

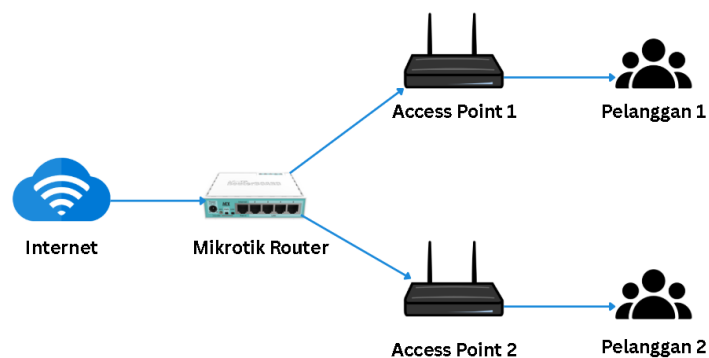
BAB II

LANDASAN TEORI

2.1. Dasar Teori yang Mendukung Produk

2.1.1. RT/RW Net

RT/RW Net merupakan jaringan internet berbasis komunitas yang dibangun dengan konsep *Resource Sharing* yang memungkinkan masyarakat di tingkat RT/RW untuk berbagi koneksi internet [6]. *Model* ini telah berkembang di Indonesia sebagai solusi untuk menjembatani kesenjangan digital di daerah-daerah yang belum terjangkau oleh ISP besar. Menurut penelitian Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), RT/RW Net menjadi salah satu alternatif penting dalam penyediaan akses internet di daerah semi-urban dan rural [7]. Gambar Diagram Konsep RT/RW Net ditunjukkan pada Gambar 2.1 sebagai berikut :



Gambar 2.1 Diagram Konsep RT/RW Net

Gambar 2.1 menggambarkan struktur jaringan RT/RW Net yang menghubungkan pengguna ke internet melalui MikroTik Router dan *Access Point* yang menyediakan koneksi nirkabel bagi pelanggan.

2.1.2. Pengembangan Sistem

Pengembangan sistem adalah proses penyusunan suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada [8]. Tujuan utama dari pengembangan sistem adalah untuk meningkatkan efisiensi, efektivitas, dan keunggulan kompetitif dari suatu

organisasi melalui penerapan teknologi informasi. Proses ini melibatkan serangkaian tahapan terstruktur, mulai dari analisis kebutuhan pengguna, perancangan, implementasi (pengkodean), pengujian, hingga pemeliharaan.

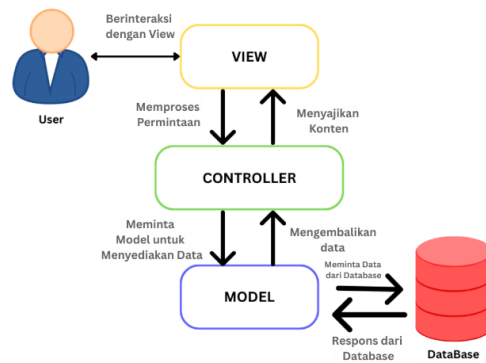
Dalam praktiknya, terdapat berbagai metode atau model pengembangan sistem yang dapat digunakan, yang biasa disebut sebagai *System Development Life Cycle* (SDLC). Salah satu metode yang populer dan relevan dengan pengembangan aplikasi modern adalah metode Agile. Metode Agile, seperti pendekatan Scrum yang digunakan dalam penelitian ini, menekankan pada pengembangan yang iteratif dan inkremental. Artinya, produk dikembangkan dalam siklus-siklus pendek (*sprint*) yang menghasilkan bagian fungsional dari sistem di setiap akhir siklusnya. Pendekatan ini memungkinkan fleksibilitas yang tinggi untuk beradaptasi terhadap perubahan kebutuhan dan memberikan hasil yang lebih cepat kepada pengguna akhir [9].

2.1.3. Sistem Informasi Berbasis Web

Sistem informasi berbasis *web* merupakan sistem yang menggunakan teknologi *web browser* dan jaringan internet sebagai *platform* untuk mengakses, mengelola, dan memproses informasi [10]. Sistem informasi berbasis *web* memiliki keunggulan dalam hal aksesibilitas, kemudahan implementasi, dan integrasi dengan sistem lain [11]. Dalam konteks pengelolaan bisnis, sistem informasi berbasis *web* memungkinkan otomatisasi proses, pengurangan biaya operasional, dan peningkatan efisiensi kerja [12].

