

BAB II TINJAUAN PUSTAKA

A. Penelitian Terkait

No	Peneliti	Keterangan
(1)	(2)	(3)
1	Sebastina Nkechi Okofu ¹ , Clive Asuai, Oniovosa Okumoku-evroro, Maureen Ifeanyi Akazue(2025)[11]	Judul : <i>Development of an Enhanced Point of Sales System for Retail Business in Developing Countries</i> Metode : SDLC(System Development Life Cycle) <i>Waterfall</i> Software : PHP dan Mysql Keterangan (Hasil & Analisis Gap): Penelitian oleh Sebastina Nkechi Okofu et al. (2025) membahas pengembangan sistem Point of Sales (POS) untuk bisnis ritel di negara berkembang menggunakan metode SDLC <i>Waterfall</i> dengan teknologi PHP dan MySQL. Berbeda dengan penelitian ini, penelitian tersebut berfokus pada sistem POS berbasis web dan tidak membahas integrasi REST API pada aplikasi POS berbasis Android sebagai klien utama.
2	Primula Juventauricula, Buce Trias Hanggara, Djoko Pramono(2024)[12]	Judul : Pengembangan Sistem Informasi <i>Point of Sale</i> (POS) berbasis Web menggunakan Pendekatan Metode <i>Waterfall</i> (Studi Kasus: Restoran Altari) Metode : <i>System Development Life Cycle</i>(SDLC) <i>Waterfall</i> Software: PHP dengan <i>framework</i> Laravel

		Keterangan (Hasil & Analisis Gap): Penelitian Primula Juventauricula et al. (2024) mengembangkan sistem informasi POS berbasis web pada Restoran Altari menggunakan metode SDLC <i>Waterfall</i> dan <i>framework</i> Laravel. Penelitian ini menitikberatkan pengujian fungsional dan <i>usability</i>. Berbeda dengan penelitian ini, sistem yang dikembangkan masih bersifat <i>web-based</i> dan belum mengimplementasikan arsitektur REST API untuk integrasi dengan aplikasi Android.
3	Putu Gede Surya Cipta Nugraha1,Ni Wayan Wardani, I Wayan Sukarmayasa (2021)[13]	Judul : RANCANG BANGUN SISTEM INFORMASI SOFTWARE <i>POINT OF SALE</i> (POS) DENGAN METODE <i>WATERFALL</i> BERBASIS WEB Metode : <i>Waterfall</i> Software: PHP, MySQL Hasil : Mengembangkan sistem POS berbasis Web dengan PHP dan MySQL untuk membantu operasional UMKM. Keterangan (Hasil & Analisis Gap): Penelitian oleh Putu Gede Surya Cipta Nugraha et al. (2021) merancang sistem POS berbasis web untuk pelaku usaha kecil dengan metode <i>Waterfall</i> menggunakan PHP dan MySQL. Sistem mencakup manajemen barang, kategori, supplier, transaksi, dan pelaporan. Berbeda dengan penelitian ini, penelitian tersebut tidak mengembangkan

		aplikasi POS berbasis Android serta tidak memfokuskan pada integrasi REST API sebagai penghubung <i>client</i> dan <i>server</i>.
4	Suprianto, Muhammad Fadlan, dan Denis Prayogi. (2021)[14]	Judul : PERANCANGAN APLIKASI <i>POINT OF SALE</i> BERBASIS WEB PADA TOKO PROJECT SALFA TARAKAN Metode : <i>Waterfall</i> Software: PHP (<i>Hypertext Pre-processor</i>), MySQL (<i>My Structure Query Language</i>) Keterangan (Hasil & Analisis Gap): Penelitian Suprianto et al. (2021) merancang aplikasi POS berbasis web pada Toko Project Salfa Tarakan untuk menggantikan sistem pencatatan manual. Pengembangan sistem dilakukan menggunakan metode <i>Waterfall</i>. Berbeda dengan penelitian ini, penelitian tersebut tidak menerapkan konsep REST API dan tidak diarahkan untuk penggunaan sistem POS pada perangkat Android.
5	Yefta Christian, Kelvin. (2021)[15]	Judul : Perancangan Dan Penerapan Sistem Pos (<i>Point Of Sale</i>) Berbasis Web Pada Warung Zikry Metode : <i>Waterfall</i> Software: PHP (<i>Hypertext Preprocessor</i>), JS (JavaScript), JQuery, dan Bootstrap CSS serta menggunakan database phpMyAdmin. Keterangan (Hasil & Analisis Gap): Penelitian Yefta Christian dan Kelvin (2021)

