sesuai dengan ekspektasi, serta mengidentifikasi dan memperbaiki potensi kesalahan dalam sistem.

3.1.1.5 Maintenance

Tahap akhir ini mencakup perbaikan bug yang ditemukan selama penggunaan, pengembangan fitur baru berdasarkan kebutuhan pengguna, serta pemantauan rutin untuk memastikan aplikasi tetap berjalan dengan optimal. Pemeliharaan dilakukan secara berkala guna menjaga kinerja sistem dan meningkatkan pengalaman pengguna.

3.1.2 Tempat Penelitian

Tempat penelitian dilakukan di Desa Jabung Sisir, yang memiliki aktivitas Posyandu secara rutin. Penelitian ini berfokus pada penerapan aplikasi Posyandu

untuk membantu kegiatan pencatatan dan penyampaian informasi kepada petugas Posyandu dan masyarakat di desa tersebut.

3.1.3 Waktu dan Jadwal Penelitian

Penelitian direncanakan berlangsung selama enam bulan, dengan rincian sebagai berikut:

1. Bulan 1: Analisis kebutuhan pengguna dan pengumpulan data awal.
2. Bulan 2-3: Perancangan sistem menggunakan UML.
3. Bulan 4: Implementasi menggunakan *react native* dan Integrasi fitur chatbot berbasis AI dalam aplikasi.
4. Bulan 5: Pengujian aplikasi menggunakan *blackbox testing*.
5. Bulan 6: *Maintenance* pada aplikasi.

3.1.4 Bahan Penelitian

Bahan yang digunakan dalam penelitian ini meliputi:

1. Data kebutuhan pengguna terkait kegiatan Posyandu di Desa Jabung Sisir.
2. Referensi literatur tentang pengembangan aplikasi *mobile* berbasis *React Native*, chatbot berbasis AI, dan sistem informasi kesehatan.
3. Dokumentasi manual pencatatan Posyandu yang digunakan saat ini sebagai acuan dalam pengembangan fitur aplikasi.

3.1.5 Alat Penelitian

Alat yang digunakan dalam penelitian ini mencakup:

1. Komputer atau laptop untuk pengembangan aplikasi menggunakan *React Native*.
2. Perangkat Android dan iOS sebagai media untuk menguji dan menerapkan aplikasi Posyandu.

3. Perangkat lunak pengembangan seperti Visual Studio Code, Android Studio, dan pustaka *React Native* yang relevan.

3.1.6 Tatalaksana Penelitian

Tatalaksana penelitian ini terdiri dari:

1. Tahap Analisis Kebutuhan: Pengumpulan data kebutuhan pengguna dari petugas Posyandu dan masyarakat Desa Jabung Sisir.
2. Tahap Perancangan: Membuat desain menggunakan UML (*Unified Modeling Language*) untuk memodelkan arsitektur aplikasi, antarmuka pengguna, serta struktur basis data. Diagram UML, seperti *use case diagram*, *class diagram*, dan *activity diagram*.
3. Tahap Implementasi dan Integrasi Chatbot: Implementasi *design system* menggunakan *React Native* dan menambahkan fitur chatbot berbasis AI yang dapat memberikan informasi kesehatan secara interaktif.
4. Tahap Pengujian: Melakukan pengujian untuk memastikan aplikasi berfungsi dengan baik di perangkat Android dan iOS serta mudah digunakan oleh pengguna.
5. Tahap *Maintenance*: Dilakukan setelah aplikasi diuji menggunakan *blackbox testing*. Selain itu, pada tahap ini juga dapat dilakukan pembaruan atau pengembangan fitur tambahan sesuai kebutuhan dan masukan dari pengguna demi menjaga performa aplikasi tetap optimal.

3.2 Variable Penelitian

Berikut variable penelitian terkait perancangan sistem informasi posyandu berbasis mobile menggunakan *framework react native* dengan fitur chatbot berbasis ai.

3.2.1 Variable Bebas

Fitur chatbot berbasis AI yang diintegrasikan dalam aplikasi mobile berfungsi sebagai media konseling untuk memberikan berbagai informasi, seperti kesehatan anak, jadwal kegiatan Posyandu, dan edukasi tentang stunting. Keberadaan fitur ini bertujuan untuk meningkatkan kemudahan pengguna dalam mengakses informasi kesehatan secara cepat dan efisien, sehingga membantu memenuhi kebutuhan informasi dengan lebih praktis dibandingkan metode konvensional.

3.2.2 Variabel Terkait

Efektivitas aplikasi diukur menggunakan beberapa indikator, seperti peningkatan pemahaman pengguna mengenai kesehatan anak dan layanan Posyandu, pengurangan waktu yang dibutuhkan untuk mengakses informasi dibandingkan metode manual, serta tingkat kepuasan pengguna dalam memanfaatkan aplikasi sebagai sumber informasi. Salah satu ukuran yang digunakan adalah persentase peningkatan kepuasan pengguna setelah menggunakan aplikasi dibandingkan dengan metode manual sebelumnya.

3.2.3 Variabel Kontrol

Tingkat kompetensi digital pengguna, termasuk kemampuan kader, admin, atau orang tua dalam mengoperasikan perangkat mobile dan memahami aplikasi, dijaga tetap seragam. Pendekatan ini bertujuan untuk memastikan bahwa efektivitas aplikasi dipengaruhi sepenuhnya oleh fitur chatbot yang disediakan, sehingga mencegah bias dalam hasil penelitian akibat perbedaan literasi digital di antara pengguna.

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Pengumpulan Data

Pada tahap ini, data dikumpulkan untuk mendukung proses perancangan dan implementasi aplikasi Posyandu berbasis mobile menggunakan *framework React Native* dengan fitur chatbot berbasis AI.

Pengumpulan data dilakukan melalui beberapa metode, yaitu:

1. Wawancara

Dilakukan dengan kader Posyandu di Desa Jabung Sisir untuk menggali informasi kebutuhan sistem, permasalahan dalam pencatatan manual, serta ekspektasi terhadap aplikasi yang dikembangkan.

2. Observasi

Observasi langsung terhadap aktivitas pencatatan data balita, penyampaian informasi kesehatan, dan proses laporan kegiatan posyandu, untuk memahami alur kerja yang berjalan secara manual.

3. Studi Dokumentasi

Mengkaji dokumen-dokumen seperti buku register Posyandu, dan laporan kegiatan Posyandu untuk mengetahui format data yang digunakan dan informasi yang diperlukan dalam sistem digital.

Data yang diperoleh dari metode-metode tersebut menjadi dasar dalam merancang fitur-fitur aplikasi, seperti pencatatan data balita secara digital, tampilan jadwal Posyandu, dan fitur chatbot untuk edukasi kesehatan.

4.2 Pengolahan Data

Data yang telah dikumpulkan kemudian diolah untuk menghasilkan spesifikasi kebutuhan sistem yang mendukung pengembangan aplikasi. Langkah-langkah pengolahan data dilakukan sebagai berikut:

4.2.1 Penyajian Data

Data yang diperoleh dari wawancara, observasi, dan kuesioner disajikan dalam bentuk tabel sederhana dan uraian teks yang sistematis. Penyajian ini bertujuan untuk menggambarkan kondisi eksisting, kebutuhan sistem, serta ekspektasi pengguna terhadap aplikasi yang dikembangkan.

Berikut adalah ringkasan data yang diperoleh :

Tabel 4.1 Ringkasan Data

No	Kategori Data	Temuan Lapangan
1	System Pencatatan Saat Ini	Pencatatan data balita masih dilakukan secara manual menggunakan buku tulis.
2	Permasalahan Pencatatan Manual	Data sering hilang, rawan kesalahan input, dan sulit diakses saat dibutuhkan.
3	Kebutuhan Utama yang Diinginkan	Digitalisasi data, akses informasi lebih cepat, dan konseling kesehatan berbasis teknologi.
4	Harapan terhadap Aplikasi	Aplikasi yang mudah digunakan, responsif, serta menyediakan fitur chatbot untuk konsultasi kapan saja.

Data ini digunakan untuk mendefinisikan kebutuhan sistem dan menjadi dasar dalam perancangan aplikasi.

4.2.2 Analisi Kebutuhan

Berdasarkan hasil pengolahan data, disusunlah kebutuhan fungsional dan non-fungsional aplikasi, meliputi :

1. Kebutuhan Fungsional :

- a. Pencatatan data balita secara digital.
- b. Pengelolaan jadwal Posyandu.

c. Konsultasi kesehatan melalui chatbot AI.

d. Akses data riwayat kesehatan balita.

2. Kebutuhan Non-Fungsional :

a. Sistem mudah digunakan (*user friendly*).

b. Aplikasi responsif dan aman dalam pengelolaan data pengguna.

4.2.3 Data untuk Implementasi Fitur Penimbangan

Fitur penimbangan pada aplikasi ini berfungsi untuk memantau status gizi dan pertumbuhan balita secara digital dan otomatis. Data masukan yang dibutuhkan dalam fitur ini meliputi berat badan (dalam kilogram), tinggi badan (dalam sentimeter), usia anak (dalam bulan), dan jenis kelamin. Dengan data tersebut, sistem menghitung Indeks Massa Tubuh (IMT) atau Body Mass Index (BMI) menggunakan rumus:

$$\text{BMI} = \frac{\text{berat badan (kg)}}{(\text{tinggi badan (m)})^2}$$

Setelah nilai BMI diperoleh, sistem menentukan status gizi berdasarkan kategori yang disederhanakan sesuai dengan ambang batas yang digunakan dalam praktik Posyandu dan pendekatan penelitian terkait. Klasifikasi ini mengacu pada penelitian oleh Wati & Istianah (2023), yang menggunakan BMI sebagai indikator penilaian gizi balita, serta pada Permenkes RI No. 2 Tahun 2020 (RI, 2020) yang menetapkan pendekatan serupa berbasis IMT/U.

Tabel 4.2 Kategori Status Gizi Berdasarkan Nilai BMI

No	Kategori Gizi	Rentang BMI (kg/m ²)	Penjelasan	Tindakan yang Disarankan
1	Gizi Buruk	< 14	Sangat kurus	Rujuk ke tenaga medis, pemberian gizi intensif

Tabel 4.2 Lanjutan

No	Kategori Gizi	Rentang BMI (kg/m ²)	Penjelasan	Tindakan yang Disarankan
2	Gizi Kurang	14 – <16	Kurus	Tambahan makanan, edukasi orang tua
3	Normal/Baik	16 – 25	Ideal/proposional	Lanjutkan pola asuh sehat
4	Gizi Lebih / Obesitas	> 25	Berat badan berlebih	Konseling gizi, aktivitas fisik teratur

Selain status gizi berdasarkan BMI, sistem juga melakukan pengukuran status pertumbuhan balita dengan membandingkan tinggi badan aktual anak terhadap tinggi badan ideal sesuai usia dan jenis kelamin. Tinggi ideal dihitung menggunakan rumus pertumbuhan WHO dengan asumsi peningkatan tinggi setiap bulan, disesuaikan untuk anak laki-laki dan perempuan. Setelah itu, sistem menghitung persentase tinggi terhadap tinggi ideal:

$$\text{Persentase Tinggi} = \frac{\text{Tinggi aktual}}{\text{Tinggi ideal}} \times 100\%$$

Berdasarkan persentase tersebut, status pertumbuhan dibedakan menjadi empat kategori sebagaimana ditunjukkan pada Tabel 4.2. Pendekatan ini sejalan dengan klasifikasi Z-score TB/U (tinggi badan menurut umur) seperti yang digunakan dalam penelitian oleh Utami et al., (2023) serta Fajar et al., (2022) yang juga mengukur status gizi berdasarkan nilai Z-score BB/U dan TB/U.

Table 4.3 Kategori Status Pertumbuhan Berdasarkan Tinggi Badan

No	Kategori Pertumbuhan	Kriteria	Penjelasan
1	Stunting	Tinggi < 85% dari tinggi ideal	Tinggi anak jauh lebih pendek dari standar usianya
2	Wasting	Tinggi 85–90% & BMI < 16	Berat badan rendah untuk tinggi badannya
3	Obesitas	BMI > 25	Berat sangat berlebih, risiko obesitas
4	Normal	Tidak termasuk kategori di atas	Tinggi dan berat badan sesuai dengan standar usia

Seluruh logika klasifikasi status gizi dan pertumbuhan ini diimplementasikan dalam fungsi Penimbangan pada sistem backend aplikasi. Pendekatan ini dipilih karena sederhana, efisien, dan sudah terbukti digunakan dalam berbagai studi serta pedoman resmi pemerintah. Melalui fitur ini, pengguna aplikasi seperti kader, bidan, dan orang tua dapat memantau kondisi balita dengan cepat dan akurat.

4.2.4 Pemrosesan Data Chatbot

4.2.4.1 Text Processing

Text preprocessing adalah langkah awal dalam pengolahan data teks untuk memastikan bahwa data dalam bentuk yang bersih dan dapat digunakan dalam tahap pemrosesan lanjutan seperti pembobotan dan perhitungan kesamaan. Proses ini diterapkan pada data pertanyaan-pertanyaan yang diajukan oleh pengguna dan dataset jawaban yang telah dikategorikan. Tujuannya adalah untuk mempersiapkan dokumen teks agar bisa diproses lebih lanjut dengan metode TF-IDF dan *Cosine Similarity*.

Adapun tahapan dalam *text preprocessing* adalah sebagai berikut:

1. Case Folding

Tahapan pertama adalah mengubah seluruh huruf dalam teks menjadi huruf kecil (lowercase). Proses ini bertujuan untuk menghilangkan perbedaan antara huruf besar dan kecil sehingga penyamaan kata dapat dilakukan secara konsisten.

Contoh:

Sebelum Case Folding	Sesudah Case Folding
Apa itu posyandu?	apa itu posyandu?

```
def case_folding(self, text):
    """
    Tahap 1: Case Folding - mengubah teks menjadi huruf kecil
    """
    return text.lower()
```

Gambar 4.1 Gambar code untuk proses *Case Folding*

2. *Tokenizing*

Tahapan ini memisahkan kalimat atau paragraf menjadi token-token atau kata-kata. Umumnya pemisahan dilakukan berdasarkan spasi.

Contoh:

Sebelum Tokenizing	Sesudah Tokenizing
apa itu posyandu?	apa, itu, posyandu

```
def tokenizing(self, text):
    """
    Tahap 2: Tokenizing - memisahkan teks menjadi token/kata
    """
    # Hapus karakter non-alfanumerik kecuali spasi
    text = re.sub(r'^a-zA-Z0-9\s', '', text)
    # Split menjadi token
    tokens = text.split()
    return tokens
```

Gambar 4.2 Gambar code untuk proses *Tokenizing*

3. *Stopword Removal*

Stopword removal dilakukan untuk menghapus kata-kata umum yang tidak memiliki makna penting dalam analisis, seperti “itu”, “dan”, “di”, “ke”, dll.

