

# Pengembangan Backend Aplikasi Pemantauan Kartu Menuju Sehat (Kms) Pada Posyandu Menggunakan Metode Iterative Incremental

1<sup>st</sup> Daffa Rifki Maulana  
Fakultas Rekayasa Industri  
Universitas Telkom  
Bandung, Indonesia

[daffarifki@student.telkomuniversity.ac.id](mailto:daffarifki@student.telkomuniversity.ac.id)

2<sup>nd</sup> Faishal Mufied Al-Anshary  
Fakultas Rekayasa Industri  
Universitas Telkom  
Bandung, Indonesia

[faishalmufied@telkomuniversity.ac.id](mailto:faishalmufied@telkomuniversity.ac.id)

3<sup>rd</sup> Ahmad Musnansyah  
Fakultas Rekayasa Industri  
Universitas Telkom  
Bandung, Indonesia

[ahmadanc@telkomuniversity.ac.id](mailto:ahmadanc@telkomuniversity.ac.id)

**Abstrak** — Kemajuan teknologi informasi memberikan dampak positif dalam berbagai sektor, termasuk layanan kesehatan. Salah satu contohnya adalah Posyandu, yang berfungsi sebagai pusat pelayanan kesehatan masyarakat di tingkat desa atau kelurahan di Indonesia. Di Posyandu, Kartu Menuju Sehat (KMS) digunakan untuk memantau perkembangan dan pertumbuhan balita serta memberikan layanan kesehatan yang sesuai. Namun, masih terdapat tantangan dalam penggunaan KMS, seperti penggunaan manual yang menyulitkan pengelolaan data dan pencatatan yang sering kali rentan terhadap kesalahan. Penelitian ini bertujuan mengatasi masalah tersebut dengan merancang dan mengembangkan aplikasi mobile berbasis Flutter yang berperan sebagai alat pemantau KMS di Posyandu. Metode pengembangan yang digunakan adalah Iterative Incremental, yang memungkinkan perkembangan bertahap dan peningkatan berdasarkan umpan balik pengguna. Aplikasi ini dilengkapi dengan fitur pemantauan pertumbuhan balita, memungkinkan ibu balita dengan mudah memasukkan data pertumbuhan dan melihat grafik perkembangannya. Bagian backend aplikasi akan digunakan untuk menyimpan dan mengelola data pertumbuhan balita, serta menyediakan logika bisnis yang mendukung fungsi aplikasi. Hasil dari penelitian ini adalah aplikasi backend yang mampu menyimpan dan mengelola data pertumbuhan balita, membantu dalam pemantauan KMS di Posyandu, meningkatkan partisipasi ibu balita, dan memberikan informasi kesehatan yang akurat dan tepat waktu. Dengan demikian, aplikasi ini diharapkan memberikan kontribusi positif dalam meningkatkan kualitas layanan kesehatan di Posyandu.

**Kata kunci**— Posyandu, KMS, Backend, Iterative Incremental, Flutter.

## I. PENDAHULUAN

Dalam upaya meningkatkan kualitas kesehatan masyarakat, layanan Posyandu telah menjadi salah satu pendekatan yang sangat penting [1]. Posyandu adalah salah satu bentuk usaha kesehatan berbasis masyarakat (UKBM) yang dijalankan oleh masyarakat, untuk masyarakat, dan bersama masyarakat dalam rangka memajukan pembangunan kesehatan. Tujuannya adalah memberdayakan masyarakat dan memudahkan akses mereka terhadap layanan kesehatan dasar atau sosial guna mengurangi angka kematian ibu dan bayi [1]. Posyandu adalah lokasi pelayanan kesehatan yang ada di desa atau kelurahan dengan fokus pada pemantauan perkembangan dan pertumbuhan anak balita. Dalam hal ini, Kartu Menuju Sehat (KMS) menjadi sarana penting untuk

mencatat dan memantau data kesehatan anak. Kartu Menuju Sehat (KMS) adalah kartu yang mengandung grafik pertumbuhan normal anak dalam aspek antropometri berdasarkan berat badan sesuai usia [2]. Menggunakan KMS, masalah pertumbuhan yang terganggu atau potensi risiko kelebihan gizi dapat teridentifikasi lebih awal. Ini memungkinkan tindakan pencegahan dilakukan lebih cepat dan akurat sebelum masalah tersebut memburuk [3].

Namun, meskipun memiliki potensi besar untuk memberikan informasi berharga, penggunaan KMS di banyak Posyandu masih menghadapi beberapa tantangan. Salah satu kendala yang sering ditemui adalah proses pengumpulan, pengelolaan, dan analisis data yang masih bersifat manual seperti yang terjadi di Posyandu Melati 08 yang terletak di Desa Citereup, Kabupaten Bandung. Penginputan data manual ini tidak hanya berisiko terhadap kesalahan, tetapi juga menghambat efisiensi serta akurasi dalam memantau pertumbuhan dan perkembangan balita. Berdasarkan studi yang dilakukan oleh Saepudin, ditemukan bahwa pengetahuan tentang kesehatan ibu dan anak yang dimiliki oleh kader Posyandu masih cukup rendah [4]. [5] menyatakan di dalam jurnalnya bahwa hasil studi yang dilakukan oleh Hartati, Kusumawardani, Suhartika, dan Rahmad menunjukkan bahwa tingkat keterampilan dan pemahaman kader dalam mengamati dan mengartikan hasil penimbangan KMS masih kurang memadai.