		mengembangkan sistem POS berbasis web. Sistem dibangun menggunakan PHP dan teknologi web pendukung lainnya. Berbeda dengan penelitian ini, penelitian tersebut hanya berfokus pada aplikasi berbasis web dan belum mengimplementasikan integrasi REST API untuk mendukung aplikasi POS berbasis Android.
6	Rifqi Fadhilah, Yudhi Raymond Ramadhan, Lise Sri Andar Muni (2024)[16]	Judul : RANCANG BANGUN APLIKASI POINT OF SALE BERBASIS ANDROID MENGGUNAKAN <i>EXTREME PROGRAMMING</i> (STUDI KASUS BERINGIN LAUNDRY PURWAKARTA) Metode : <i>Agile Extreme Programming</i> Software: <i>framework</i> React Native, dan Firebase Keterangan (Hasil & Analisis Gap): Penelitian Rifqi Fadhilah et al. (2024) mengembangkan aplikasi POS berbasis Android pada UMKM Beringin Laundry menggunakan metode <i>Agile Extreme Programming</i> dengan teknologi React Native dan Firebase. Sistem ini bertujuan mendigitalisasi pencatatan transaksi dan laporan secara otomatis. Berbeda dengan penelitian ini, penelitian tersebut menggunakan pendekatan agile dengan backend Firebase, sedangkan penelitian ini menggunakan metode SDLC <i>Waterfall</i> serta REST API berbasis Laravel sebagai <i>backend</i> terpisah.

7	Riah Ukur Ginting, Aestetika O Sarumaha, Alexander Sibero, Harold Situmorang (2022) [17]	Judul : IMPLEMENTASI CUSTOMER RELATIONSHIP MANAGEMENT DALAM PERANCANGAN APLIKASI <i>POINT OF SALE</i> (POS) BERBASIS ANDROID Metode : <i>Waterfall</i> Software: Java dan MySQL. Keterangan (Hasil & Analisis Gap): Penelitian oleh Riah Ukur Ginting et al. (2022) mengimplementasikan aplikasi POS berbasis Android pada rumah makan Bakmi Yaahowu Sistem dibangun menggunakan bahasa pemrograman Java dan basis data MySQL. Berbeda dengan penelitian ini, penelitian tersebut tidak memfokuskan pembahasan pada perancangan dan integrasi REST API sebagai arsitektur utama pertukaran data.
8	Syahrul Ramadi, Yanto, Alhidayatuddiniyah T.W. (2025)[18]	Judul : PERANCANGAN APLIKASI <i>POINT OF SALES</i> (POS) DI CAFÉ REHAT DAN SINGGAH BERBASIS ANDROID Metode : Grounded Research dan pendekatan pengembangan sistem SDLC Agile Software: Kotlin, Android Studio Keterangan (Hasil & Analisis Gap): Penelitian Syahrul Ramadi et al. (2025) merancang aplikasi POS berbasis Android untuk Cafe Rehat dan Singgah menggunakan pendekatan <i>Agile</i> dan <i>Grounded Research</i>

		dengan bahasa pemrograman Kotlin. Berbeda dengan penelitian ini, penelitian tersebut tidak menitikberatkan pada implementasi REST API sebagai layanan backend yang terintegrasi secara sistematis.
9	Ainun Najib, Muhammad Yasir Zain (2020) [19]	Judul : APLIKASI <i>POINT OF SALE</i> MULTI OUTLET DAN MULTI PAYMENT BERBASIS WEB DAN ANDROID Metode : pengumpulan data, analisis sistem, perancangan sistem, perancangan basis data, perancangan antarmuka (metode tidak disebut namun pendekatannya mirip dengan SDLC klasik) Software: AngularJS, Flutter & Dart Keterangan (Hasil & Analisis Gap): Penelitian Ainun Najib dan Muhammad Yasir Zain (2020) mengembangkan aplikasi POS berbasis web dan Android dengan dukungan multi outlet dan multi payment menggunakan AngularJS dan Flutter. Penelitian ini berfokus pada perluasan fitur dan fleksibilitas penggunaan sistem. Berbeda dengan penelitian ini, penelitian tersebut lebih menekankan pada kompleksitas fitur, sementara penelitian ini berfokus pada integrasi REST API untuk sinkronisasi data transaksi secara terstruktur.
10	Marco Alfian Sumarto (2023)[20]	Judul : ANALISIS DAN PERANCANGAN APLIKASI <i>POINT OF SALE</i> BERBASIS MOBILE MENGGUNAKAN