Contoh :

Sebelum Stopword Removal	Sesudah Stopword Removal
apa itu posyandu	apa, posyandu

```
def stopwords_removal(self, tokens):
    """
    Tahap 3: Stopword Removal - menghapus kata-kata yang tidak bermakna
    """
    # Gabungkan token menjadi string untuk diproses
    text = ' '.join(tokens)
    # Hapus stopwords
    text = self.stopword_remover.remove(text)
    # Kembalikan ke bentuk token
    return text.split()
```

Gambar 4.3 Gambar code untuk proses *Stopword Removal*

4. Stemming

Stemming adalah proses mengembalikan kata ke bentuk dasarnya. Misalnya, kata "penimbangan", "penimbanganan", atau "menimbang" akan dikembalikan menjadi "timbang".

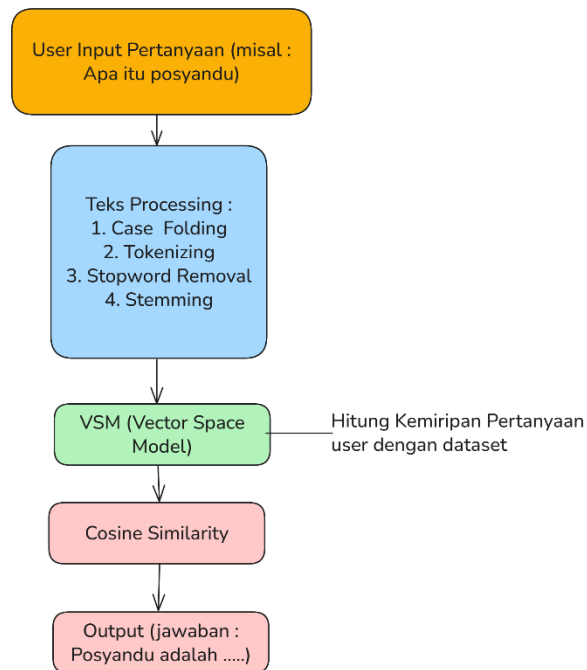
Contoh:

Sebelum Stemming	Sesudah Stemming
penimbangan anak balita	timbang anak balita

```
def stemming(self, tokens):
    """
    Tahap 4: Stemming - mengubah kata ke bentuk dasarnya
    """
    # Gabungkan token menjadi string untuk diproses
    text = ' '.join(tokens)
    # Lakukan stemming
    text = self.stemmer.stem(text)
    # Kembalikan ke bentuk token
    return text.split()
```

Gambar 4.4 Gambar code untuk proses *Stemming*

Setelah melalui tahapan *text preprocessing* seperti *case folding*, *tokenizing*, *stopword removal*, dan *stemming*, chatbot akan memproses teks lebih lanjut menggunakan metode pembobotan dan pencocokan kesamaan. Untuk memperjelas tahapan ini, berikut disajikan diagram alur kerja chatbot dalam memproses pertanyaan dari pengguna:



Gambar 4.5 Alur Kerja Chatbot Posyandu

Gambar tersebut menjelaskan proses kerja chatbot dimulai dari input pertanyaan pengguna yang kemudian diproses melalui beberapa tahap. Tahap pertama adalah *text preprocessing*, yang berfungsi membersihkan teks dari elemen-elemen yang tidak penting dan mengubahnya ke bentuk dasar. Setelah teks dibersihkan, sistem melakukan pembobotan kata menggunakan metode **TF-IDF** untuk menentukan seberapa penting suatu kata dalam dokumen.

Selanjutnya, chatbot membentuk representasi vektor dari teks menggunakan pendekatan *Vector Space Model* (VSM). Proses ini memungkinkan sistem untuk menghitung tingkat kemiripan antara pertanyaan pengguna dengan dataset jawaban yang tersedia. Penghitungan kemiripan dilakukan dengan metode *Cosine Similarity*, dan hasil dengan nilai kemiripan tertinggi akan dipilih sebagai jawaban yang ditampilkan oleh chatbot.

Dengan tahapan ini, chatbot dapat memberikan respons yang relevan meskipun kalimat yang digunakan oleh pengguna tidak persis sama dengan yang ada di database, asalkan memiliki konteks atau kata kunci yang serupa.

4.2.4.2 TF-IDF dan VSM

TF-IDF (*Term Frequency-Inverse Document Frequency*) merupakan salah satu metode pembobotan kata yang digunakan dalam sistem chatbot ini. TF-IDF menghitung bobot suatu kata dengan memperhatikan frekuensi kemunculan kata tersebut dalam sebuah dokumen dan membandingkannya dengan jumlah dokumen secara keseluruhan. Semakin sering kata muncul dalam dokumen tertentu dan semakin jarang kata tersebut muncul di dokumen lain, maka bobotnya akan semakin tinggi (Nuzul Hikmah et al., 2022).

TF-IDF merupakan teknik pembobotan kata yang dilakukan melalui beberapa tahapan preprocessing seperti *tokenizing* (tokenisasi), penghapusan stopwords, dan stemming. Frekuensi kemunculan kata dalam dokumen dianggap sebagai indikator pentingnya kata tersebut dalam konteks dokumen tersebut. Hasil akhir dari pembobotan ini adalah bobot numerik yang menunjukkan seberapa penting sebuah kata dalam dokumen, yang kemudian digunakan untuk mencocokkan pertanyaan pengguna dengan jawaban yang paling relevan dari dataset .

1. (TF) *Term Frequency*

Term Frequency (TF) adalah ukuran yang menunjukkan seberapa sering suatu kata (*term*) muncul dalam sebuah dokumen. Semakin sering kata tersebut muncul, semakin besar nilai TF-nya, yang menunjukkan bahwa kata tersebut memiliki peran penting dalam dokumen tersebut.

Rumus umum TF:

$$tf = tf_{ij}$$

Keterangan:

- a. tf : nilai term frequency
- b. $tf_{(ij)}$: jumlah kemunculan kata i dalam dokumen j

Dengan demikian, TF membantu mengukur seberapa penting suatu kata di dalam satu dokumen tanpa mempertimbangkan kemunculannya di dokumen lain.

Dalam contoh ini, penulis asumsikan semua dokumen sudah melewati tahapan *preprocessing*. Berikut contohnya:

- a. d1 = posyandu pos layanan terpadu pusat giat masyarakat upaya layanan sehat keluarga rencana kelola sama masyarakat daya masyarakat beri mudah masyarakat oleh layanan sehat dasar.
- b. d2 = posyandu pos layanan terpadu beri layanan sehat dasar ibu anak lansia tingkat masyarakat bantu kader latih.
- c. d3 = posyandu wahana daya masyarakat beri layanan sehat dasar liput sehat ibu anak keluarga rencana imunisasi gizi cegah sakit nular
- d. Kata unik : posyandu, pos, layanan, terpadu, pusat, giat, masyarakat, upaya, sehat, keluarga, rencana, kelola, sama, daya, beri, mudah, oleh, dasar, ibu, anak, lansia, tingkat, bantu, kader, latih, wahana, liput, imunisasi, gizi, cegah, sakit, nular.

Dalam perhitungan ini kita akan menggunakan:

$$tf = \log(1 + tf_{ij})$$

Contoh :

Tabel 4.4 Perhitungan (TF) Term Frequency

Term	d1	d2	d3	tf = $\log(1+tf_{ij})$		
				d1	d2	d3
posyandu	1	1	1	0.301	0.301	0.301
pos	1	1	0	0.301	0.301	0
layan	3	2	1	0.602	0.477	0.301
terpadu	1	1	0	0.301	0.301	0
pusat	1	0	0	0.301	0	0
giat	1	0	0	0.301	0	0
masyarakat	4	1	1	0.698	0.301	0.301
upaya	1	0	0	0.301	0	0
sehat	2	1	2	0.477	0.301	0.477
keluarga	1	0	1	0.301	0	0.301
rencana	1	0	1	0.301	0	0.301
kelola	1	0	0	0.301	0	0
sama	1	0	0	0.301	0	0
daya	1	0	1	0.301	0	0.301
beri	1	1	1	0.301	0.301	0.301
mudah	1	0	0	0.301	0	0
oleh	1	0	0	0.301	0	0
dasar	1	1	1	0.301	0.301	0.301
ibu	0	1	1	0	0.301	0.301
anak	0	1	1	0.000	0.301	0.301
lansia	0	1	0	0	0.301	0
tingkat	0	1	0	0	0.301	0
bantu	0	1	0	0	0.301	0
kader	0	1	0	0	0.301	0
latih	0	1	0	0	0.301	0
wahana	0	0	1	0	0	0.301
liput	0	0	1	0	0	0.301
imunisasi	0	0	1	0	0	0.301
gizi	0	0	1	0	0	0.301
cegah	0	0	1	0	0	0.301
sakit	0	0	1	0	0	0.301
nular	0	0	1	0	0	0.301

2. IDF (Inverse Document Frequency)

IDF (Inverse Document Frequency) merupakan perhitungan dari jumlah seluruh dokumen (D) dibagi dengan Frekuensi Dokumen (Dft) dari term (kata) t . Jika suatu istilah sering muncul, itu tidak dianggap sebagai istilah dokumen. Sebaliknya, jika istilah tersebut jarang muncul dalam korpus, maka istilah tersebut dikatakan terkait dengan dokumen.

IDF (Inverse Document Frequency) dapat dinyatakan sebagai berikut:

$$idf_t = \log\left(\frac{N}{D_{ft}}\right)$$

Keterangan:

- a. Idf_t = inverse document frequency
- b. N = Jumlah dokumen
- c. D_{ft} = Jumlah kemunculan term t terhadap dokumen

Contoh :

Tabel 4.5 Perhitungan *IDF (Inverse Document Frequency)*

Term	tf			idf	
	d1	d2	d3	D_{ft}	$\text{Log}(N/D_{ft})$
posyandu	0.301	0.301	0.301	3	0.000
pos	0.301	0.301	0	2	0.176
layan	0.602	0.477	0.301	3	0.000
terpadu	0.301	0.301	0	2	0.176
pusat	0.301	0	0	1	0.477
giat	0.301	0	0	1	0.477
masyarakat	0.698	0.301	0.301	3	0.000
upaya	0.301	0	0	1	0.477
sehat	0.477	0.301	0.477	3	0.000
keluarga	0.301	0	0.301	2	0.176
rencana	0.301	0	0.301	2	0.176
kelola	0.301	0	0	1	0.477
sama	0.301	0	0	1	0.477
daya	0.301	0	0.301	2	0.176
beri	0.301	0.301	0.301	3	0.000
mudah	0.301	0	0	1	0.477
oleh	0.301	0	0	1	0.477
dasar	0.301	0.301	0.301	3	0.000
ibu	0	0.301	0.301	2	0.176
anak	0.000	0.301	0.301	2	0.176
lansia	0	0.301	0	1	0.477
tingkat	0	0.301	0	1	0.477
bantu	0	0.301	0	1	0.477
kader	0	0.301	0	1	0.477
latih	0	0.301	0	1	0.477
wahana	0	0	0.301	1	0.477
liput	0	0	0.301	1	0.477
imunisasi	0	0	0.301	1	0.477
gizi	0	0	0.301	1	0.477

Tabel 4.5 Lanjutan

Term	tf			idf	
	d1	d2	d3	D_{ft}	$\text{Log}(N/D_{ft})$
cegah	0	0	0.301	1	0.477
sakit	0	0	0.301	1	0.477
nular	0	0	1	1	0.477

3. Term Weighting

Term Weighting adalah bobot akhir suatu kata (*term*) dalam dokumen, dihitung berdasarkan:

$$\text{Term Weight}_{ij} = TF_{ij} \times IDF_i$$

Berikut contoh perhitungannya:

Table 4.6 Perhitungan Term Weighting

Term	tf			IDF	W_{ij}		
	d1	d2	d3		d1	d2	d3
posyandu	0.301	0.301	0.301	0.000	0	0	0
pos	0.301	0.301	0	0.176	0.053	0.053	0
layan	0.602	0.477	0.301	0.000	0	0	0
terpadu	0.301	0.301	0	0.176	0.053	0.053	0
pusat	0.301	0	0	0.477	0.144	0	0
giat	0.301	0	0	0.477	0.144	0	0
masyarakat	0.698	0.301	0.301	0.000	0	0	0
upaya	0.301	0	0	0.477	0.144	0	0
sehat	0.477	0.301	0.477	0.000	0	0	0
keluarga	0.301	0	0.301	0.176	0.053	0	0.053
rencana	0.301	0	0.301	0.176	0.053	0	0.053
kelola	0.301	0	0	0.477	0.144	0	0
sama	0.301	0	0	0.477	0.144	0	0
daya	0.301	0	0.301	0.176	0.053	0	0.053
beri	0.301	0.301	0.301	0.000	0	0	0
mudah	0.301	0	0	0.477	0.144	0	0
oleh	0.301	0	0	0.477	0.144	0	0
dasar	0.301	0.301	0.301	0.000	0	0	0
ibu	0	0.301	0.301	0.176	0	0.053	0.053
anak	0	0.301	0.301	0.176	0	0.053	0.053
lansia	0	0.301	0	0.477	0	0.144	0
tingkat	0	0.301	0	0.477	0	0.144	0
bantu	0	0.301	0	0.477	0	0.144	0
kader	0	0.301	0	0.477	0	0.144	0
latih	0	0.301	0	0.477	0	0.144	0
wahana	0	0	0.301	0.477	0	0.000	0.144
liput	0	0	0.301	0.477	0	0	0.144
imunisasi	0	0	0.301	0.477	0	0	0.144

Table 4.6 Lanjutan

Term	tf			IDF	W _{ij}		
	d1	d2	d3		d1	d2	d3
gizi	0	0	0.301	0.477	0	0	0.144
cegah	0	0	0.301	0.477	0	0	0.144
sakit	0	0	0.301	0.477	0	0	0.144
nular	0	0	0.301	0.477	0	0	0.144

```

def build_tfidf_matrix(self):
    """
    Tahap 5: Membangun matriks TF-IDF dari dataset pertanyaan
    """
    try:
        # Preprocessing semua pertanyaan
        processed_questions = [self.text_preprocessing(q) for q in self.questions]

        # Filter pertanyaan kosong
        processed_questions = [q for q in processed_questions if q.strip()]

        if not processed_questions:
            raise ValueError("Tidak ada pertanyaan yang valid untuk diproses")

        # Inisialisasi TF-IDF Vectorizer
        self.vectorizer = TfidfVectorizer(
            max_features=5000, # Maksimum fitur yang digunakan
            ngram_range=(1, 2), # Unigram dan bigram
            min_df=1, # Minimum document frequency
            max_df=0.8 # Maximum document frequency
        )

        # Fit dan transform pertanyaan menjadi matriks TF-IDF
        self.tfidf_matrix = self.vectorizer.fit_transform(processed_questions)

        logger.info(f"Matriks TF-IDF berhasil dibuat dengan shape: {self.tfidf_matrix.shape}")
    except Exception as e:
        logger.error(f"Error saat membangun matriks TF-IDF: {str(e)}")
        raise

```

Gambar 4.6 Gambar code TF-IDF

Setelah dilakukan pembobotan kata menggunakan metode TF-IDF, setiap dokumen direpresentasikan dalam bentuk vektor berdasarkan bobot term masing-masing. Representasi vektor dokumen diperlukan untuk menghitung kemiripan antar dokumen dengan pertanyaan pengguna (*query*).