Dalam konteks ini, perkembangan teknologi memiliki peran yang sangat penting. Terutama dalam bidang kesehatan, kemajuan Teknologi Informasi telah memberikan dukungan yang besar terhadap layanan. Hal ini lebih krusial dalam dunia medis, di mana pengetahuan terus berkembang pesat (sekitar 750.000 artikel baru dalam jurnal kedokteran diterbitkan setiap tahun), sehingga para profesional medis, termasuk kader posyandu, akan cepat tertinggal jika tidak menggunakan berbagai alat untuk mengikuti perkembangan terbaru. Selain itu, teknologi informasi juga memiliki kemampuan untuk menyaring data dan mengubahnya menjadi informasi yang berarti [6]. Dari sudut pandang teori dan konseptual, juga diakui bahwa pelayanan publik yang optimal memiliki peran kunci dalam mendorong kemajuan dan meningkatkan daya saing. Oleh karena itu, sangat penting bagi sektor pelayanan untuk terus meningkatkan kualitasnya [6]. Pengembangan *backend* aplikasi untuk pemantauan KMS pada Posyandu memiliki potensi untuk mengatasi berbagai masalah tersebut. *Backend* adalah sebuah

perangkat lunak yang beroperasi di sisi server (*server-side*) yang bertugas untuk berhubungan langsung dengan basis data, melakukan pengolahan data pada basis data, dan karena itu, *backend* tidak berinteraksi secara langsung dengan pengguna [7]. Dengan memanfaatkan *backend*, data pertumbuhan dan perkembangan balita dapat diakses dan dikelola secara lebih efisien serta akurat.

Pengembangan *backend* pada penelitian ini dilakukan dengan memanfaatkan sistem manajemen basis data yaitu MySQL. Dalam konteks ini, pengembangan *backend* MySQL pada aplikasi pemantauan KMS di Posyandu memiliki potensi besar untuk mengatasi masalah pengumpulan, penyimpanan, dan analisis data pada KMS. MySQL sebagai sistem manajemen basis data relasional mampu menyediakan struktur yang terstruktur bagi data KMS. Ini berarti bahwa data yang diatur dalam basis data akan ditempatkan dalam beberapa tabel yang terpisah, memungkinkan manipulasi data menjadi lebih efisien [8]. Dengan adanya *backend* MySQL yang terintegrasi, data pertumbuhan dan perkembangan balita dapat diakses, dikelola, dan dianalisis dengan lebih efisien.

Dalam pengembangan ini, metode *Iterative Incremental* diadopsi untuk memastikan fleksibilitas dan responsifitas terhadap perubahan kebutuhan pengguna serta pengembangan yang terarah. Dalam konteks ini, *Iterative Incremental Development* merujuk pada seperangkat praktik dalam membangun perangkat lunak di mana keseluruhan *lifecycle* terdiri dari beberapa iterasi, di mana setiap iterasi secara bertahap mengembangkan sistem yang dihasilkan. Setiap iterasi terdiri dari aktivitas seperti analisis kebutuhan, desain, pemrograman, dan pengujian [9]. Dengan merancang sebuah *backend* melalui metode ini dan terintegrasi dengan baik, sistem pemantauan KMS di Posyandu dapat menjadi lebih efektif, efisien, dan memberikan dampak positif terhadap pelayanan kesehatan masyarakat secara keseluruhan.

## II. KAJIAN TEORI

Bab kajian teori ini berfokus pada paparan temuan studi pustaka terkait dengan permasalahan, kerangka kerja, dan metode yang relevan dengan topik penelitian. Pada bagian ini juga dijelaskan berbagai hal yang dapat memberikan pemahaman yang lebih mendalam tentang topik penelitian, termasuk Posyandu, Flutter, Bahasa Pemrograman Dart, Visual Studio Code, Xampp, dan *Unit Testing*.

### A. Posyandu

Pos Pelayanan Terpadu (Posyandu) adalah bentuk upaya yang dilakukan bersama masyarakat, untuk mendorong pemberdayaan dan mempermudah masyarakat dalam mendapatkan layanan kesehatan untuk ibu dan anak balita [10]. Keberadaan Posyandu memiliki dampak yang signifikan terhadap kesehatan ibu dan anak, karena dapat berkontribusi dalam mempercepat penurunan Angka Kematian Ibu (AKI), Angka Kematian Bayi (AKB), serta Angka Kematian Anak Balita (AKABA) melalui upaya pemberdayaan masyarakat [11]. Dalam Pedoman Pengintegrasian Layanan Sosial di Pos Pelayanan Terpadu sesuai Peraturan Menteri Dalam Negeri Nomor 19 Tahun 2011, disebutkan bahwa posyandu melibatkan tujuh kegiatan utama. Ini mencakup pendaftaran, penimbangan, pencatatan, pelayanan kesehatan, penyuluhan kesehatan, serta upaya

untuk meningkatkan keragaman dan ekonomi keluarga. Semua kegiatan ini ditargetkan pada keluarga yang memiliki anak berusia 0-6 tahun. Dengan peran yang aktif, kader menjadi inisiator dan pelaksana layanan di posyandu, yang merupakan salah satu upaya kesehatan berbasis masyarakat. Kader Posyandu adalah individu yang ditugaskan untuk mengelola layanan posyandu di suatu desa karena memiliki pengetahuan dan kemampuan dalam kesehatan masyarakat, terutama terkait ibu dan anak.