2.1.4. Laravel Framework

Laravel adalah *framework* PHP *open-source* yang dikembangkan oleh Taylor Otwell pada tahun 2011 dan telah berkembang menjadi salah satu *framework* PHP paling populer [13]. *Laravel* 10, yang dirilis pada tahun 2023, menawarkan berbagai fitur canggih seperti Eloquent ORM, migrasi *database*, sistem *template Blade*, *middleware*, dan fitur keamanan yang komprehensif [14]. *Laravel* mengadopsi arsitektur MVC (*Model-View-Controller*) yang memfasilitasi pengembangan aplikasi yang terstruktur dan mudah dipelihara [15]. Gambar *Model-View-Controller* (MVC) pada *Laravel Framework* ditunjukkan pada Gambar 2.2 sebagai berikut :



Gambar 2.2 Diagram *Model-View-Controller* (MVC) pada *Laravel Framework*

Gambar 2.2 menggambarkan alur kerja *Model-View-Controller* (MVC) pada *Laravel*, yang menunjukkan interaksi antara Pengguna, *View*, *Controller*, *Model*, dan *Database*. Pengguna berinteraksi dengan *View* untuk mengirimkan permintaan, yang diproses oleh *Controller*. *Controller* kemudian meminta *Model* untuk menyediakan data yang diperlukan, yang kemudian diambil dari *Database* dan dikembalikan ke *Controller* sebelum akhirnya disajikan kembali melalui *View*.

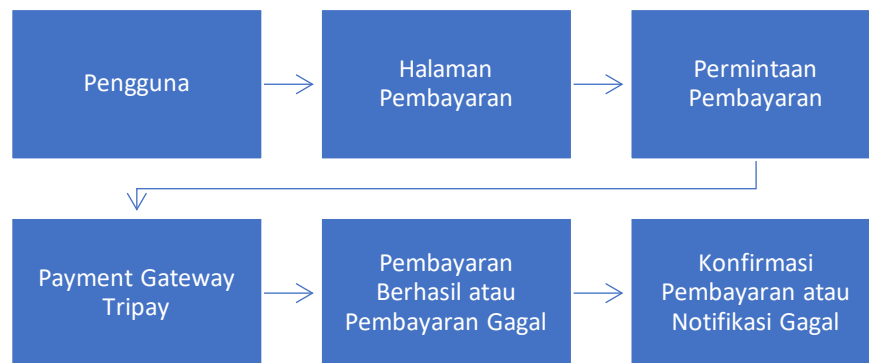
2.1.5. *Bootstrap 5*

Bootstrap adalah *framework front-end open-source* yang dikembangkan oleh Twitter untuk memfasilitasi pengembangan *web* responsif dan *mobile-first* [16]. *Bootstrap 5*, versi terbaru, menghilangkan ketergantungan pada *jQuery*, meningkatkan komponen, dan menawarkan utilitas CSS yang lebih fleksibel [17]. *Sneat Admin Template* adalah *template admin* berbasis *Bootstrap 5* yang menyediakan komponen UI modern dan responsif untuk aplikasi *web* [18].

2.1.6. *Payment Gateway*

Payment gateway adalah layanan yang memfasilitasi transaksi pembayaran *online* antara *merchant* dan *customer* dengan cara yang aman dan efisien [19]. *Tripay* adalah *payment gateway* lokal Indonesia yang menyediakan berbagai metode pembayaran seperti transfer bank, *e-wallet*, dan *minimarket* [20]. Integrasi *payment gateway* pada aplikasi *web* memungkinkan pelanggan melakukan

pembayaran secara *online* dengan aman dan nyaman [21]. Gambar Diagram Alur *Payment Gateway* Tripay ditunjukkan pada Gambar 2.3 sebagai berikut :