Sebagai contoh, berdasarkan perhitungan sebelumnya, kita bisa menyusun vektor dokumen sebagai berikut (nilai disederhanakan untuk keperluan ilustrasi):

- a. $d1 = [0.053, 0.144, 0.053, 0.144, 0.053, 0.053, 0.144, 0.053]$
- b. $d2 = [0.053, 0.053, 0.053, 0.144, 0.144, 0.144, 0.053, 0.053]$

$$c. \quad d3 = [0.053, 0.053, 0.144, 0.144, 0.053, 0.053, 0.144, 0.144]$$

Setiap elemen dalam vektor merepresentasikan bobot suatu term tertentu berdasarkan urutan term yang sama pada seluruh dokumen.

4.2.4.3 Cosine Similarity

Cosine similarity adalah metode yang digunakan untuk mengukur seberapa mirip dua teks. Metode ini memungkinkan sistem untuk memberikan peringkat pada dokumen berdasarkan tingkat relevansinya terhadap suatu *query*. Pengukuran dilakukan dengan menghitung nilai sudut kosinus antara dua vektor, yaitu vektor *query* dan vektor dokumen, guna menentukan tingkat kemiripan di antara keduanya (Hikmah, 2018). Nilai similarity berkisar antara 0 (tidak mirip) hingga 1 (sangat mirip).

$$\text{cosine - similarity} = \frac{A \cdot B}{||A|| \times ||B||}$$

Dimana:

1. A dan B adalah vektor dokumen dan vektor *query*
2. $\cdot$ adalah operasi dot product
3. $||A||$ dan $||B||$ adalah panjang (magnitudo) dari vector

Jika pengguna mengetikkan pertanyaan, sistem akan menghitung *Cosine Similarity* antara *query* tersebut dan seluruh dokumen. Sistem kemudian akan memilih dokumen dengan nilai similarity tertinggi sebagai sumber jawaban yang dianggap paling relevan.

Berikut implementasi *cosine similarity*:

Tabel 4.7 Implementasi *cosine similarity*

Query	Apakah posyandu memberikan layanan imunisasi
Case folding	apakah posyandu memberikan layanan imunisasi

Tabel 4.7 Lanjutan

Tokenizing	['apakah', 'posyandu', 'memberikan', 'layanan', 'imunisasi']
Stopword Removal	['posyandu', 'memberikan', 'layanan', 'imunisasi']
Stemming	['posyandu', 'beri', 'layan', 'imunisasi']
Term Frequency	posyandu : 1, pos : 0, layan : 1, terpadu : 0, pusat : 0, giat : 0, masyarakat : 0, upaya : 0, sehat : 0, keluarga : 0, rencana : 0, kelola : 0, sama : 0, daya : 0, beri : 1, mudah : 0, oleh : 0, dasar : 0, ibu : 0, anak : 0, lansia : 0, tingkat : 0, bantu : 0, kader : 0, latih : 0, wahana : 0, liput : 0, imunisasi : 1, gizi : 0, cegah : 0, sakit : 0, nular : 0.
Term Weighting	posyandu : 0, pos : 0, layan : 0, terpadu : 0, pusat : 0, giat : 0, masyarakat : 0, upaya : 0, sehat : 0, keluarga : 0, rencana : 0, kelola : 0, sama : 0, daya : 0, beri : 0, mudah : 0, oleh : 0, dasar : 0, ibu : 0, anak : 0, lansia : 0, tingkat : 0, bantu : 0, kader : 0, latih : 0, wahana : 0, liput : 0, imunisasi : 0.144, gizi : 0, cegah : 0, sakit : 0, nular : 0.
Vector Query	0.144

Kemudian dilakukan perhitungan untuk mencari nilai *similarity*. Perhitungan ini dilakukan berdasarkan rumus yang sudah tertera, sehingga ditemukan hasil sebagai berikut:

- A = Vector Query = 0.144
- B = Vector Dokumen
- Vector Dokumen d1 dan d2 = 0.697
- Vector Dokumen d3 = 0.788

$$\text{sim}(q, d1) = \frac{q \cdot d}{|q| \cdot |d|} = \frac{0.144 \times 0.697}{0.379473 \times 0.276139} = \frac{0.100368}{0.1047873} = 0.957826$$

$$\text{sim}(q, d2) = \frac{q \cdot d}{|q| \cdot |d|} = \frac{0.144 \times 0.697}{0.379473 \times 0.276139} = \frac{0.100368}{0.1047873} = 0.957826$$

$$\text{sim}(q, d3) = \frac{q \cdot d}{|q| \cdot |d|} = \frac{0.144 \times 0.788}{0.379473 \times 0.306888} = \frac{0.113472}{0.116455} = 0.974385$$

Berdasarkan hasil perhitungan tersebut, diketahui bahwa dokumen d1 dan d2 memiliki tingkat kemiripan sebesar 0.957826, sedangkan dokumen d3 memiliki nilai kemiripan yang lebih tinggi, yaitu 0.974385. Dengan demikian, dapat disimpulkan bahwa dokumen d3 lebih relevan atau sesuai dengan *query* dibandingkan dengan dokumen d1 dan d2.

```
def calculate_cosine_similarity(self, user_question):
    """
    Tahap 6: Menghitung Cosine Similarity antara pertanyaan user dan dataset
    """
    # Preprocessing pertanyaan user
    processed_question = self.text_preprocessing(user_question)

    # Transform pertanyaan user menjadi vektor TF-IDF
    user_tfidf = self.vectorizer.transform([processed_question])

    # Hitung cosine similarity dengan semua pertanyaan di dataset
    similarities = cosine_similarity(user_tfidf, self.tfidf_matrix)

    return similarities.flatten()
```

Gambar 4.7 Gambar code *Cosine Similarity*

Dengan pendekatan ini, chatbot dapat merespons dengan relevan bahkan untuk pertanyaan yang tidak persis sama dengan database, selama memiliki konteks atau kata kunci yang serupa.

BAB V

HASIL DAN PEMBAHASAN

5.1 Hasil Penelitian

Penelitian ini menghasilkan sebuah aplikasi Sistem Informasi Posyandu berbasis *mobile* yang dikembangkan menggunakan *framework React Native*. Aplikasi ini dilengkapi fitur chatbot berbasis AI dan berbagai fungsionalitas lain yang dirancang untuk mendukung kegiatan operasional Posyandu secara digital.

Fitur utama yang berhasil dikembangkan antara lain :

1. Login berbasis peran (admin, kader, bidan, orang tua).
2. Pencatatan dan pengelolaan data balita.
3. Pencatatan dan pengolahan data posyandu.
4. Akses riwayat tumbuh kembang anak oleh orang tua.
5. Monitoring data balita.
6. Chatbot untuk konsultasi dan informasi kesehatan

5.1.1 Langkah Implementasi

Tahapan implementasi sistem dilakukan melalui beberapa proses utama, mulai dari pembuatan aplikasi *mobile* menggunakan *React Native CLI*, pengembangan chatbot backend berbasis Python dan Flask, integrasi antara frontend dan backend, hingga deployment ke Firebase dan Railway. Struktur proyek terdiri dari dua bagian utama, yaitu **AwesomeProject** untuk aplikasi *React Native* dan **AwesomeProject-backendChatbot** untuk backend chatbot.

1. Pembuatan Proyek *React Native CLI*

Pembuatan aplikasi dimulai dengan inisialisasi proyek *React Native* menggunakan template TypeScript. Perintah yang digunakan:

- a. `npx react-native init AwesomeProject --template react-native-template-typescript`

Proyek disimpan dalam folder `AwesomeProject`. Setelah proses inisialisasi selesai, dilakukan uji coba awal untuk memastikan aplikasi dapat dijalankan di emulator atau perangkat fisik menggunakan perintah:

- b. `npx react-native run-android`

2. Instalasi Dependencies Tambahan di *React Native*

Setelah proyek berhasil dibuat, dilakukan instalasi beberapa dependencies tambahan untuk mendukung pengembangan antarmuka dan komunikasi dengan backend. Beberapa library yang digunakan antara lain:

- a. **Axios**, untuk melakukan HTTP request ke API backend: `npm install axios`
- b. **React Navigation**, untuk pengelolaan navigasi antar halaman: `npm install @react-navigation/native; npm install react-native-screens react-native-safe-area-context`
- c. **Type definitions** untuk integrasi TypeScript: `npm install --save-dev @types/react @types/react-native`
- d. Dan beberapa dependencies lainnya sesuai kebutuhan.

Struktur folder aplikasi disusun dengan pola modular, seperti `src/screens`, `src/services`, `src/types`, dan `src/components`, sehingga memudahkan pengelolaan kode.

3. Pengembangan Backend Chatbot

Bagian backend diletakkan di dalam folder `AwesomeProject-backendChatbot`. Backend dikembangkan menggunakan framework Flask,

dan menjalankan model chatbot berbasis pemrosesan teks seperti TF-IDF, *Cosine Similarity*, dan model Wikidepia (untuk pharaprase). Untuk memulai proyek backend, dibuat *virtual environment* dan diinstal dependencies:

- a. `python -m venv venv`
- b. `venv\Scripts\activate`
- c. `pip install flask flask-cors pandas scikit-learn transformers torch`
sastrawi

Logika utama chatbot ditulis di dalam file `chatbot_model.py`, sedangkan API Flask berada di `app.py` dengan beberapa endpoint seperti `/chat`, `/health`, dan `/update-dataset`.

4. Integrasi Frontend dan Backend

Integrasi antara frontend dan backend dilakukan melalui modul *ChatService* di sisi *React Native* yang bertugas mengatur request ke server backend menggunakan Axios. Base URL akan diatur sesuai dengan domain dari backend yang sudah dideploy, contoh:

- `const BASE_URL = 'https://web-production-61097.up.railway.app';`

Komunikasi antara frontend dan backend difokuskan pada endpoint `/chat`, di mana pengguna mengirimkan pertanyaan yang akan diproses oleh chatbot dan mengembalikan respons.

5. Deployment Backend ke Railway

Backend chatbot kemudian diunggah ke Railway, platform *cloud* yang mendukung deployment aplikasi Python berbasis Flask. Beberapa konfigurasi yang dilakukan:

- a. Menambahkan file requirements.txt: flask; flask-cors; pandas; scikit-learn; transformers; torch; sastrawi
- b. Menambahkan Procfile: web: python app.py
- c. Mengunggah kode ke Railway atau menghubungkan repository GitHub

Setelah berhasil dideploy, Railway akan memberikan URL publik yang digunakan sebagai endpoint API oleh aplikasi *React Native*.

6. Deployment Aplikasi Mobile ke Firebase

Setelah pengembangan selesai, aplikasi *React Native* dibuild menjadi APK menggunakan perintah:

- a. `cd android`
- b. `./gradlew assembleRelease`

APK yang dihasilkan dapat ditemukan di direktori:

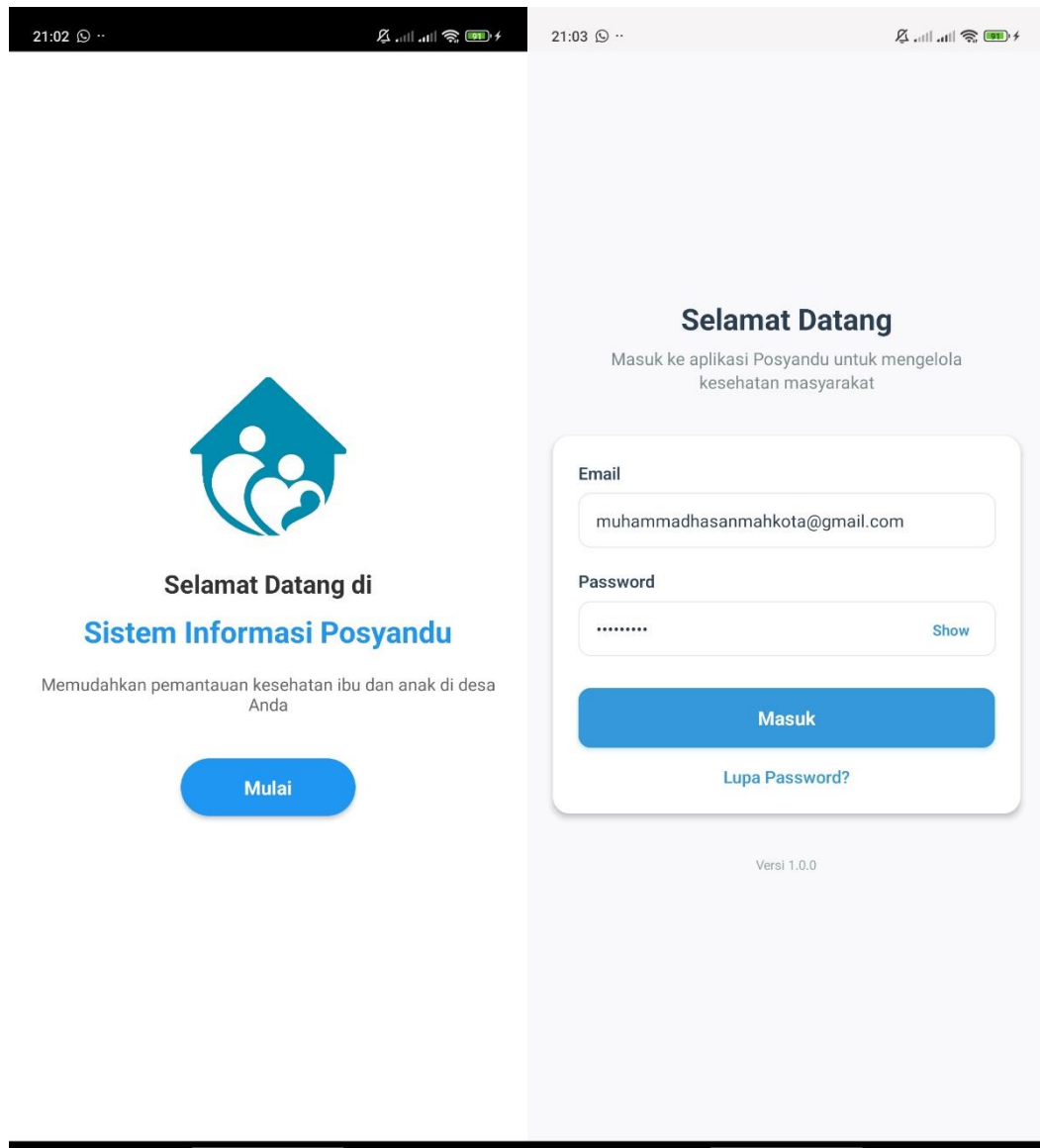
- c. `android/app/build/outputs/apk/release/app-release.apk`

File APK tersebut kemudian diunggah ke Firebase melalui layanan **App Distribution** untuk diuji oleh pengguna tester sebelum dirilis secara publik.