### B. Kartu Menuju Sehat

Kartu Menuju Sehat (KMS) merupakan kartu yang memuat grafik pertumbuhan standar anak berdasarkan indeks antropometri berat badan sesuai dengan usia [2]. Dengan adanya KMS, gangguan pertumbuhan atau potensi risiko kelebihan gizi dapat diidentifikasi lebih awal, sehingga langkah-langkah pencegahan dapat diambil dengan cepat dan tepat sebelum situasinya menjadi lebih serius [2]. Secara keseluruhan, peran KMS adalah sebagai alat pengawasan pertumbuhan, pencatatan layanan kesehatan anak, dan juga sebagai sarana edukasi [2].

### C. Iterative Incremental Model

*Iterative Incremental* adalah gabungan dua kata, yakni "*Iterative*" yang mengacu pada suatu pendekatan pengembangan di mana bagian-bagian dari sistem dikembangkan secara terpisah dalam tahapan dan skala yang berbeda [12]. Sementara itu, "*Incremental*" merujuk pada pendekatan pengembangan sistem yang melibatkan perulangan untuk melakukan revisi atau penyempurnaan dari tahap-tahap pengembangan sistem [12]. Tahapan utama dalam proses ini melibatkan langkah awal dengan menerapkan versi sederhana dari sebagian persyaratan perangkat lunak, dan kemudian secara berulang meningkatkan urutan versi tersebut hingga sistem secara keseluruhan berhasil diimplementasikan [13]. Dalam setiap siklus iterasi, dilakukan perubahan desain bersama dengan penyertaan kemampuan fungsional baru [13]. Di dalam jurnalnya [14] menyatakan bahwa menurut (Rational Software Development Company, 1998) seperti yang dijelaskan dalam *Best Practices For Software Development Time*, Metode *Iterative Incremental* memiliki siklus pengembangan yang terdiri dari empat tahap pokok, yakni *Inception*, *Elaboration*, *Construction*, dan *Transition*. Masing-masing siklus ini mencakup satu atau beberapa iterasi.

### D. Application Program Interface (API)

*Application Programming Interface* (API) adalah antarmuka teknologi yang dibuat oleh pengembang sistem untuk memungkinkan akses baik sebagian maupun keseluruhan fungsi sistem secara efisien [15]. Secara umum, API merupakan representasi yang terfokus dari semua fungsionalitas dalam sebuah modul perangkat lunak yang dapat diakses oleh pihak yang membutuhkannya melalui layanan yang telah ditentukan [15]. API juga memiliki kemampuan untuk memberikan mekanisme ekstensibilitas, yang memungkinkan pengguna untuk memperluas fungsionalitas yang ada dalam berbagai cara dan tingkat yang berbeda [15].

#### E. *Representatif State Transition API (RESTful API)*

REST, singkatan dari Representational State Transfer, merupakan standar arsitektur komunikasi yang umumnya digunakan dalam pembangunan situs web dan layanan berbasis aplikasi. Di sisi lain, API adalah suatu antarmuka yang memungkinkan aplikasi untuk berinteraksi dan bertukar data [16]. REST adalah sekumpulan aturan arsitektur yang memfasilitasi pengiriman data melalui antarmuka yang telah terstandarisasi, seperti HTTP [16]. Dalam kerangka arsitektur REST, server REST menyediakan sumber daya atau data, dan klien REST mengakses serta menampilkan sumber daya tersebut untuk penggunaannya [16]. Tiap sumber daya diidentifikasi melalui URI (*Universal Resource Identifiers*) atau ID global. Sumber daya ini direpresentasikan dalam format teks, JSON, atau XML [16]. Keunikan dari REST terletak pada cara klien dan server berinteraksi, yang dipermudah oleh berbagai jenis operasi (verba) dan *Universal Resource Identifiers* (URIs) yang khas untuk setiap sumber daya. Setiap verba seperti *GET*, *POST*, *PUT*, dan *DELETE* memiliki makna operasional yang spesifik untuk menghindari kebingungan [17].

#### F. *JavaScript Object Notation (JSON)*

JSON adalah sebuah skema untuk pertukaran data. Meskipun berasal dari bahasa pemrograman JavaScript, JSON dapat digunakan dalam berbagai bahasa lain seperti Python, Ruby, PHP, dan Java. JSON biasanya memiliki ekstensi .json saat digunakan sendiri. Ketika diintegrasikan ke dalam format file lain (seperti dalam file .html), data JSON dapat diwakili dalam tanda kutip sebagai string JSON, atau dimasukkan ke dalam variabel. Format ini sangat mudah untuk dikirimkan antara server web dan klien atau peramban [16]. JSON memiliki dua jenis struktur [17], yaitu:

##### 1. Kombinasi antara nama dan nilai.

Dalam beberapa bahasa pemrograman, ini dikenal sebagai objek, rekaman, struktur, kamus, daftar yang memiliki kunci, atau *array* asosiatif.

##### 2. Kumpulan nilai yang diurutkan (urutan nilai yang tidak terurut).

Dalam kebanyakan bahasa, ini diartikan sebagai *array*, vektor, daftar, atau urutan.