		FRAMEWORK FLUTTER DENGAN METODE RAPID APPLICATION DEVELOPMENT (RAD) Metode : <i>Rapid Application Development</i> (RAD) Software: Dart dengan Flutter <i>framework</i>. Keterangan (Hasil & Analisis Gap): Penelitian oleh Marco Alfian Sumarto (2023) menganalisis dan merancang aplikasi <i>Point of Sale</i> berbasis mobile menggunakan <i>framework</i> Flutter dengan metode Rapid Application Development (RAD). Berbeda dengan penelitian ini, penelitian tersebut menggunakan metode RAD, sedangkan penelitian ini menggunakan metode SDLC <i>Waterfall</i> dengan fokus utama pada integrasi REST API sebagai arsitektur <i>backend</i> aplikasi POS berbasis Android.
--	--	--

Tabel 1 Penelitian Terkait

Berdasarkan pemaparan penelitian terkait pada Tabel 1 Penelitian Terkait, sebagian besar pengembangan sistem Point of Sales (POS) sebelumnya masih didominasi oleh platform berbasis Web menggunakan PHP dan MySQL, atau aplikasi Android yang dibangun menggunakan Java dan React Native dengan basis data langsung (seperti Firebase). Meskipun terdapat penelitian yang menggunakan framework Flutter, fokusnya cenderung pada metode pengembangan cepat (RAD).

Perbedaan mendasar dan kebaruan dalam penelitian ini terletak pada implementasi arsitektur sistem yang memisahkan *backend* dan *frontend* secara tegas melalui integrasi REST API. Penelitian ini menggabungkan keandalan Laravel sebagai penyedia layanan data (*backend*) dengan fleksibilitas Flutter sebagai antarmuka pengguna (*mobile frontend*). Arsitektur sistem ini diterapkan secara spesifik untuk menyelesaikan permasalahan operasional di Bengkel MMT Racing,



Gambar 1 Mind Map Penelitian

B. Landasan Teori

1. Sistem Point Of Sales (POS)

Sistem Point of Sale (POS) adalah platform komputer atau aplikasi yang digunakan di bisnis untuk mengotomatisasi proses transaksi penjualan, mengelola stok barang, dan menyajikan laporan keuangan secara efisien. Implementasi POS bertujuan untuk meningkatkan kecepatan serta akurasi dalam operasional bisnis, terutama dalam hal pencatatan penjualan, manajemen inventaris, dan analisis keuangan. Sistem ini membantu pemilik bisnis untuk memiliki kontrol yang lebih baik atas aspek-aspek kunci dari operasi mereka, memungkinkan pengambilan keputusan yang lebih cepat dan informasi yang lebih akurat terkait dengan kinerja bisnis[2].

Point of Sale (POS) sering digunakan di industri ritel untuk mengelola proses data transaksi seperti pembelian, penjualan eceran, pengembalian barang belanja, dan pelaporan transaksi. Penggunaan POS membantu dalam mempercepat proses transaksi, mengurangi kesalahan manusia, dan meningkatkan efisiensi operasional sebuah bisnis, terutama pada sektor ritel seperti minimarket, restoran, toko retail, atau bengkel. Informasi yang dihasilkan oleh sistem ini memiliki dampak yang signifikan dalam mendukung pengambilan keputusan strategis bagi pemilik bisnis swalayan, organisasi, atau perusahaan skala kecil dan menengah[21].

2. Android

Android adalah sistem operasi *mobile* berbasis Linux yang dikembangkan Google untuk perangkat layar sentuh seperti *smartphone* dan tablet, dengan arsitektur terdiri dari lima lapisan: *Linux Kernel*, *Libraries*, *Application Framework*, *Android Runtime*, dan *System Apps*. Sebagai sistem multi-user dimana setiap aplikasi berjalan terisolasi, Android telah merevolusi cara berkomunikasi dan bekerja, menggantikan metode tradisional dengan berbagai kemudahan seperti pemesanan makanan, perbankan digital, dan belanja online melalui aplikasi. Perkembangan Android terus mendorong inovasi di berbagai sektor melalui fitur-fitur canggih seperti notifikasi pintar, layanan berbasis lokasi, serta kolaborasi dokumen dan *video conference*[22].

3. *Application Programming Interface* (API)

API adalah sekumpulan protokol, definisi, dan tools yang memungkinkan berbagai aplikasi atau sistem untuk saling berkomunikasi dan bertukar data. API berfungsi sebagai perantara antara dua aplikasi yang berbeda, sehingga memungkinkan pengembang untuk mengakses fitur atau data dari suatu layanan tanpa perlu memahami detail implementasinya[23].