5.1.2 Tampilan dan Fitur Aplikasi

1. Halaman Welcome dan Login

Halaman ini merupakan tampilan awal saat aplikasi dibuka. Halaman welcome menampilkan informasi singkat mengenai aplikasi Posyandu. Selanjutnya, pengguna diarahkan ke halaman login untuk memasukkan username dan password sesuai dengan peran masing-masing (admin, kader, bidan, atau orang tua). Proses login ini bertujuan untuk memastikan autentikasi dan memberikan akses fitur sesuai dengan hak pengguna.

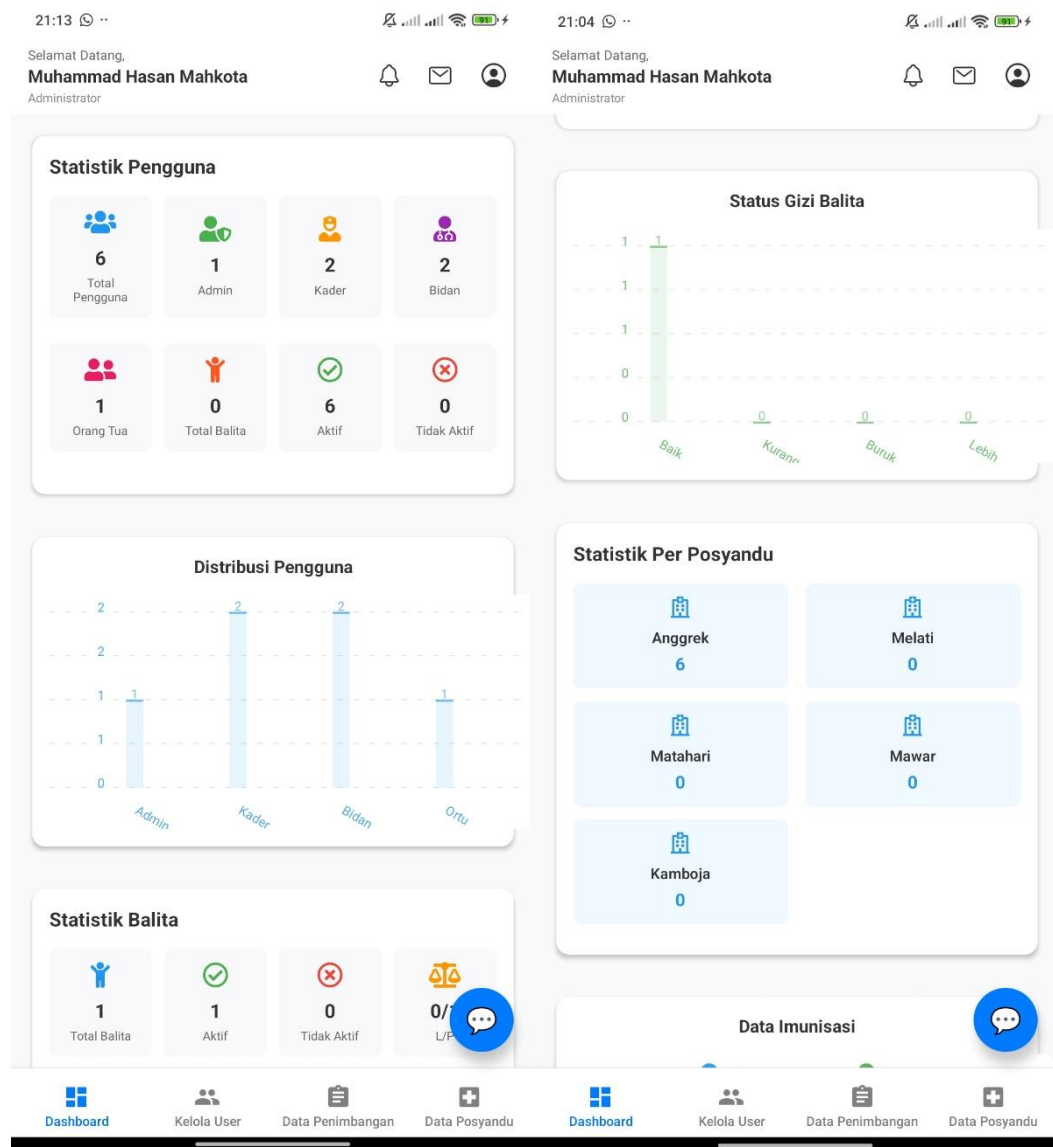


Gambar 5.1 Halaman Welcome dan Login

2. Dashboard dan Monitoring

Setelah berhasil login, pengguna akan diarahkan ke dashboard yang sesuai dengan peran masing-masing. Admin memiliki akses untuk melihat statistik umum aplikasi dan mengelola data pengguna, seperti menambah, mengubah, atau menghapus akun kader, bidan, dan orang tua. Kader dapat melihat daftar data balita yang terdaftar serta memantau perkembangan anak melalui fitur pencatatan digital. Bidan memiliki akses untuk meninjau data

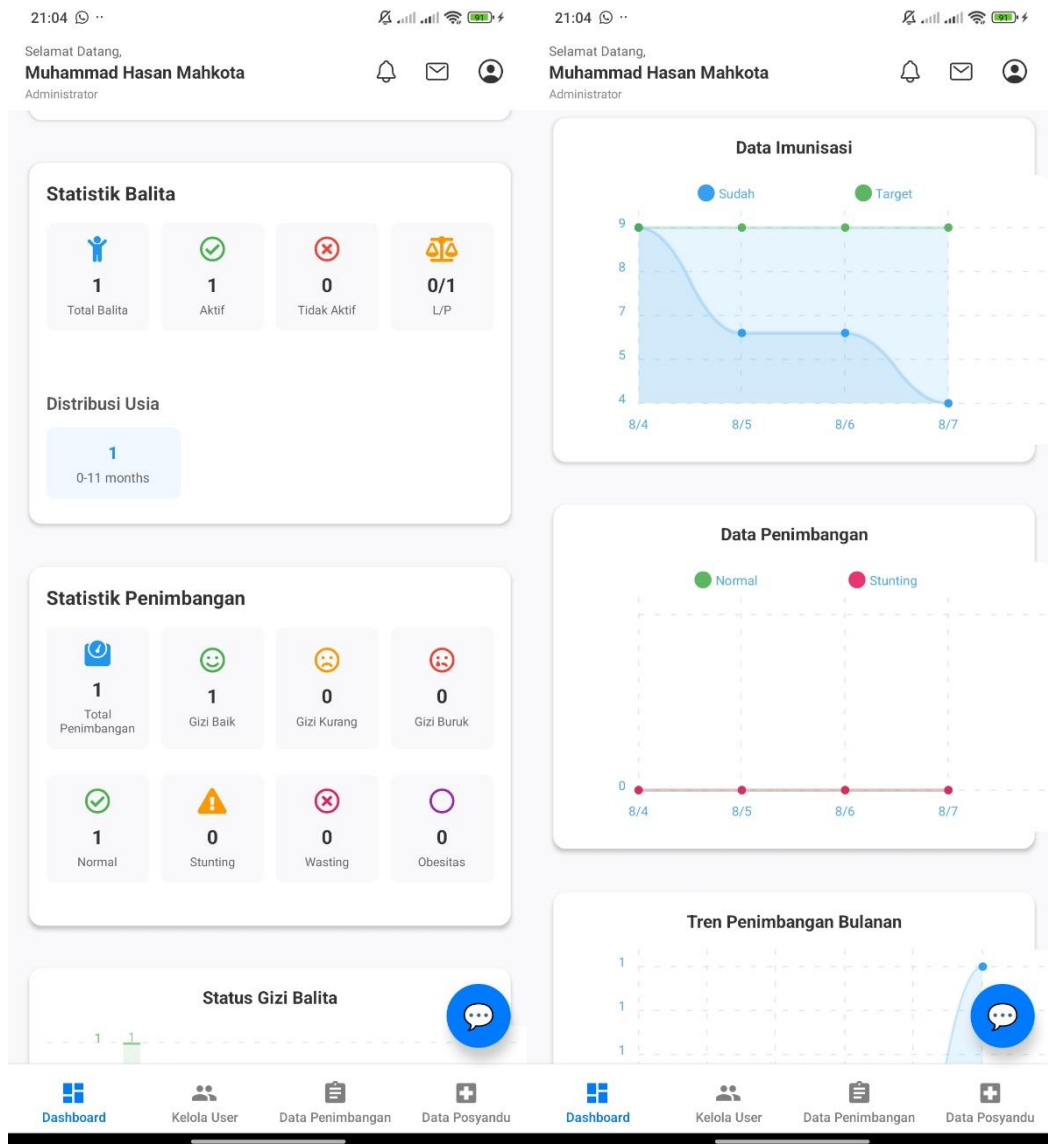
kesehatan anak sebagai dasar dalam pengambilan keputusan medis. Sementara itu, orang tua dapat memantau pertumbuhan anak mereka secara berkala, melihat jadwal kegiatan Posyandu, serta berkonsultasi melalui fitur chatbot.



Gambar 5.2 Dashboard Admin dan Monitoring bagian 1

Menampilkan tampilan utama setelah admin berhasil login. Pada dashboard ini, admin dapat melihat ringkasan data seperti jumlah pengguna, data balita, jadwal Posyandu, serta akses ke menu pengelolaan pengguna dan data.

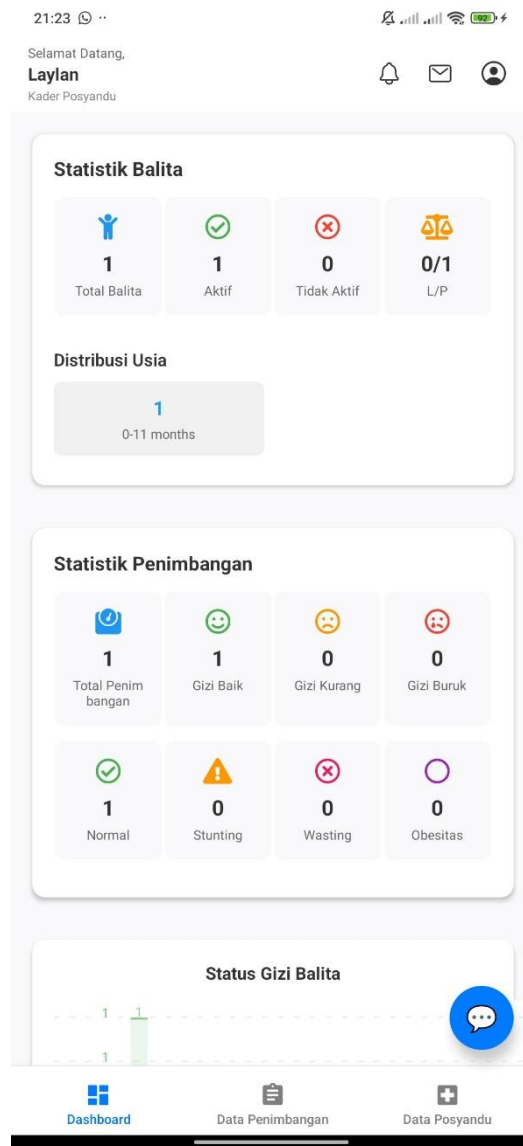
Tujuannya adalah memberikan kontrol penuh terhadap sistem dan monitoring aktivitas secara global.



Gambar 5.3 Dashboard Admin dan Monitoring bagian 2

Gambar 5.3 merupakan kelanjutan dari tampilan dashboard admin yang menampilkan visualisasi data dalam bentuk grafik. Grafik tersebut digunakan untuk memperlihatkan perkembangan balita berdasarkan data penimbangan yang telah dicatat, seperti grafik berat badan, tinggi badan, dan status gizi dalam rentang waktu tertentu. Visualisasi ini bertujuan untuk

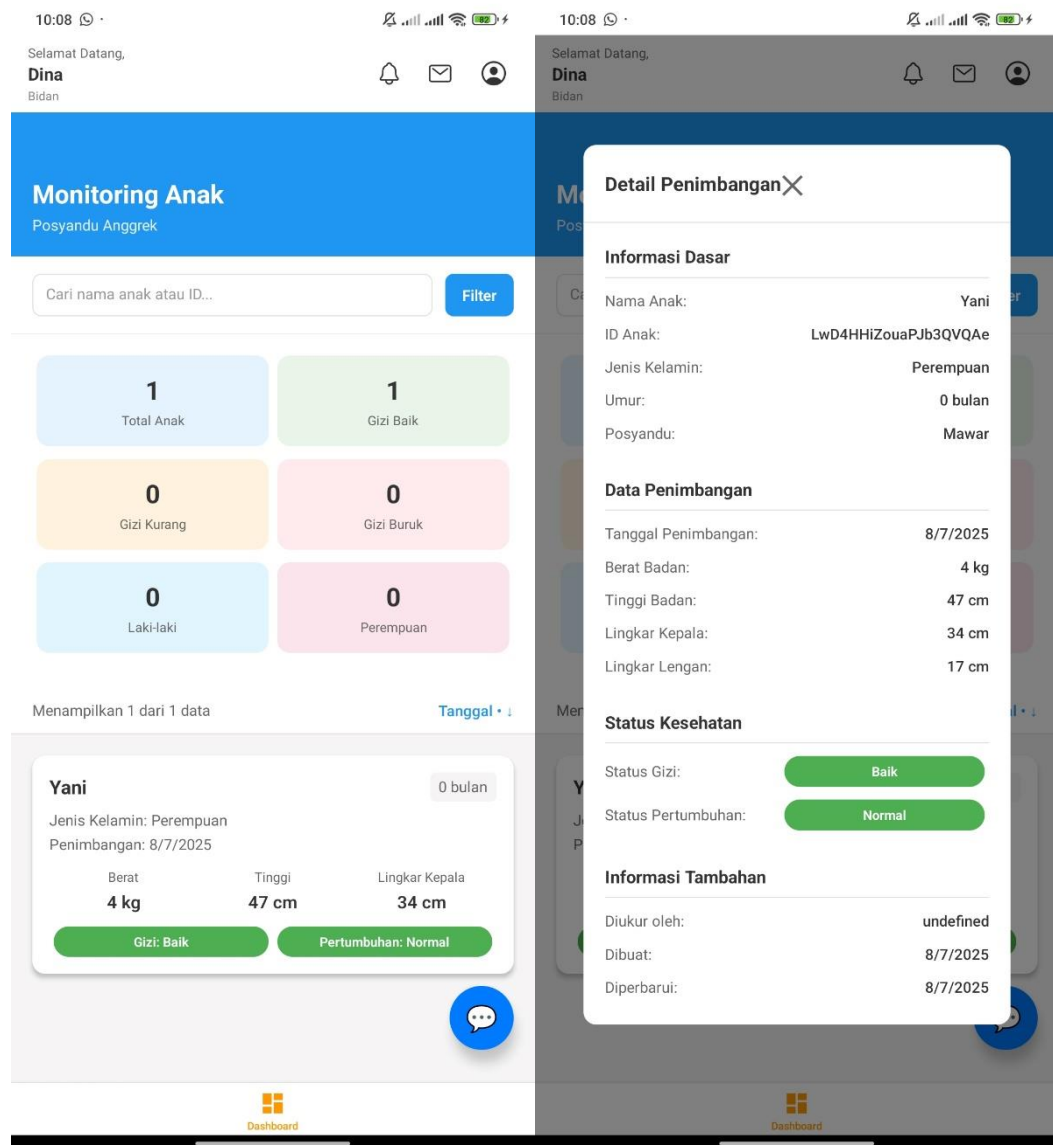
membantu admin dalam melihat tren perkembangan dan pertumbuhan anak-anak secara kolektif di wilayah layanan Posyandu.