#### G. *Unified Modeling Language (UML)*

UML (*Unified Modeling Language*) merupakan suatu bentuk bahasa yang berfokus pada gambar atau diagram untuk menggambarkan, mendefinisikan, membangun, dan mendokumentasikan sistem pengembangan perangkat lunak yang berbasis *Object-Oriented* (berorientasi objek) [18]. UML juga menetapkan aturan baku untuk merancang skema dasar sebuah sistem, termasuk menggambarkan proses bisnis, menulis kode kelas dalam bahasa pemrograman tertentu, mengatur struktur basis data, serta komponen-komponen esensial yang diperlukan dalam perangkat lunak sistem [18]. UML adalah sebuah pola perencanaan sistem yang memiliki keunggulan dalam membantu pengembang sistem dalam merencanakan sistem yang akan dikembangkan, karena pendekatannya yang berfokus pada objek [19].

#### H. *MySQL*

MySQL adalah sebuah perangkat lunak yang termasuk dalam sistem manajemen data SQL (*database management system*) atau DBMS. Ini adalah sistem multithread dan *multi-*

*user* dengan lebih dari 6 juta instalasi global [20]. MySQL adalah sistem manajemen *database* yang bersifat relasional [8]. Ini berarti bahwa data yang dikelola dalam *database* akan ditempatkan pada tabel-tabel terpisah, memungkinkan manipulasi data menjadi lebih cepat [8]. Dalam perkembangannya, MySQL juga sering disebut sebagai SQL, singkatan dari *Structured Query Language* [8].

#### I. *Flutter*

Flutter adalah framework lintas platform yang dirilis oleh Google dan memiliki target untuk mengembangkan aplikasi berbasis *mobile* dengan performansi yang tinggi [1]. Flutter pertama kali dirilis secara publik pada tahun 2016 oleh Google. Selain berjalan di Android dan iOS, aplikasi Flutter juga dapat berjalan di sistem operasi Fuchsia [1]. Dalam Flutter, setiap aplikasi dikodekan menggunakan bahasa pemrograman Dart. Dart adalah bahasa pemrograman yang dikembangkan oleh Google sebagai pengganti bahasa JavaScript [1]. Pembuatan antarmuka pengguna pada Flutter sepenuhnya bergantung pada konsep *widget*. Sebagai contoh, terdapat *widget* seperti *RaisedButton*, *ListView*, *DateTimePicker*, *TabBar*, *Text*, *Label*, dan berbagai *widget* lainnya [2].

#### J. *Bahasa Pemrograman Dart*

Google telah menciptakan sebuah bahasa pemrograman yang dikenal sebagai Dart. Bahasa pemrograman ini memiliki penggunaan yang luas di dalam lingkungan Google dan telah terbukti mampu mengembangkan aplikasi web besar, termasuk contohnya AdWords [1]. Dart menggunakan pendekatan *static typing*, di mana variabel harus dideklarasikan sebelum digunakan [2]. Awalnya, Dart dirancang untuk menjadi alternatif bagi JavaScript [1]. Dalam konteks ini, bahasa pemrograman ini menggabungkan banyak fitur penting yang ditemukan pada standar JavaScript berikutnya (ES7), seperti kata kunci "*async*" dan "*await*". Namun, untuk mendorong adaptasi oleh para pengembang yang belum terbiasa dengan JavaScript, Dart dirancang dengan sintaks yang lebih mirip dengan bahasa pemrograman Java [1].

#### K. *Visual Studio Code*

Visual Studio Code adalah sebuah aplikasi pengedit kode yang bersifat *open source* yang dirancang oleh Microsoft untuk berjalan pada sistem operasi Windows, Linux, dan MacOS. Aplikasi ini memiliki kemampuan yang memudahkan pengguna dalam menulis kode dan mendukung beragam jenis bahasa pemrograman seperti C++, C#, Java, Python, PHP, dan GO [3]. Satu dari kelebihanannya adalah kapasitasnya dalam mengidentifikasi bahasa pemrograman yang sedang digunakan dan memberikan pewarnaan yang beragam sesuai dengan fungsi dalam susunan kode tersebut [3]. Tidak hanya itu, Visual Studio Code telah terhubung dengan Github. Aplikasi ini juga menawarkan opsi untuk diperluas, di mana para pengembang dapat menambahkan ekstensi untuk meningkatkan fungsi yang tidak ada di dalam Visual Studio Code [3].

#### L. *Xampp*

XAMPP memiliki peran sebagai server yang berdiri sendiri di lokal (*localhost*), yang terdiri dari beberapa komponen seperti Apache HTTP Server, basis data MySQL,

dan interpreter bahasa seperti PHP dan Perl [4]. XAMPP singkatan dari X (merekpresentasikan empat sistem operasi berbeda), Apache, MySQL, PHP, dan Perl [4]. Perangkat lunak ini dapat diunduh dan digunakan secara gratis sesuai dengan lisensi GNU (*General Public License*) [4]. XAMPP adalah *server web* yang sederhana dalam penggunaannya dan mampu menampilkan halaman *web* yang dinamis. Pengguna dapat mengunduh XAMPP secara langsung melalui situs *web* resmi yang disediakan [4].