4. *Representational State Transfer* (REST)

RESTful API adalah sebuah arsitektur desain untuk membangun aplikasi web yang berinteraksi melalui protokol HTTP. Prinsip utamanya meliputi penggunaan *stateless communication*, identifikasi sumber daya melalui URL, dan representasi data dalam format standar seperti JSON atau XML. Dengan mengikuti prinsip-prinsip ini, API menjadi lebih mudah dipahami, dikembangkan, dan diskalakan[24].

Keuntungan utama dari penggunaan RESTful API adalah kemudahan integrasi antar sistem yang berbeda platform dan bahasa pemrograman. Setiap permintaan dari klien ke server bersifat independen dan berisi semua informasi yang dibutuhkan untuk diproses. Hal ini memungkinkan server untuk tidak

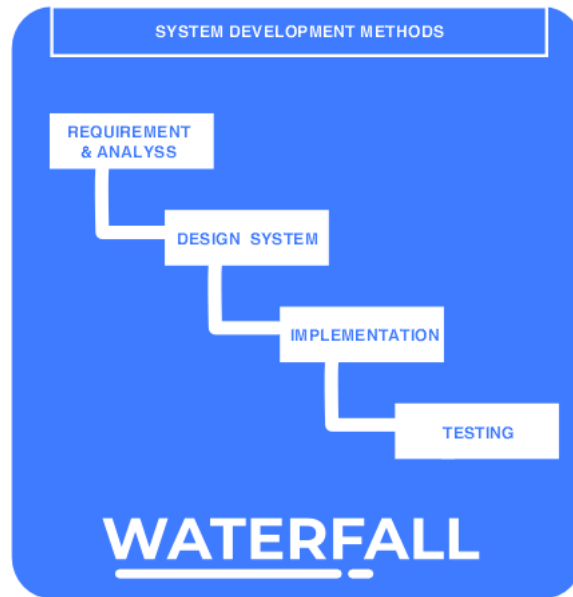
perlu menyimpan informasi sesi klien, sehingga meningkatkan skalabilitas dan reliabilitas aplikasi[25].

REST memanfaatkan protokol HTTP yang bersifat *stateless*, yang berarti setiap permintaan dari klien ke server bersifat independen. Perintah-perintah HTTP yang umum digunakan dalam REST meliputi fungsi GET untuk mengambil data, POST untuk membuat sumber daya baru, PUT atau PATCH untuk memperbarui, dan DELETE untuk menghapus. Data yang dikirimkan dari server umumnya berformat XML atau JSON sederhana tanpa adanya protokol pembungkusan data yang kompleks, sehingga informasi yang diterima lebih mudah dibaca dan diurai (*parsing*) oleh sisi klien[26].

Dalam penerapannya, REST (*Representational State Transfer*) lebih dominan digunakan untuk membangun layanan web yang berorientasi pada sumber daya. Konsep orientasi pada sumber daya ini berarti bahwa API menyediakan entitas data sebagai layanannya, bukan sekadar kumpulan aktivitas yang memanipulasi data tersebut. Arsitektur web *service* dengan gaya REST sangat ideal sebagai *back-end* untuk aplikasi berbasis *mobile*, karena kemudahan aksesnya dan format data JSON yang ringkas sehingga menghasilkan ukuran file yang lebih kecil[25].

5. Metode *Software Development Life Cycle (SDLC) Waterfall*

Metode *Waterfall* adalah model pengembangan perangkat lunak yang berurutan dan linear, di mana setiap fase harus diselesaikan sebelum fase berikutnya dimulai. Model ini mirip dengan air terjun, di mana setiap langkah mengalir ke langkah berikutnya[8]. Tahapan metode *waterfall* dapat dilihat pada gambar berikut:



Gambar 2 *Waterfall Development Method*

Metode air terjun, yang juga dikenal sebagai siklus hidup klasik atau "*Linear Sequential Model*", merupakan pendekatan pengembangan perangkat lunak yang sistematis dan berurutan. Prosesnya dimulai dengan spesifikasi kebutuhan pengguna, kemudian dilanjutkan melalui tahapan perencanaan (*requirement & analysis*), desain sistem (*design system*), implementasi (*implementation*), pengujian (*testing*), dan diakhiri dengan dukungan terhadap perangkat lunak yang dihasilkan. Pendekatan ini menggambarkan alur pengembangan yang terstruktur dan tidak dapat kembali ke tahap sebelumnya setelah dilalui[27].