Gambar 5.4 Dashboard Kader

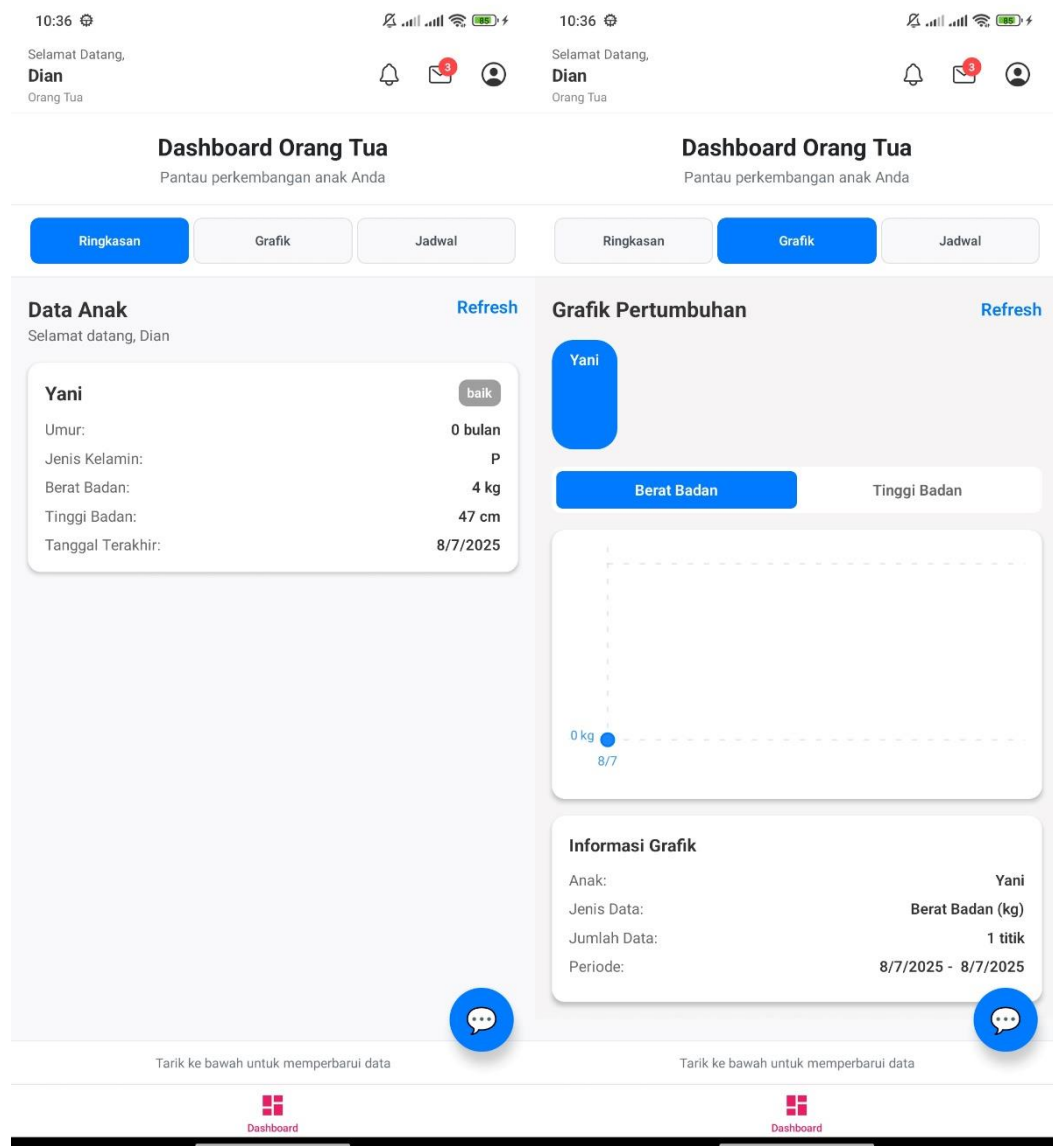
Dashboard yang disediakan untuk kader memiliki tampilan dan fungsi yang serupa dengan milik admin, seperti menampilkan ringkasan jumlah pengguna, data balita, serta visualisasi grafik berdasarkan data yang telah diinput guna memudahkan pemantauan dan analisis perkembangan balita di posyandu. Namun, terdapat perbedaan dari sisi hak akses, di mana

dashboard kader tidak menyediakan fitur untuk mengelola data pengguna sebagaimana yang tersedia pada dashboard admin. Oleh karena itu, akses kader dibatasi hanya pada fitur utama seperti dashboard dan data posyandu, sesuai dengan tugas dan tanggung jawab yang diemban.



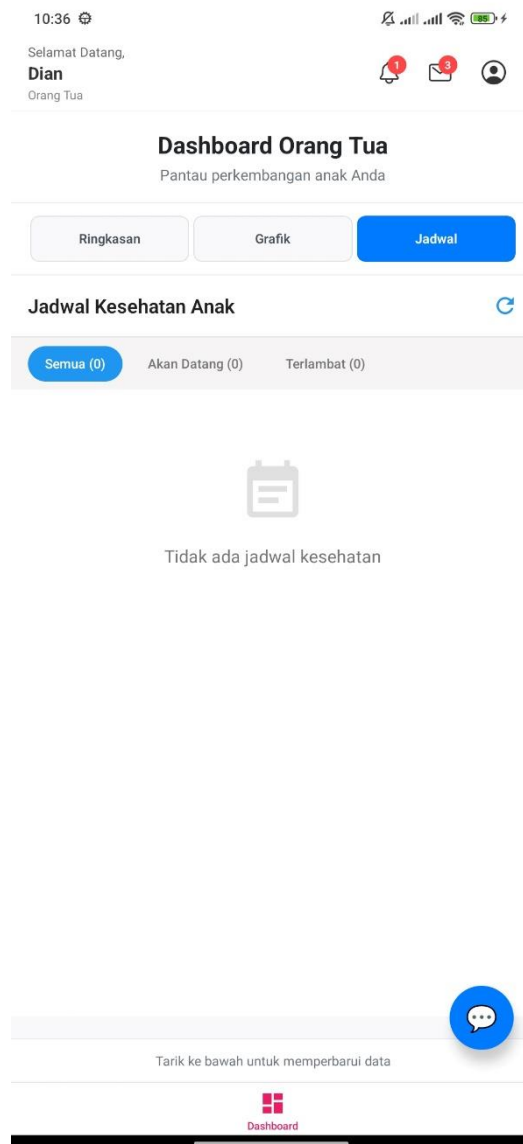
Gambar 5.5 Dashboard Bidan

Dashboard untuk bidan menampilkan informasi kesehatan balita secara detail. Bidan dapat melihat riwayat tinggi dan berat badan balita, status imunisasi. Fitur ini penting untuk mendukung analisis kesehatan anak.



Gambar 5.6 Dashboard Orang Tua bagian 1

Dashboard ini menampilkan halaman awal yang dilihat oleh orang tua setelah login. Pada bagian ini, orang tua dapat melihat informasi umum anak mereka, seperti nama, tanggal lahir, serta ringkasan data pertumbuhan. Tujuan dari fitur ini adalah untuk meningkatkan keterlibatan orang tua dalam memantau tumbuh kembang anak secara mandiri. Orang tua tidak perlu lagi menunggu jadwal Posyandu untuk mengetahui perkembangan anak, karena semua informasi tersedia langsung di aplikasi.

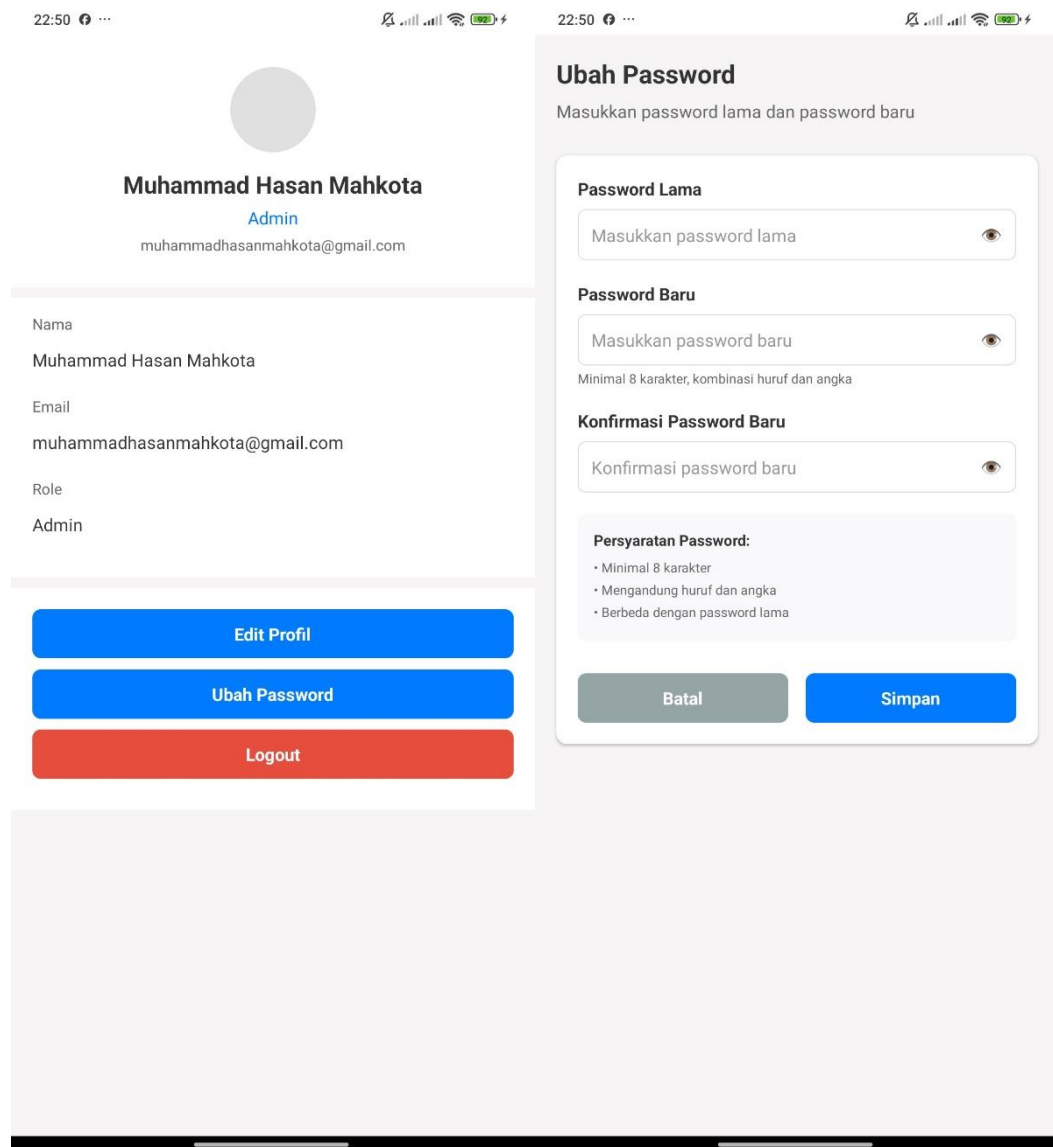


Gambar 5.7 Dashboard Orang Tua bagian 2

Melanjutkan tampilan sebelumnya, bagian ini menampilkan jadwal Posyandu yang akan datang. Juga tersedia akses langsung ke fitur chatbot untuk konsultasi kesehatan, menjadikan aplikasi ini sebagai alat bantu pemantauan tumbuh kembang yang praktis.

3. Tampilan Profil dan Ubah Password

Masing-masing pengguna dapat mengakses halaman profil untuk melihat informasi akun mereka. Di dalamnya juga tersedia opsi untuk mengubah password secara mandiri, demi menjaga keamanan akses aplikasi.



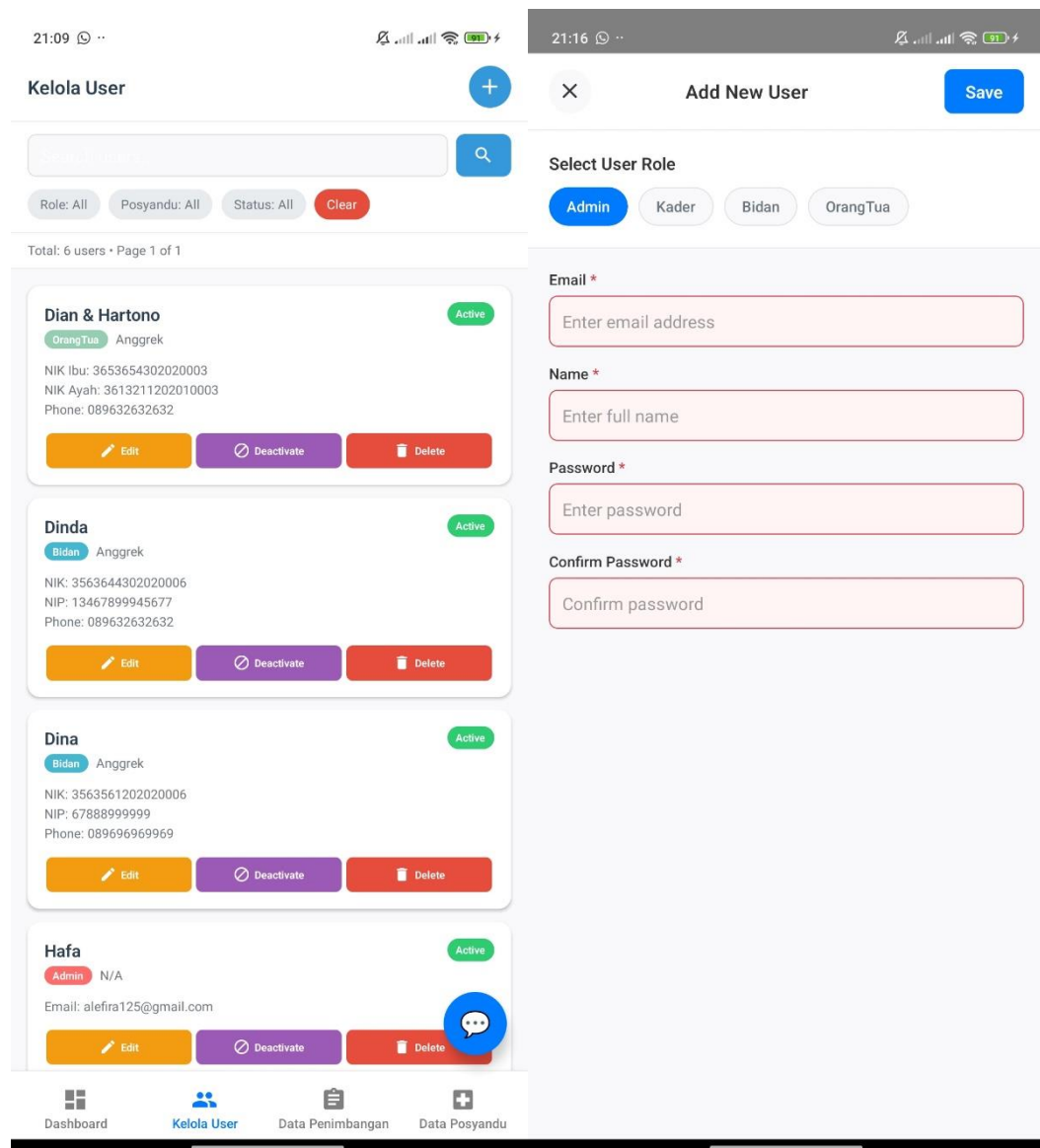
Gambar 5.8 Profil dan Ubah Password

4. Pengelolaan Pengguna

Berikut tampilan antarmuka pengelolaan pengguna yang hanya dapat diakses oleh admin. Pada bagian awal ini, admin dapat menambahkan pengguna baru ke dalam sistem dengan mengisi data seperti nama, email,

username, password, serta menentukan peran pengguna (kader, bidan, atau orang tua). Proses ini sangat penting untuk mendukung manajemen pengguna yang terstruktur dan memastikan bahwa setiap pengguna memiliki akses yang sesuai dengan perannya.