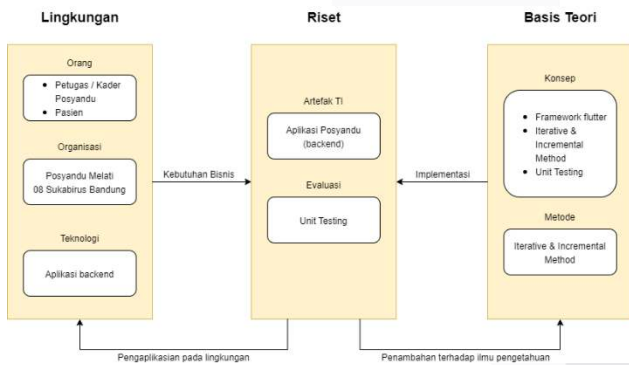
M. *Postman*

Postman adalah alat yang sangat berguna bagi para pengembang yang terlibat dalam pembuatan API. Fungsinya utama adalah sebagai GUI untuk memanggil API, tetapi saat ini Postman juga menyediakan berbagai fitur lainnya, seperti berbagi koleksi API untuk dokumentasi (gratis), menguji API (gratis), kolaborasi tim secara *real-time* (berbayar), pemantauan API (berbayar), dan integrasi (berbayar) [21]. *Tool* ini bertugas untuk memeriksa bahwa kualitas dan kehandalan API dapat berjalan efektif [22].

N. *Unit Testing*

Perkembangan pemrograman berorientasi objek telah mempengaruhi strategi pengujian yang digunakan oleh para pengembang perangkat lunak. Pengujian unit melibatkan pengujian terhadap "unit" kode secara individu dan membandingkan hasil yang sebenarnya dengan hasil yang diharapkan. Dalam konteks bahasa pemrograman Java, unit ini biasanya merujuk pada kelas. Pengujian unit ini melibatkan pemanggilan satu atau lebih metode dari kelas tersebut untuk menghasilkan hasil yang dapat diamati dan dapat divalidasi secara otomatis [5].

III. METODE

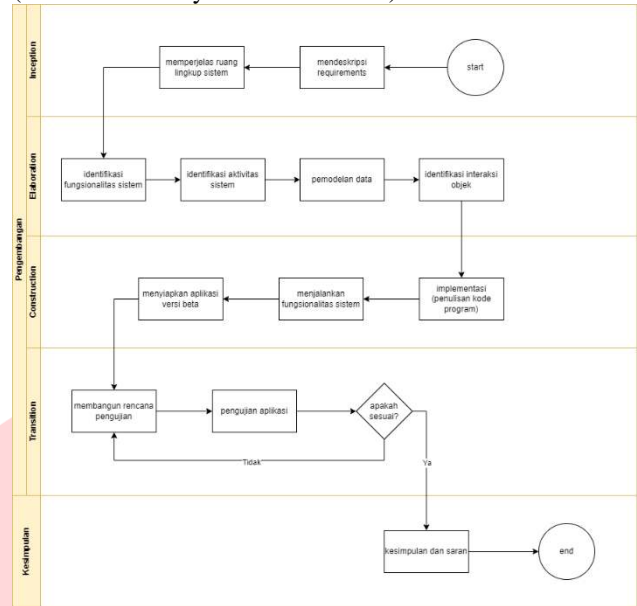


GAMBAR 1 (Kerangka Berpikir)

Penelitian ini dilakukan di lingkungan organisasi Posyandu Melati 08 Sukaburus Bandung dengan tujuan untuk digunakan oleh petugas posyandu dan ibu balita, sebagaimana terlihat pada gambar 1 (Kerangka Berpikir). Dalam penelitian ini, digunakan basis teori berupa framework Flutter, Metode Iterative Incremental, dan Unit Testing. Selanjutnya, untuk mengevaluasi aplikasi, akan dilakukan pengujian menggunakan metode unit testing.

Dalam penelitian ini, diperlukan sebuah metodologi untuk mengatur jalannya penelitian agar menjadi lebih terstruktur, teratur, dan sesuai dengan tujuan yang telah ditentukan. Penelitian ini dibagi menjadi empat tahapan, yaitu: *inception*, *elaboration*, *construction*, dan *transition*.

Informasi lebih rinci mengenai metode penyelesaian masalah pada penelitian ini dapat ditemukan pada gambar 2 (Sistematika Penyelesaian Masalah).



GAMBAR 2 (SISTEMATIKA PENYELESAIAN MASALAH)

IV. HASIL DAN PEMBAHASAN

A. Fase Inception

Hasil dari fase awal (*inception*) pada setiap iterasi adalah analisis dari kebutuhan pengguna aplikasi, sebagaimana dijelaskan dalam Tabel 1.

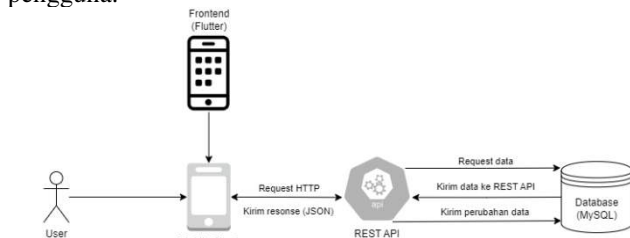
TABEL 1 (KEBUTUHAN FUNGSIONAL SISTEM)