Berdasarkan gambar 2, terdapat beberapa uraian mengenai tahapanyang dilewati, di antaranya :

- a. *Requirements & Analysis* (Perencanaan Kebutuhan)
Tahap ini adalah proses mengumpulkan dan mendokumentasikan semua kebutuhan yang diinginkan oleh pengguna atau klien. Kebutuhan ini mencakup semua aspek dari sistem yang akan dikembangkan, termasuk fungsionalitas, kinerja, dan antarmuka pengguna.
- b. *Design System* (Desain Sistem)
Setelah kebutuhan dikumpulkan, tahap berikutnya adalah mendesain sistem. Ini mencakup desain arsitektur sistem, desain database, desain antarmuka pengguna, dan desain modul-modul sistem. Tujuan utama dari tahap ini adalah untuk membuat cetak biru yang jelas untuk pengembangan sistem.
- c. *Implementation* (Impelementasi)

Pada tahap ini, desain yang telah dibuat diubah menjadi kode program yang dapat dijalankan. Para pengembang akan menulis kode program berdasarkan spesifikasi yang telah ditentukan dalam tahap desain.

d. *Testing* (Pengujian)

Setelah sistem diimplementasikan, tahap berikutnya adalah testing. Ini mencakup pengujian sistem untuk memastikan bahwa sistem berfungsi sesuai dengan kebutuhan yang telah ditentukan. Pengujian ini bisa mencakup *unit testing*, *integration testing*, *system testing*, atau *acceptance testing*.

Berikut ini adalah beberapa kelebihan dan kekurangan metode *Waterfall* [27] :

Kelebihan	Kekurangan
Struktur yang Jelas: Setiap fase memiliki tujuan yang jelas dan harus diselesaikan sebelum melanjutkan ke fase berikutnya.	Kurang Fleksibel: Sulit untuk kembali ke fase sebelumnya jika terjadi perubahan kebutuhan atau spesifikasi.
Dokumentasi Lengkap: Setiap langkah dalam pengembangan didokumentasikan dengan baik, memudahkan pemeliharaan dan pembaruan di masa mendatang.	Risiko Tinggi: Kesalahan yang terjadi di awal proyek dapat memiliki dampak yang besar dan sulit diperbaiki di fase selanjutnya.
Kemudahan Pengelolaan Proyek: Dengan pendekatan yang linear, lebih mudah untuk mengelola dan melacak progres proyek.	Tidak Cocok untuk Proyek yang Dinamis: Proyek yang membutuhkan perubahan dan penyesuaian terus-menerus kurang cocok menggunakan metode ini.

Tabel 2 Kelebihan dan kekurangan metode *waterfall*

6. Metode *Software Development Life Cycle* (SDLC) *Waterfall*

Postman adalah platform *tester* API yang memudahkan pengembang dalam membangun, menguji, dan mengelola API melalui antarmuka yang ramah pengguna untuk mengirim permintaan HTTP dan memeriksa respons. Postman mampu mensimulasikan perilaku API sebelum *backend* sebenarnya siap, dan menyediakan alat untuk dokumentasi API dan pemantauan[28].



Gambar 3 Logo Postman

7. Flutter

Flutter adalah sebuah *framework* sekaligus SDK (*Software Development Kit*) *open-source* yang dikembangkan oleh Google untuk membuat aplikasi *mobile* Android dan iOS[29]. Salah satu keunggulan utamanya adalah kemampuan pengembangan lintas platform, di mana developer dapat menggunakan satu basis kode (*codebase*) tunggal yang ditulis dalam bahasa pemrograman Dart untuk kedua platform tersebut [30]. Kode Dart ini kemudian diubah menjadi kode C/C++ dan dikompilasi secara native [37]. Oleh karena itu secara spesifik dirancang untuk membangun sisi *frontend* atau antarmuka pengguna (UI) aplikasi[30].



Gambar 4 Logo Flutter

8. Laravel

Laravel adalah sebuah kerangka kerja (*framework*) PHP yang sangat populer dan bersifat *open-source*, dikenal luas karena sintaksisnya yang elegan, fitur-fiturnya yang kaya, dan ekosistemnya yang matang untuk pengembangan aplikasi web modern[31]. Salah satu kegunaan utama Laravel adalah sebagai *backend* yang kuat untuk membangun *Application Programming Interface* (API)[32]. API ini berfungsi sebagai jembatan komunikasi antara sisi klien (seperti aplikasi *mobile*, web *frontend* JavaScript, atau sistem lain) dengan server atau *database*.



Gambar 5 Logo Laravel