Tersedia juga fitur pencarian yang memudahkan admin untuk menemukan pengguna tertentu tanpa harus menelusuri seluruh daftar secara manual.



Gambar 5.9 Pengelolaan Pengguna bagian 1

The image shows a mobile application interface for adding new users. It consists of two side-by-side forms, each titled 'Add New User' with a 'Save' button. Both forms have a 'Select User Role' section at the top with buttons for 'Admin', 'Kader', 'Bidan', and 'OrangTua'. The left form is for a 'Kader' user, and the right form is for an 'OrangTua' user. Both forms include fields for 'Email *', 'Password *', 'Confirm Password *', 'Name *', 'NIK *', 'Posyandu', 'Desa *', 'Kecamatan *', and 'Tahun Menjadi'. The right form also includes fields for 'Data Ibu' (NIK Ibu, Nama Ibu), 'Date of Birth Ibu', 'Place of Birth Ibu', 'Golongan Darah Ibu', and 'Data Ayah'.

Add New User **Save**

Select User Role

Admin Kader Bidan OrangTua

Email *

Enter email address

Password *

Enter password

Confirm Password *

Confirm password

Name *

Enter full name

NIK *

Enter 16-digit NIK

Posyandu

Anqqrek

Desa *

Jabunq Sisir

Kecamatan *

Jabunq

Tahun Menjadi

2024

Add New User **Save**

Select User Role

Admin Kader Bidan OrangTua

Email *

Enter email address

Password *

Enter password

Confirm Password *

Confirm password

Data Ibu

NIK Ibu *

Enter NIK Ibu

Nama Ibu *

Enter mother name

Date of Birth Ibu

Select date

Place of Birth Ibu

Enter mother place of birth

Golongan Darah Ibu

O

Data Ayah

Gambar 5.10 Pengelolaan Pengguna bagian 2

21:16 21:16

Add New User Save

Select User Role

Admin Kader **Bidan** OrangTua

Email *
Enter email address

Password *
Enter password

Confirm Password *
Confirm password

NIK *
Enter 16-digit NIK

NIP *
Enter NIP

Posyandu *
Anqqrek

Name *
Enter full name

Place of Birth *
Enter place of birth

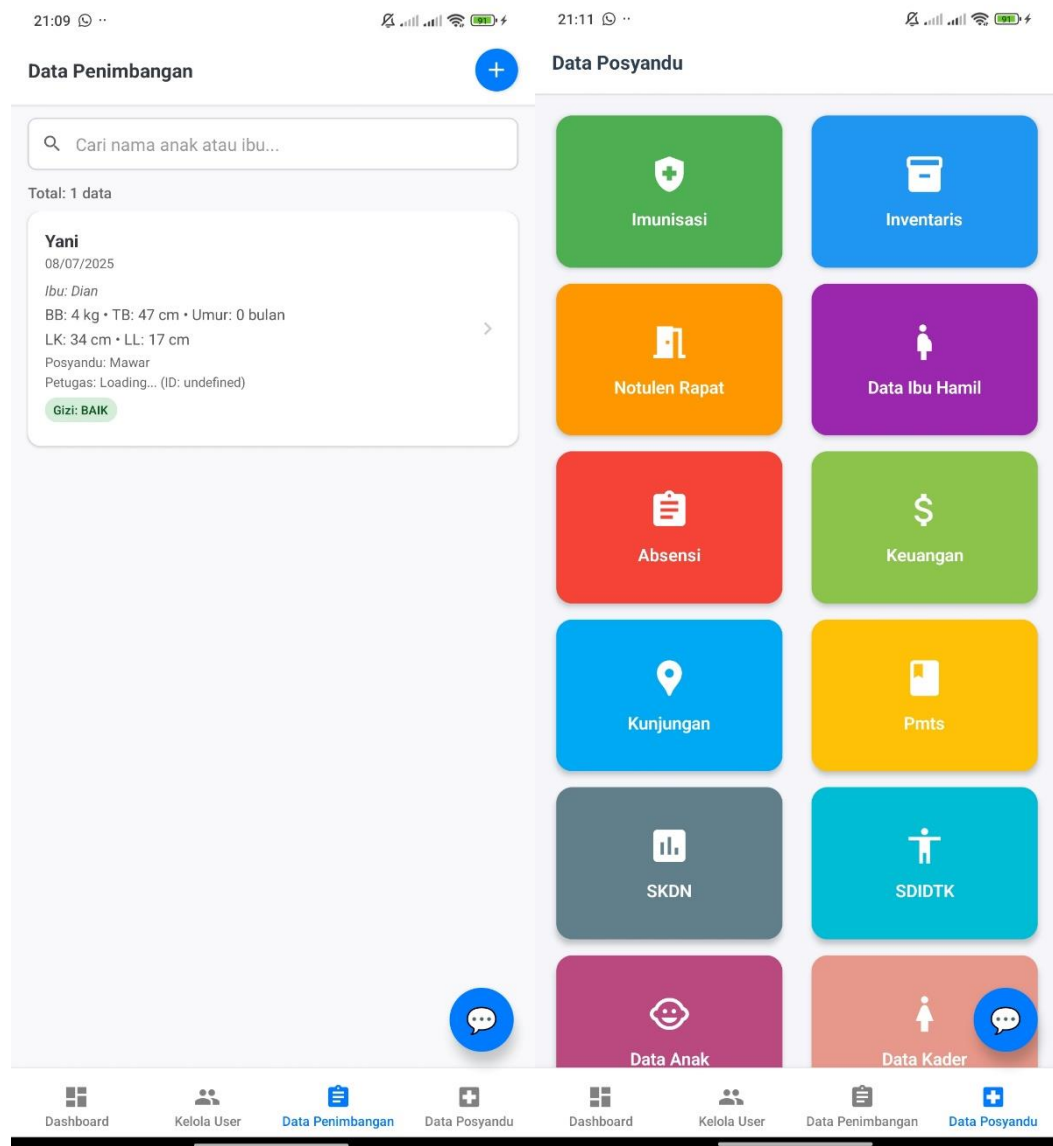
Date of Birth *
Select date

Gender *

Gambar 5.11 Pengelolaan Pengguna bagian 3

5. Data Posyandu

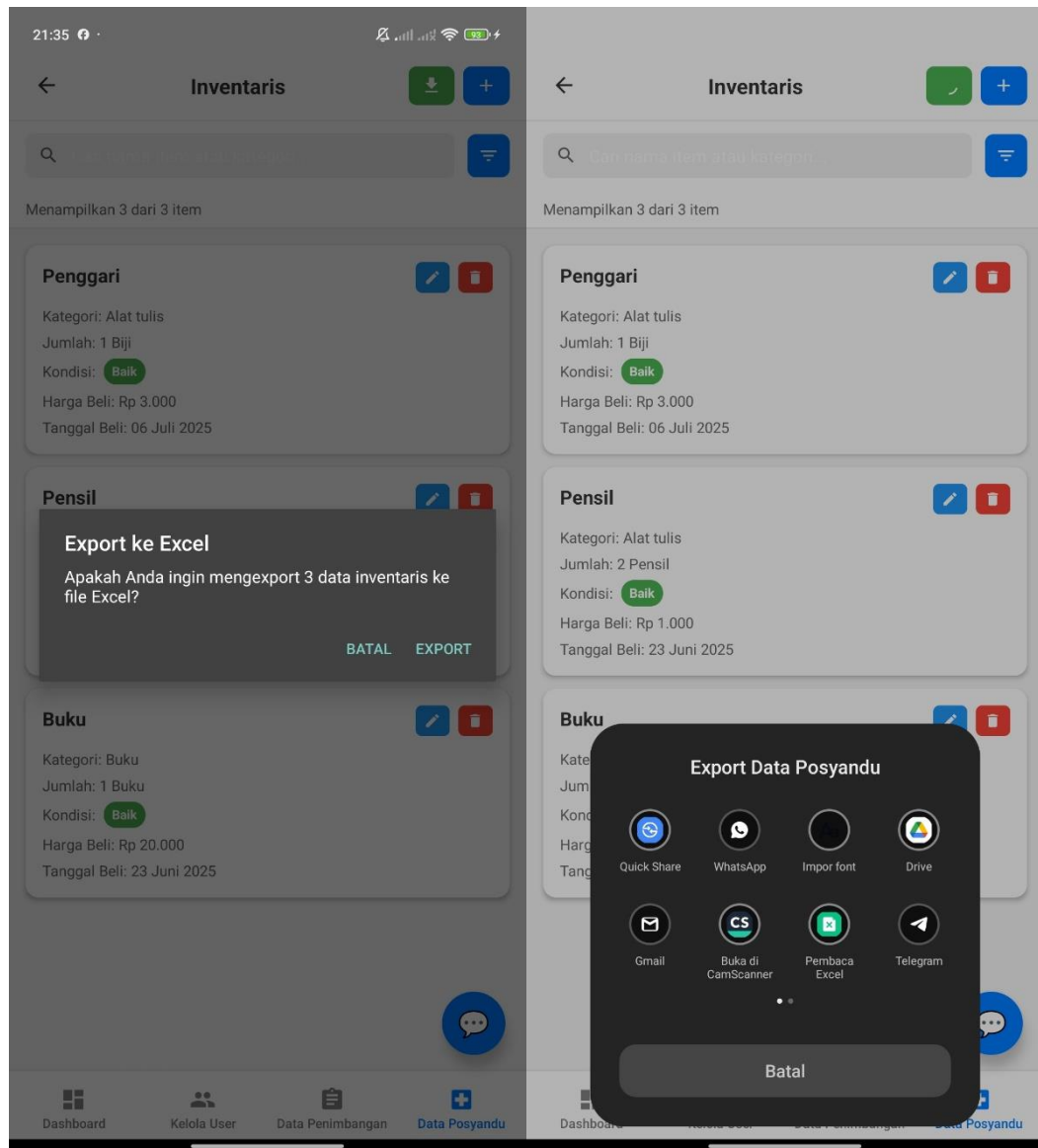
Menyediakan berbagai data penting terkait kegiatan Posyandu. Di dalamnya terdapat data penimbangan anak (berat, tinggi, dan status gizi) serta menu untuk mengakses data posyandu lainnya. Data ditampilkan secara terstruktur dan mudah dipahami.



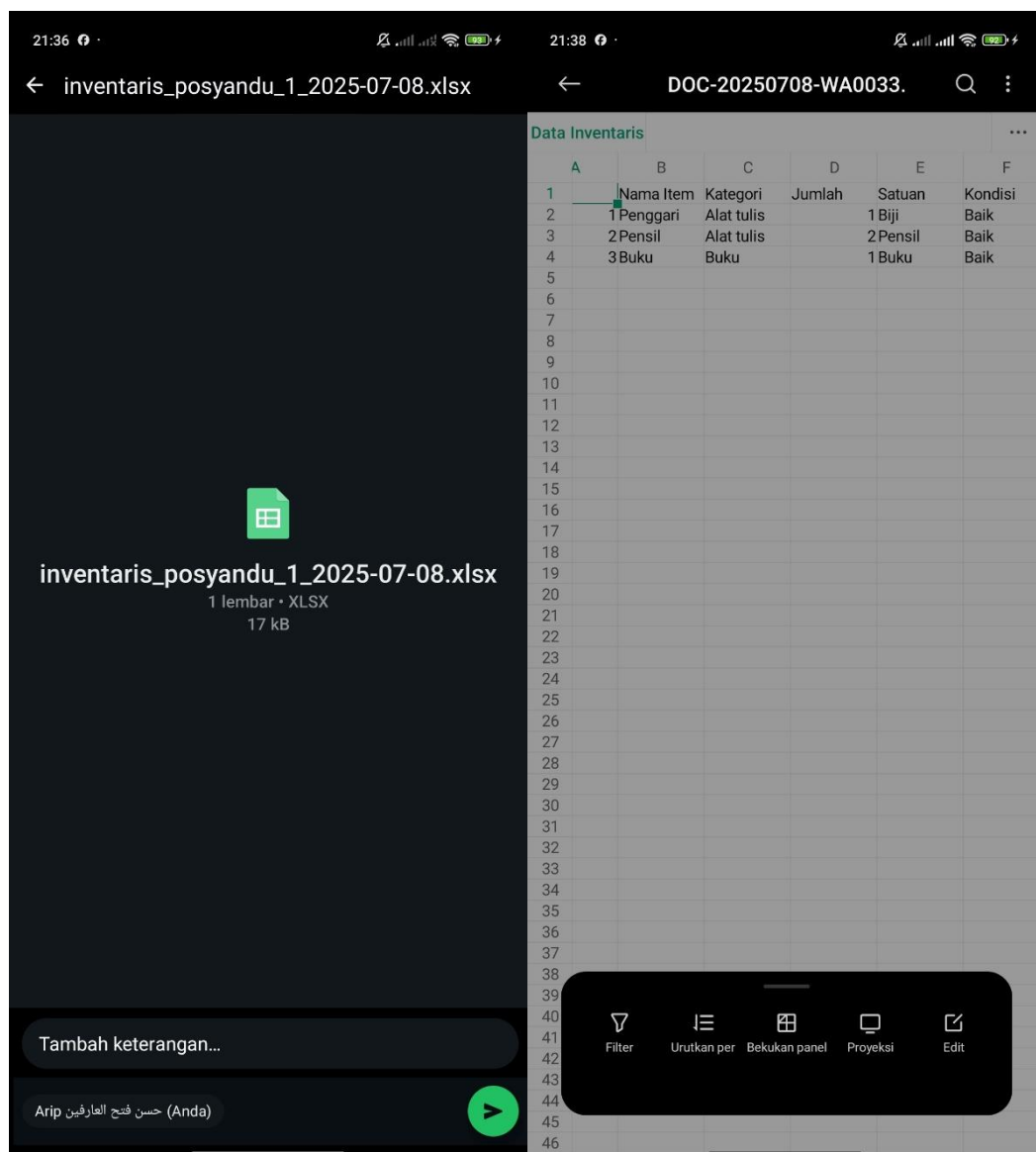
Gambar 5.12 Data Posyandu

6. Export Data

Fitur ini memungkinkan admin atau kader untuk mengunduh data dalam format Excel (.xlsx), seperti data penimbangan atau laporan bulanan. Hal ini berguna untuk pelaporan ke puskesmas dan dokumentasi offline. Proses ekspor dilakukan dengan satu klik, dan data akan otomatis terformat rapi.