ID	Grup Kebutuhan	Kebutuhan	Deskripsi
REQ-01.01	Autentikasi	<i>Login</i>	Proses mengakses <i>endpoint</i> pada suatu <i>session</i> dengan menggunakan token API
REQ-01.02		<i>Register Akun User</i>	Proses mendaftarkan akun <i>user</i> ke dalam aplikasi
REQ-01.03		<i>Change Password</i>	Proses perubahan <i>password user</i>
REQ-01.04		<i>Logout</i>	Proses menghilangkan <i>session</i> yang sedang aktif
REQ-02.01	Presensi Ibu Balita	<i>Add Presensi</i>	Proses pendaftaran <i>user</i> untuk pemeriksaan
REQ-02.02		<i>Get Presensi</i>	Proses mendapatkan data <i>user</i> yang

			sudah terdaftar pemeriksaan				ditampilkan pada <i>Homepage</i>
REQ-02.03		<i>Update Presensi</i>	Proses <i>Update Presensi</i> yang sudah diinputkan	REQ-06.02		<i>Get Jadwal Pemeriksaan</i>	Proses petugas posyandu mendapatkan data jadwal pemeriksaan yang akan ditampilkan pada <i>Homepage</i>
REQ-02.04		<i>Delete Presensi</i>	Proses menghapus data <i>user</i>				
REQ-03.01	Data Penimbangan	<i>Add Data Penimbangan</i>	Proses petugas posyandu menambahkan data penimbangan	REQ-06.03		<i>Get Status Imunisasi</i>	Proses petugas posyandu mendapatkan Status Imunisasi <i>user</i> yang akan ditampilkan pada <i>Homepage</i>
REQ-03.02		<i>Get Data Penimbangan</i>	Proses petugas posyandu mendapatkan data penimbangan				
REQ-04.01	Jadwal Pemeriksaan	<i>Add Jadwal Pemeriksaan</i>	Proses petugas posyandu menambahkan jadwal pemeriksaan	REQ-07.01	<i>Homepage Ibu Balita</i>	<i>Get Data Penimbangan</i>	Proses <i>user</i> mendapatkan data penimbangan yang akan ditampilkan pada <i>Homepage</i>
REQ-04.02		<i>Get Jadwal Pemeriksaan</i>	Proses petugas posyandu mendapatkan jadwal pemeriksaan	REQ-07.02		<i>Get Jadwal Pemeriksaan</i>	Proses <i>user</i> mendapatkan data jadwal pemeriksaan yang akan ditampilkan pada <i>Homepage</i>
REQ-04.03		<i>Update Jadwal Pemeriksaan</i>	Proses petugas posyandu mengubah jadwal pemeriksaan				
REQ-04.04		<i>Delete Jadwal Pemeriksaan</i>	Proses petugas posyandu menghapus data jadwal pemeriksaan	REQ-07.03		<i>Get Status Imunisasi</i>	Proses <i>user</i> mendapatkan Status Imunisasi yang akan ditampilkan pada <i>Homepage</i>
REQ-05.01	Riwayat Imunisasi	<i>Set Status Imunisasi</i>	Proses petugas posyandu menambahkan Status Imunisasi	REQ-08.01	<i>User Profile</i>	<i>Get Data User</i>	Proses menampilkan data <i>user</i> yang sedang aktif digunakan
REQ-05.02		<i>Get Status Imunisasi</i>	Proses <i>user</i> mendapatkan Status Imunisasi apa saja yang sudah dan belum dilakukan	REQ-08.02		<i>Update Data User</i>	Proses mengubah data <i>user</i>
REQ-05.03		<i>Update status imunisasi</i>	Proses petugas posyandu mengupdate status imunisasi	REQ-09.01	Data Anak	<i>Add Data Anak</i>	Proses menambahkan data anak pada akun ibu balita
REQ-06.01	<i>Homepage Petugas Posyandu</i>	<i>Get Data Penimbangan</i>	Proses petugas posyandu mendapatkan data penimbangan yang akan	REQ-09.02		<i>Get Data Anak</i>	Proses mengakses dan menampilkan data anak
				REQ-09.03		<i>Update Data Anak</i>	Proses mengubah data anak
				REQ-09.04		<i>Delete Data Anak</i>	Proses menghapus data anak

B. Fase Elaboration

Pada tahap ini, hasil utama yang dihasilkan adalah desain dalam bentuk model berdasarkan hasil analisis kebutuhan pengguna.



GAMBAR 3 (Rancangan Sistem Aplikasi)

C. Fase Construction

Pada tahap ini, kode program aplikasi dibangun sesuai dengan desain yang telah disusun sebelumnya. Output dari aplikasi didokumentasikan melalui Dokumentasi API. Dokumentasi API ini berfungsi untuk memberikan panduan kepada pengguna mengenai cara melakukan permintaan (request) dan menerima tanggapan (response) dari aplikasi backend API Rumanta. Berikut adalah contoh dokumentasi API yang diimplementasikan, seperti yang terlihat pada Gambar 4:

HTTP Request

Tabel V.1 HTTP Request Login

Environment	
Method	POST
Endpoint	api_posyandu_apps/user/login.php
Header	
Accept	application/json
Body	
nik	352
password	123

HTTP Response

```

    {
      "success": "Berhasil login",
      "data": {
        "id_user": "1",
        "nik": "352",
        "nama": "admin4",
        "alamat": "Bandung",
        "password": "202cb962ac59075b964b07152d234b70",
        "role": "Admin"
      }
    }
  
```

GAMBAR 4 (Implementasi Backend)

D. Transition

Bagian ini menggambarkan tahap transisi, di mana pengujian aplikasi dilakukan melalui uji unit. Berikut ini adalah hasil dari beberapa uji yang dilakukan, yang dijelaskan di bawah ini:

TABEL 2 (Unit Testing 01)

Nomor test	Unit Testing 01
Fungsi	Login
Teknik	<pre>           Run   Debug           test('login', () async {             String dummynik = '352';             String dummypassword = '123';             bool success = await sourceUser.login(dummynik, dummypassword);             expect(success, true);           });         </pre>