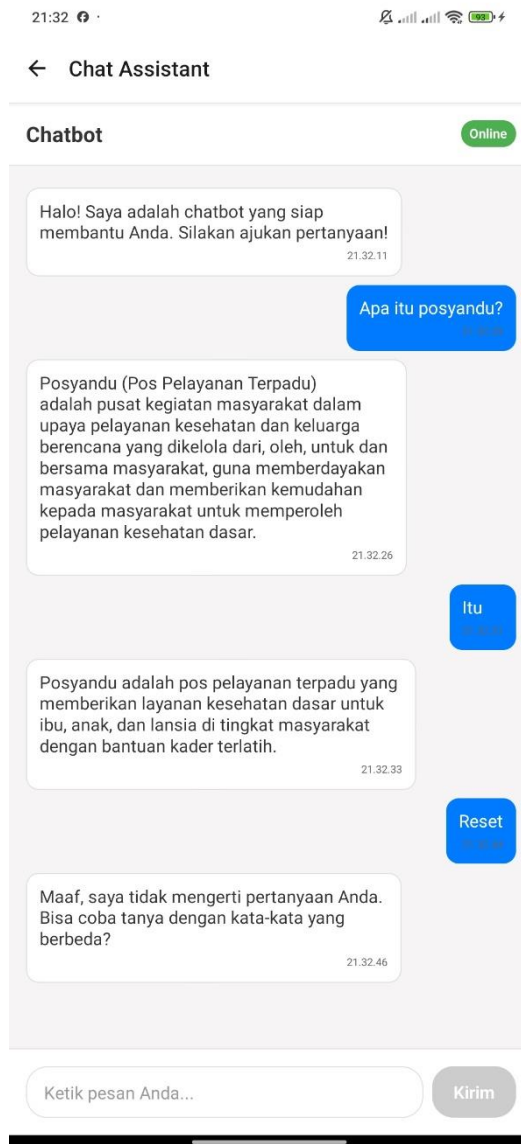
Gambar 5.13 Simulasi Export Data bagian 1



Gambar 5.14 Simulasi Export Data bagian 2

7. Fitur Chatbot

Chatbot tersedia sebagai sarana konsultasi dan edukasi kesehatan. Pengguna dapat mengajukan pertanyaan seputar jadwal imunisasi, pencegahan stunting, atau informasi lain terkait kesehatan anak. Chatbot dirancang untuk memberikan jawaban cepat dan relevan, sehingga memudahkan pengguna mendapatkan informasi tanpa harus menunggu kegiatan Posyandu.



Gambar 5.15 Fitur Chatbot

Fitur ini masih memiliki keterbatasan dalam menangani variasi pertanyaan, karena jumlah dataset yang digunakan masih terbatas. Akibatnya, jika pengguna mengajukan pertanyaan yang tidak sesuai atau tidak tercakup dalam dataset, sistem cenderung memberikan jawaban yang kurang tepat atau tidak relevan.

5.2 Pembahasan

5.2.1 Kesesuaian Sistem dengan Spesifikasi

Pengujian sistem dilakukan dengan pendekatan *Black Box Testing* untuk menilai apakah setiap fitur utama berfungsi sesuai kebutuhan yang telah ditentukan.

Metode ini difokuskan pada aspek fungsionalitas sistem tanpa memeriksa kode program secara langsung. Sebagai bagian dari proses ini, dilakukan pula *User Acceptance Testing (UAT)* dengan melibatkan pengguna untuk memastikan bahwa aplikasi dapat digunakan secara optimal dalam skenario nyata. Uji coba dilakukan terhadap berbagai fitur penting, seperti proses login, *input/update/delete* user, pencatatan data, dan konsultasi melalui chatbot. Chatbot juga diuji secara internal guna menjamin bahwa respons yang diberikan sesuai dengan pertanyaan pengguna. Hasil pengujian menunjukkan bahwa seluruh fitur telah berjalan sesuai ekspektasi pengguna.

Berikut hasil pengujian system berdasarkan setiap user:

Tabel 5.1 Hasil Pengujian Sistem Admin dan Kader

No.	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Login User	Pengguna memasukkan username dan password sesuai akun	Sistem menampilkan halaman sesuai peran pengguna	Sesuai harapan	Valid
2	Pencatatan Data Balita	Kader menginput data balita (nama, usia, berat, tinggi)	Data balita tersimpan dan muncul dalam daftar riwayat	Sesuai harapan	Valid
3	Tambah User	Admin menambahkan user baru	User baru terinput	Sesuai harapan	valid
4	Input Data Posyandu	Admin dan kader mengisi, mengedit dan menghapus data posyandu dalam form input	Data posyandu tersimpan dan tampil pada daftar posyandu. Data juga bias diedit dan dihapus	Belum Sesuai	Tidak Valid
5	Monitoring Data Penimbangan dan Imunisasi.	Pengguna dapat melakukan monitoring sesuai role masing masing	Data terpampang didashboard untuk monitoring	Sesuai harapan	Valid
6	Export Data ke File Excel	Admin/kader memilih menu export dan mengunduh data ke format .xlsx	Data berhasil diekspor ke file Excel dan bisa dibuka dengan rapi	Sesuai harapan	Valid

Tabel 5.1 Lanjutan

No.	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
7	Akses Chatbot	Pengguna mengajukan pertanyaan ke chatbot (misal: “Apa itu Posyandu”)	Chatbot merespons dengan informasi yang sesuai dan relevan	Belum Sesuai	Tidak Valid

Tabel 5.2 Hasil Pengujian Sistem Bidan

No.	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Login User	Pengguna memasukkan username dan password sesuai akun	Sistem menampilkan halaman sesuai peran pengguna	Sesuai harapan	Valid
2	Monitoring Data Balita	Bidan dapat melakukan monitoring pada tumbuh kembang seluruh balita	Data terpampang didashboard untuk monitoring	Sesuai harapan	Valid
3	Konsultasi Chatbot	Pengguna mengajukan pertanyaan ke chatbot (misal: “Apa itu Posyandu”)	Chatbot merespons dengan informasi yang sesuai dan relevan	Belum Sesuai	Tidak Valid

Tabel 5.3 Hasil Pengujian Sistem Orang Tua

No.	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Login User	Pengguna memasukkan username dan password sesuai akun	Sistem menampilkan halaman sesuai peran pengguna	Sesuai harapan	Valid
2	Monitoring Data Anak	Orang tua dapat melakukan monitoring pada tumbuh kembang anak	Data terpampang didashboard untuk monitoring	Sesuai harapan	Valid
3	Konsultasi Chatbot	Pengguna mengajukan pertanyaan ke chatbot (misal: “Apa itu Posyandu”)	Chatbot merespons dengan informasi yang sesuai dan relevan	Belum Sesuai	Tidak Valid

Pada fitur input data posyandu, sebagaimana tercantum dalam Tabel 5.1, dinyatakan tidak valid karena masih terdapat bug, khususnya pada mekanisme

callback yang belum berjalan dengan baik pada beberapa tombol. Hal serupa terjadi pada fitur chatbot yang ditampilkan dalam Tabel 5.1, 5.2, dan 5.3, di mana sistem belum mampu memberikan jawaban yang akurat terhadap pertanyaan pengguna. Hal ini disebabkan oleh keterbatasan jumlah dan cakupan data pada dataset yang digunakan.

Meskipun masih terdapat kekurangan pada fitur input data posyandu dan chatbot, secara keseluruhan hasil pengujian menggunakan metode *Black Box* menunjukkan bahwa fungsi utama dalam aplikasi dapat berjalan sesuai dengan skenario yang telah dirancang, tanpa ditemukan kesalahan dalam interaksi sistem. Untuk memastikan kesesuaian aplikasi dengan kebutuhan dan harapan pengguna akhir, dilakukan pula pengujian lanjutan melalui pendekatan *User Acceptance Test* (UAT).

Pengujian *User Acceptance Test* (UAT) dilakukan untuk mengetahui sejauh mana sistem yang dibangun telah sesuai dengan kebutuhan pengguna dan layak untuk digunakan dalam kondisi operasional sebenarnya. UAT melibatkan beberapa pengguna akhir yang berasal dari berbagai peran, seperti admin, kader, dan orang tua, untuk mencoba langsung sistem dan memberikan evaluasi terhadap kemudahan penggunaan, kejelasan antarmuka, serta kesesuaian fitur.

Pengujian dilakukan melalui penyebaran kuesioner berbasis skala Likert dengan lima pilihan jawaban: Sangat Setuju, Setuju, Netral, Tidak Setuju, dan Sangat Tidak Setuju.

Tabel 5.4 Skor Jawaban

No	Jawaban	Skor
1	Sangat Setuju (SS)	5
2	Setuju (S)	4
3	Netral (N)	3
4	Tidak Setuju (TS)	2
5	Sangat Tidak Setuju (STS)	1

Tiap jawaban memiliki bobot nilai dari 5 hingga 1. Skor total dihitung menggunakan rumus berikut:

1. Skor Total = $\sum (\text{jawaban} \times \text{skor})$
2. Skor Ideal = jumlah pernyataan $\times$ jumlah responden $\times$
 skor tertinggi Nilai Persentase (%) = $(\text{Total Skor} / \text{Skor Ideal}) \times 100$

(Ini adalah interval jarak dari terendah 0% hingga tertinggi 100%) Kriteria skor berdasarkan interval dapat dilihat pada tabel dibawah ini :

Tabel 5.5 Kriteria Skor

No	Kategori	Keterangan
1	0%-19,99%	Sangat Tidak Setuju
2	20%-39,99%	Tidak Setuju
3	40%-59,99%	Netral
4	60%-79,99%	Setuju
5	80%-100%	Sangat Setuju

Berikut adalah data hasil kuesioner terhadap 10 pernyataan yang telah dirumuskan dengan redaksi yang lebih panjang dan jelas:

Tabel 5.6 Kuisioner Pengguna

No	Pernyataan	SS	S	N	TS	STS	Total Skor
1	Aplikasi mudah diakses dan diinstal di perangkat Android/iOS saya.	3	5	2	0	0	43

Tabel 5.6 Lanjutan

No	Pernyataan	SS	S	N	TS	STS	Total Skor
1	Aplikasi mudah diakses dan diinstal di perangkat Android/iOS saya.	3	5	2	0	0	43
2	Proses login sesuai peran (admin/kader/bidan/orang tua) berjalan dengan baik.	3	4	2	1	0	42
3	Tampilan aplikasi mudah dipahami dan digunakan.	2	4	3	1	0	40
4	Fitur pengelolaan user/pengguna berfungsi sesuai dengan harapan.	2	4	3	1	0	40
5	Saya dapat melihat dan memantau riwayat tumbuh kembang anak dengan jelas melalui aplikasi.	1	4	4	1	0	39
6	Fitur pengelolaan data Posyandu (input/edit/hapus) mudah digunakan.	2	3	3	2	0	38
7	Fitur ekspor data ke Excel mudah digunakan.	2	3	3	1	1	37
8	Chatbot dapat memberikan jawaban yang akurat terhadap pertanyaan saya.	0	0	5	3	2	25
9	Chatbot merespons pertanyaan dengan cepat.	0	0	4	4	2	24
10	Aplikasi ini membantu saya dalam kegiatan Posyandu sehari-hari.	2	4	2	1	1	39
11	Saya merasa puas setelah menggunakan aplikasi ini.	1	4	3	1	1	38
12	Saya bersedia merekomendasikan aplikasi ini untuk digunakan oleh Posyandu lainnya.	2	4	2	2	0	40

$$\text{Skor Ideal} = 12 \times 10 \times 5 = 600$$

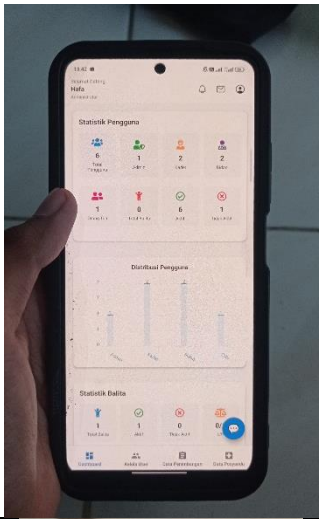
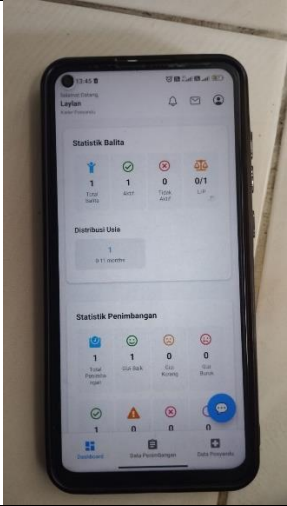
$$\text{Persentase} = (455/600) \times 100\% = 75,83\%$$

Berdasarkan hasil pengisian kuisioner oleh 10 responden terhadap 12 pernyataan, diperoleh skor total sebesar **455** dari skor ideal **600**. Nilai ini menghasilkan persentase **75,83%**, yang berada pada kategori **“Setuju”** sesuai

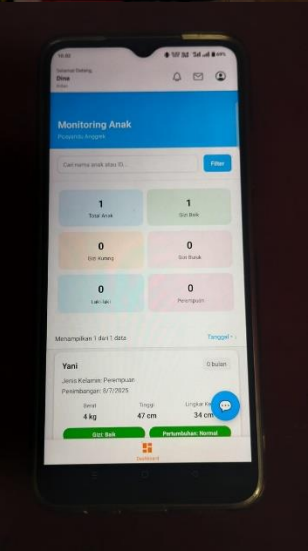
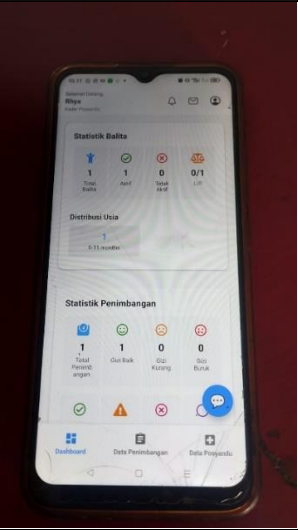
dengan Tabel 5.2 Kriteria Skor. Dengan demikian, aplikasi dinilai telah sesuai dengan kebutuhan sebagian besar pengguna, meskipun masih terdapat beberapa aspek yang perlu ditingkatkan, terutama pada fitur chatbot.

Berikut tabel simulasi pada beberapa device/handphone:

Tabel 5.7 Hasil Simulasi pada Beberapa Device

No	Nama Device/Handphone	Gambar
1	Redmi note 11	
2	Redmi note 9	

Tabel 5.7 Lanjutan

No	Nama Device/Handphone	Gambar
3	Samsung Galaxy A21s	
4	Oppo A38	
5	Realme Narzo 50a	

5.2.2 Pembahasan Akurasi Chatbot (Simulasi)

Setelah dilakukan *User Acceptance Testing* (UAT), fitur chatbot telah diuji secara langsung oleh 10 pengguna aktif Posyandu dengan mengajukan pertanyaan sesuai kebutuhan informasi mereka. Di samping itu, pengujian internal juga dilakukan oleh pengembang melalui sejumlah pertanyaan simulatif untuk menguji kemampuan pencocokan teks.

Sebagai bagian dari pengujian internal, chatbot diuji menggunakan pendekatan *information retrieval* berbasis teks, dengan cara mencocokkan pertanyaan (*query*) pengguna terhadap dokumen jawaban yang telah disediakan sebelumnya. Setiap dokumen berisi pernyataan atau informasi terkait Posyandu, yang telah melalui tahap *preprocessing* menggunakan metode seperti *case folding*, *tokenizing*, *stopword removal*, dan *stemming*. Tahapan ini dilanjutkan dengan pembobotan menggunakan *Term Frequency–Inverse Document Frequency* (TF-IDF) dan pemodelan dengan *Vector Space Model* (VSM).^j

Dalam pengujian ini, chatbot akan menghitung tingkat kemiripan (*similarity*) antara *query* dan dokumen menggunakan metode *cosine similarity*. Dokumen dengan nilai kemiripan tertinggi akan dijadikan dasar jawaban yang diberikan oleh chatbot.

Berikut merupakan contoh dokumen yang menjadi referensi chatbot:

Tabel 5.8 Dataset Dokumen Chatbot Berdasarkan Kategori

No	Dokumen	Kategori
1	Posyandu adalah pos pelayanan terpadu yang dikelola oleh masyarakat.	Informasi Umum
2	Posyandu merupakan tempat pelayanan kesehatan dasar bagi ibu dan anak.	Informasi Umum
3	Posyandu biasanya buka pada hari Minggu atau hari libur pagi hari.	Jadwal

Tabel 5.8 Lanjutan

No	Dokumen	Kategori
4	Jadwal kegiatan posyandu disesuaikan dengan masing-masing wilayah.	Jadwal
5	Imunisasi anak merupakan tindakan wajib untuk mencegah penyakit berbahaya.	Imunisasi

Kemudian, chatbot diuji dengan sejumlah pertanyaan atau *query* yang disusun oleh pengembang sebagai simulasi uji akurasi:

Tabel 5.9 Query (Pertanyaan Pengguna) untuk Uji Chatbot

No	Query
1	Apa itu posyandu?
2	Posyandu buka hari apa saja?
3	Anak saya belum imunisasi, apakah boleh ditunda?
4	Jam berapa posyandu dimulai?
5	Imunisasi itu penting atau tidak?

Berdasarkan pengujian internal, chatbot mampu memberikan respons yang cukup relevan dan akurat, sebagaimana ditampilkan pada Gambar 5.10. Fitur ini juga berhasil melewati uji fungsionalitas pada *Black Box Testing*, dengan hasil bahwa chatbot dapat dijalankan tanpa kesalahan sistem, dan merespons sesuai algoritma pencocokan teks.

Namun, hasil UAT menunjukkan skor chatbot merupakan yang paling rendah dibanding fitur lainnya, dengan perolehan skor:

1. Pertanyaan 8 (akurasi jawaban): 25 dari 50
2. Pertanyaan 9 (kecepatan respons): 24 dari 50

Dari total skor 100 pada dua indikator tersebut, hanya tercapai 49 poin (49%), yang berada dalam kategori Netral berdasarkan Tabel 5.3 Kriteria Skor.

Perbandingan ini mengindikasikan bahwa meskipun secara teknis chatbot telah berfungsi dengan baik dan lolos uji fungsional, namun dari sisi pengguna, sistem masih dirasa belum cukup akurat dan responsif dalam menjawab pertanyaan, khususnya untuk pertanyaan dengan gaya bahasa yang lebih bebas atau tidak baku.

Dengan demikian, dapat disimpulkan bahwa pengujian internal dan pengujian pengguna menghasilkan temuan yang berbeda:

1. Secara sistem: chatbot berjalan sesuai rancangan.
2. Secara pengalaman pengguna: chatbot masih memerlukan perbaikan, khususnya pada pengayaan dataset dan pemahaman konteks bahasa alami pengguna agar jawaban yang diberikan lebih tepat sasaran dan informatif.

5.2.3 Implikasi dan Harapan

Sistem yang dikembangkan diharapkan dapat memberikan manfaat berikut:

1. Memudahkan kader dalam pencatatan data secara digital.
2. Meningkatkan efisiensi pelaporan dan penyimpanan data.
3. Memberikan akses informasi kesehatan secara real-time kepada masyarakat melalui chatbot.
4. Mendukung peningkatan pengetahuan masyarakat terhadap isu stunting dan imunisasi anak.

Berdasarkan hasil pengujian yang dilakukan melalui metode Black Box Testing serta evaluasi pengguna menggunakan *User Acceptance Testing* (UAT) dengan skor kepuasan mencapai 75,83%, sistem yang dikembangkan terbukti mampu meningkatkan efisiensi layanan Posyandu. Aplikasi telah memfasilitasi pencatatan data secara digital dan mempercepat penyampaian informasi kepada pengguna.

Meskipun fitur chatbot masih memiliki keterbatasan dalam hal akurasi dan konteks respons, fungsinya sebagai media konseling interaktif telah berjalan sesuai harapan. Chatbot mampu memberikan jawaban atas berbagai pertanyaan kesehatan dasar, terutama terkait stunting dan imunisasi.

Dengan mempertimbangkan hasil-hasil tersebut, maka dapat disimpulkan bahwa hipotesis alternatif (H1), yaitu “penggunaan aplikasi berbasis *React Native* dengan chatbot AI dapat meningkatkan efisiensi pencatatan dan penyampaian informasi di Posyandu”, dapat diterima. Sebaliknya, hipotesis nol (H0) yang menyatakan tidak adanya peningkatan efisiensi ditolak, karena aplikasi ini terbukti mampu memberikan manfaat nyata dibandingkan metode manual yang digunakan sebelumnya.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Penelitian ini menghasilkan aplikasi Posyandu berbasis mobile yang dikembangkan menggunakan framework *React Native* dengan tambahan fitur chatbot berbasis AI. Aplikasi dirancang untuk mendukung proses digitalisasi kegiatan Posyandu, terutama dalam hal pencatatan data balita, pemantauan tumbuh kembang, serta konsultasi kesehatan.

Berdasarkan hasil pengujian sistem menggunakan *Black Box Testing*, seluruh fitur utama aplikasi, seperti login berdasarkan peran, pengelolaan data posyandu, monitoring tumbuh kembang anak, dan fitur ekspor data ke Excel, dapat berjalan dengan baik dan sesuai dengan kebutuhan yang dirancang.

Sementara itu, dari hasil *User Acceptance Test* (UAT) yang melibatkan 10 kader Posyandu, diperoleh total skor 455 dari skor ideal 600, atau 75,83%, yang berada pada kategori “Setuju”. Hal ini menunjukkan bahwa secara umum, aplikasi telah diterima dengan baik oleh pengguna dan dianggap sesuai dengan kebutuhan Posyandu.

Namun, khusus untuk fitur chatbot, ditemukan bahwa sebagian besar pengguna memberikan nilai netral hingga tidak setuju, baik dari segi akurasi jawaban maupun kecepatan respons. Temuan ini mengindikasikan bahwa meskipun secara teknis chatbot mampu memberikan jawaban berbasis pencocokan teks menggunakan metode TF-IDF dan *Cosine Similarity*, namun dari sisi pengguna, kualitas jawabannya belum sepenuhnya memuaskan. Hal ini juga didukung oleh pengujian simulatif internal yang menunjukkan bahwa chatbot masih terbatas dalam menjawab pertanyaan dengan kalimat tidak baku atau variasi bahasa alami.

Dengan demikian, dapat disimpulkan bahwa aplikasi yang dikembangkan telah berhasil memenuhi sebagian besar fungsi yang dibutuhkan oleh Posyandu secara digital, namun masih perlu dilakukan pengembangan lebih lanjut pada fitur chatbot, terutama dalam hal perluasan dataset dan peningkatan pemahaman konteks pertanyaan pengguna.

6.2 Saran

1. Pengembangan Dataset Chatbot.

Diperlukan penambahan dan pengayaan dataset pertanyaan dan jawaban yang lebih beragam agar chatbot mampu menjawab pertanyaan dengan variasi bahasa yang lebih luas dan kontekstual.

2. Evaluasi Ulang Respons Chatbot.

Perlu dilakukan evaluasi lebih lanjut terhadap jawaban yang dihasilkan chatbot, baik melalui uji semantik maupun melalui masukan pengguna secara berkelanjutan untuk memastikan relevansi dan akurasi informasi yang diberikan.

3. Integrasi NLP yang Lebih Canggih.

Sebagai pengembangan selanjutnya, sistem chatbot dapat diperkuat dengan model pemahaman bahasa alami (*natural language understanding*) yang lebih canggih, seperti menggunakan transformer-based models yang telah dilatih khusus untuk domain kesehatan masyarakat.

4. Peningkatan Pelatihan Pengguna.

Perlu disediakan panduan penggunaan chatbot atau pelatihan singkat kepada pengguna agar mereka dapat menyampaikan pertanyaan dengan struktur yang lebih sesuai dengan cara kerja chatbot saat ini.

5. Uji Pengguna Lebih Luas

Disarankan untuk melibatkan lebih banyak responden dari berbagai latar belakang pada UAT selanjutnya, agar dapat diperoleh gambaran evaluasi yang lebih representatif terhadap semua fitur aplikasi, khususnya chatbot.

DAFTAR PUSTAKA

- Anwar, F., Senung, B., Mulawati Mas Pratama, A., & Rahman Ismail, A. (2023). KLIK: Kajian Ilmiah Informatika dan Komputer Aplikasi Pos Pelayanan Terpadu (Posyandu) Berbasis Android Menerapkan Metode Research and Development. *Media Online*, 4(3), 1493–1500. <https://doi.org/10.30865/klik.v4i3.1292>
- Atmaji Perdana, R., Hindratno, M. N., M. Santoso, T., Try Harmanda, T., Resvy Mahda, F., Andreas Lapien, Y., Dwi W. S., M., Aldion Renata, D., Aji, P., Layooari, M., Supriatna, K., Kusumajaya, R., & Hamdani, M. (2023). Pengembangan Aplikasi Mobile Pengisian Daya Kendaraan Listrik Sesuai Standar OCPP 1.6. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 10(6), 1167–1176. <https://doi.org/10.25126/jtiik.20231056647>
- Fajar, J., Pesihatu, S., Rahmawati, E., Keperawatan, A., Florensia, A., Kunci, K., Gizi, S., Balita, ;, Kelamin, J., Berat, ;, & Lahir, B. (2022). *Humantech Jurnal Ilmiah Multi Disiplin Indonesia Status Gizi Balita*. 2(1).
- Farmani, P. I., Adiputra, I. N. M., & Laksmini, P. A. (2021). Perancangan Sistem Informasi Posyandu Sebagai Upaya Digitalisasi Data Posyandu di UPTD Puskesmas II Dinas Kesehatan Kecamatan Denpasar Timur. *Indonesian of Health Information Management Journal (INOHIM)*, 9(2), 115–126. <https://doi.org/10.47007/inohim.v9i2.311>
- Fatahila, D., Rahmawati, D., & Arlianti, L. (2023). Rancang Bangun Aplikasi Posyandu Berbasis Android Menggunakan UML (Studi Kasus: Posyandu Dahlia Perumahan Total Persada). *Jurnal Ilmiah Fakultas Teknik*, 3(2), 104–114. <https://doi.org/10.33592/jimtek.v3i2.3792>
- Gifari Aji, B., & Prasetyo, M. A. W. (2024). Perancangan Sistem Informasi Posyandu Berbasis Web Dengan Metode Extreme Programming Pada Desa Candinata. *Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)*, 9(1), 82–93. <https://tunasbangsa.ac.id/ejurnal/index.php/jurasik>
- Harry, S., Wijonarko, D., & Afif, H. (2024). Rancang Bangun Aplikasi Pengelolaan Keuangan Berbasis Android Menggunakan Framework React Native. *INFORMAL: Informatics Journal*, 8(3), 199. <https://doi.org/10.19184/isj.v8i3.44032>
- Hikmah, N. (2018). Pemanfaatan Text Mining dalam Pencarian Ayat AlQuran menggunakan TF-IDF dan Cosine Similarity. *Jurnal Antartika*, 8, 13–22. <http://www.sttar.ac.id>
- Nuzul Hikmah, Dyah Ariyanti, & Ferry Agus Pratama. (2022). Implementasi Chatbot Sebagai Virtual Assistant di Universitas Panca Marga Probolinggo menggunakan Metode TF-IDF. *JTIM: Jurnal Teknologi Informasi Dan Multimedia*, 4(2), 133–148. <https://doi.org/10.35746/jtim.v4i2.225>

- Prasetyo, D. Y., Yunita, F., & Setiawan, R. (2021). APLIKASI ePOSYANDU KESEHATAN IBU DAN ANAK DI KECAMATAN KEMPAS JAYA. *Jurnal Perangkat Lunak*, 3(3), 136–147. <https://doi.org/10.32520/jupel.v3i3.1738>
- Ramadhan Sukoco, G., Irawan, D., & Prabowo, D. (2022). Perancangan Sistem Informasi Posyandu Desa Tulus Rejo Berbasis Web. *Jurnal Mahasiswa Ilmu Komputer (JMIK)*, 03(01), 278–283.
- RI, P. (2020). Permenkes RI No. 2 Tahun 2020. *Permenkes RI No. 2 Tahun 2020*, 53(1), 1–9. <http://dx.doi.org/10.1016/j.biochi.2015.03.025><http://dx.doi.org/10.1038/nature10402><http://dx.doi.org/10.1038/nature21059><http://journal.stainkudus.ac.id/index.php/equilibrium/article/view/1268/1127><http://dx.doi.org/10.1038/nrmicro2577>
- Santoso, M. H., & Anggara, E. D. (2021). Rancang Bangun Aplikasi Mobile Sistem Informasi. *JIKA (Jurnal Informatika) Universitas Muhammadiyah Tangerang*, May, 235–244.
- Utami, R., Setianto, R., Dewi, B., & Maftukhin, A. (2023). Peningkatan Tinggi Badan Z Score TB/U pada Balita Stunting dengan Pemberian Zinc. *PREPOTIF : Jurnal Kesehatan Masyarakat*, 7(1), 756–760.
- Wati, R., & Istianah, I. (2023). Risk Factors Associated with Nutritional Status of Toddlers. *Jurnal Ilmiah Kesehatan (JIKA)*, 5(2), 396–404. <https://doi.org/10.36590/jika.v5i2.395>

LAMPIRAN