Hasil	<pre>           ✓ Unit Testing : Login           Exited (1)         </pre>
-------	--

TABEL 3 (Unit Testing 02)

Nomor test	Unit Testing 02
Fungsi	Register Akun User
Teknik	<pre>           Run   Debug           test('Register Akun User', () async {             String dummyNik = '358';             String dummyNama = 'admin7';             String dummyAlamat = 'Bandung';             String dummyPassword = '123';             String dummyRole = 'Admin';              bool success = await SourceUser.register(               dummyNik,               dummyNama,               dummyAlamat,               dummyPassword,               dummyRole,             );              expect(success, true);           });         </pre>
Hasil	<pre>           ✓ Unit Testing : Register Akun User           Exited         </pre>

TABEL 4 (Unit Testing 03)

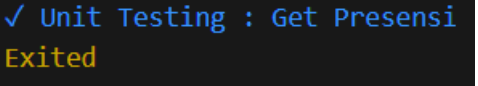
Nomor test	Unit Testing 03
Fungsi	Change Password
Teknik	<pre>           Run   Debug           test('Change Password', () async {             String dummyIdUser = '1';             String dummyPassword = '123';              bool success =               await SourceUser.changePassword(dummyIdUser, dummyPassword);              expect(success, true);           });         </pre>
Hasil	<pre>           ✓ Unit Testing : Change Password           Exited         </pre>

TABEL 5 (Unit Testing 04)

Nomor test	Unit Testing 04
Fungsi	Add Presensi
Teknik	<pre>           Run   Debug           test('Add Presensi', () async {             String dummyIdUser = '3';             String dummyIdKegiatan = '1';              bool success = await SourceUser.addPresensi(dummyIdUser, dummyIdKegiatan);              expect(success, true);           });         </pre>
Hasil	<pre>           ✓ Unit Testing : Add Presensi           Exited         </pre>

TABEL 6 (Unit Testing 05)

Nomor test	Unit Testing 05
------------	-----------------

Fungsi	Get Presensi
Teknik	<pre> Run   Debug test('Get Presensi', () async {   String dummyIdKegiatan = '1';    final presensi = await SourceUser.presensi(dummyIdKegiatan);    expect(presensi.isNotEmpty, true); }); </pre>
Hasil	

## V. KESIMPULAN

Berdasarkan hasil dari penelitian serta analisis yang dilakukan oleh penulis maka ada kesimpulan yang dapat diambil:

Pengembangan *backend* pada aplikasi berbasis mobile dengan menggunakan *framework* Flutter memiliki banyak keuntungan yang dirasakan oleh penulis. Salah satunya adalah kemudahan dalam proses pengembangan karena terdapat fitur-fitur yang membantu dalam mengintegrasikan dengan *back-end* secara lebih cepat. Contohnya, adanya *hot reload* memungkinkan penulis untuk melihat perubahan secara real-time tanpa harus me-restart aplikasi berulang kali. Fitur ini tidak hanya mempercepat proses pengembangan, tetapi juga memudahkan dalam menguji dan mengimplementasikan perubahan. Selain itu, Flutter menyediakan banyak *plugin* dan *package* yang mendukung integrasi dengan *back-end* dan layanan lainnya, seperti *package* untuk HTTP *requests*, state management, dan manajemen database. Dengan adanya *package* ini, mengintegrasikan aplikasi dengan *back-end* menjadi lebih mudah dan efisien.

Penerapan *Iterative Incremental* dalam mengembangkan *backend* aplikasi pemantauan kartu menuju sehat memberikan berbagai keuntungan dan manfaat yang berdampak positif pada proses pengembangan aplikasi. Metode ini memungkinkan penulis untuk membangun aplikasi secara bertahap dengan peningkatan yang berkelanjutan. Dalam setiap iterasi, hasil yang dapat diukur diperoleh. Hal ini memungkinkan penulis dan pemangku kepentingan untuk memberikan tanggapan lebih awal terhadap setiap perubahan atau penambahan, meminimalkan risiko arah yang salah.

Dengan mengulang-ulang tahap pengembangan, kesempatan untuk memperbaiki dan meningkatkan aplikasi semakin terbuka. Hal ini memungkinkan solusi yang lebih baik dan penanganan masalah yang lebih baik seiring berjalannya waktu. Penulis juga dapat dengan mudah menyesuaikan perubahan kebutuhan yang muncul dari pemangku kepentingan selama proses pengembangan. Hal ini membantu dalam menghasilkan solusi yang lebih sesuai dengan harapan dan kebutuhan pengguna.

Dengan menggunakan *unit testing* sebagai metode pengujian aplikasi, penulis dapat mendeteksi dan memperbaiki bug atau kesalahan lebih awal dari setiap bagian kode (*unit*) sebelum mereka menyebar ke bagian lain dari aplikasi.

## REFERENSI

- [1] Departemen Kesehatan R.I, *Pedoman Umum Pengelolaan Posyandu*. Jakarta: Departemen Kesehatan RI, 2006.
- [2] Romzah, Y. Eka Wibawa, and P. Dwi Larasati, "Pembangunan Sistem Informasi Kartu menuju Sehat (KMS) Balita Berbasis WEB Studi Kasus: Posyandu KASIH BUNDA II," 2021.
- [3] A. I. Susanti, F. R. Rinawan, and I. Amelia, "Mothers Knowledge and Perception of Toddler Growth Monitoring Using iPosyandu Application," *Global Medical & Health Communication (GMHC)*, vol. 7, no. 2, Aug. 2019, doi: 10.29313/gmhc.v7i2.3892.
- [4] E. Saepudin, E. Rizal, and A. Rusman, "Peran Posyandu Sebagai Pusat Informasi Kesehatan Ibu dan Anak Posyandu Roles as Mothers and Children Health Information Center," 2017.
- [5] F. Agiwahyunto and D. Ernawati, "Analisis Literasi Kartu Menuju Sehat terhadap Peningkatan Kualitas dan Mutu Kader Posyandu," 2021, doi: 10.15294/higeia/v5i1/38811.
- [6] A. Yani, "PEMANFAATAN TEKNOLOGI DALAM BIDANG KESEHATAN MASYARAKAT," *Artikel XII*, vol. 8, no. 1, 2018, [Online]. Available: <http://jurnal.unismuhpalu.ac.id/index.php/PJKM>
- [7] A. Mubariz, D. Nur, E. Tungadi, M. Nur, and Y. Utomo, "Perancangan Back-End Server Menggunakan Arsitektur Rest dan Platform Node.JS (Studi Kasus: Sistem Pendaftaran Ujian Masuk Politeknik Negeri Ujung Pandang)," 2020.
- [8] M. Saed Novendri, A. Saputra, and C. E. Firman, "APLIKASI INVENTARIS BARANG PADA MTS NURUL ISLAM DUMAI MENGGUNAKAN PHP DAN MYSQL," 2019.
- [9] S. M. Mitchell and C. B. Seaman, *A Comparison of Software Cost, Duration, and Quality for Waterfall vs. Iterative and Incremental Development: A Systematic Review*. IEEE, 2009.
- [10] L. Lukwan, "Kontribusi Pengetahuan Kader Terhadap Kinerja Kader Posyandu di Puskesmas Matandahi Konawe Utara," *Jurnal Penelitian dan Pengembangan Pelayanan Kesehatan*, pp. 17–22, Apr. 2018, doi: 10.22435/jpppk.v2i1.37.
- [11] Kementerian Kesehatan R.I, *Pedoman Umum Pengelolaan Posyandu*. Jakarta, 2011.
- [12] Y. Rahmadi, Y. Adam P, and M. Azani, "PENGEMBANGAN MODUL FREEMIUM APLIKASI TEL-US (TELKOM UNIVERSITY STORE) MENGGUNAKAN METODE ITERATIVE INCREMENTAL DAN FRAMEWORK LARAVEL," 2015.
- [13] C. Larman and V. R. Basili, *Iterative and Incremental Development: A Brief History*. 2003.
- [14] A. Indra Damarani, I. Darmawan, and T. Nur Adi, "Pembangunan Portal Web Crowdsourcing Event Perguruan Tinggi Menggunakan Metode Iterative Incremental (Modul Peserta Event)," Taufik Nur Adi, 2015.
- [15] A. Triawan and A. R. Y. Siboro, "Penerapan Application Programming Interface (API) Pada Push

- Notification Untuk Informasi Monitoring Stok Barang Minim,” vol. 11, pp. 107–114, 2021, doi: 10.36350/jbs.v11i2.
- [16] Hasanuddin, H. Asgar, and B. Hartono, “RANCANG BANGUN REST API APLIKASI WESHARE SEBAGAI UPAYA MEMPERMUDAH PELAYANAN DONASI KEMANUSIAAN,” *Jurnal Informatika Teknologi dan Sains*, pp. 2686–3359, 2022.
- [17] A. Firdaus, S. Widodo, A. Sutrisman, S. Gading, F. Nasution, and R. Mardiana, “RANCANG BANGUN SISTEM INFORMASI PERPUSTAKAAN MENGGUNAKAN WEB SERVICE PADA JURUSAN TEKNIK KOMPUTER POLSRI,” *Jurnal Informanika*, vol. 5, no. 2, 2019, [Online]. Available: [www.kursuswebsite.org](http://www.kursuswebsite.org)
- [18] A. Mubarak, “RANCANG BANGUN APLIKASI WEB SEKOLAH MENGGUNAKAN UML (UNIFIED MODELING LANGUAGE) DAN BAHASA PEMROGRAMAN PHP (PHP HYPertext PREPROCESSOR) BERORIENTASI OBJEK,” 2019.
- [19] M. T. Prihandoyo, “Unified Modeling Language (UML) Model Untuk Pengembangan Sistem Informasi Akademik Berbasis Web,” *Jurnal Informatika: Jurnal Pengembangan IT (JPIT)*, vol. 03, no. 01, pp. 2477–5126, 2018.
- [20] H. Dhika, N. Isnain, and M. Tofan, “MANAJEMEN VILLA MENGGUNAKAN JAVA NETBEANS DAN MYSQL,” 2019.
- [21] I. Priyadi and R. J. Wellem Sagay, “PERANCANGAN DAN IMPLEMENTASI APLIKASI PASAR KOMODITAS ANDALAN DESA (PAKADES) BERBASIS MOBILE ANDROID UNTUK PETANI DESA,” 2019. [Online]. Available: [www.journal.ibmasmi.ac.id](http://www.journal.ibmasmi.ac.id)
- [22] R. K. Safitri and H. P. Putro, “Implementasi REST API untuk Komunikasi Antara ReactJS dan NodeJS (Studi Kasus : Modul Manajemen User Solusi247),” 2020.