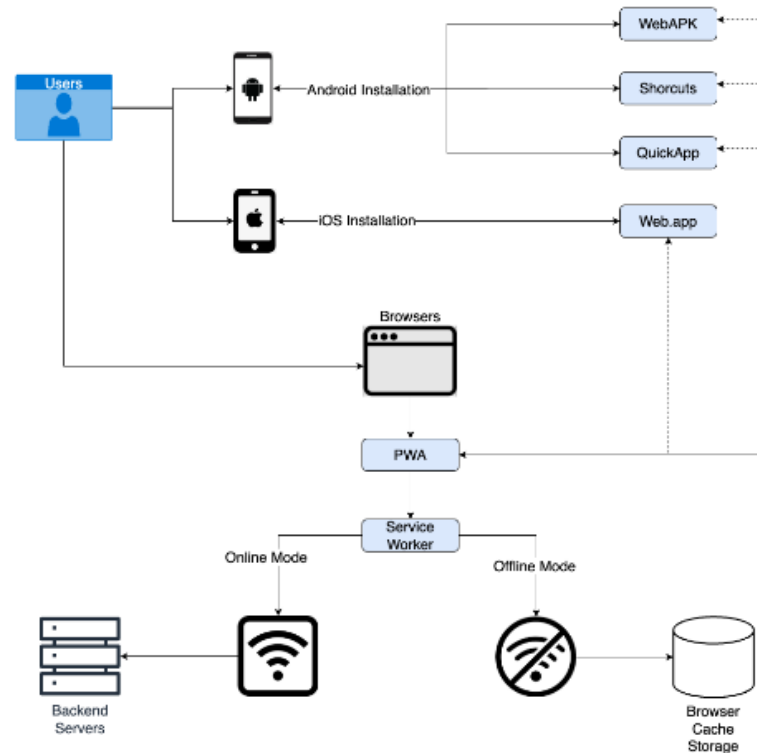


Gambar 2.3 Diagram Alur *Payment Gateway* Tripay

Gambar 2.3 menggambarkan alur pembayaran menggunakan *Payment Gateway* Tripay, di mana pengguna mengakses Halaman Pembayaran, kemudian mengirimkan Permintaan Pembayaran. *Payment Gateway* Tripay memproses permintaan pembayaran tersebut, dan hasilnya dapat berupa Pembayaran Berhasil atau Pembayaran Gagal. Setelah itu, sistem akan mengirimkan Konfirmasi Pembayaran jika berhasil atau Notifikasi Gagal jika pembayaran tidak berhasil.

2.1.7. *Progressive Web App* (PWA)

Progressive Web App (PWA) adalah pendekatan pengembangan aplikasi *web* yang menggabungkan kelebihan *web* dan aplikasi *native* [22]. PWA memungkinkan aplikasi *web* dapat diinstal pada perangkat pengguna, berfungsi secara *offline*, dan memberikan pengalaman pengguna yang mirip dengan aplikasi *native* [23]. Teknologi PWA memanfaatkan *service workers*, *manifest*, dan *cache API* untuk meningkatkan kinerja dan *user experience* [24]. Gambar Diagram Alur Penggunaan *Progressive Web App* (PWA) ditunjukkan pada Gambar 2.4 sebagai berikut :



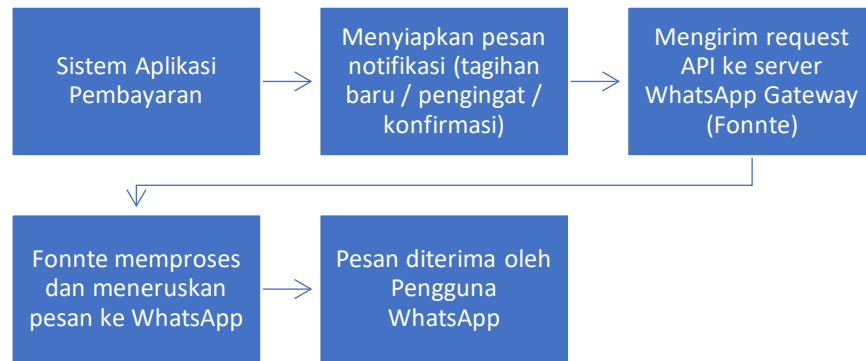
Gambar 2.4 Diagram Alur Penggunaan *Progressive Web App* (PWA)

Gambar 2.4 menggambarkan alur penggunaan *Progressive Web App* (PWA), di mana pengguna dapat menginstal aplikasi pada perangkat Android atau iOS. Aplikasi dapat diakses melalui browser, dengan dukungan mode *online* dan *offline* berkat *service worker* yang mengelola *cache* dan komunikasi dengan server *backend*. Dalam mode *online*, aplikasi dapat mengakses server *backend*, sementara dalam mode *offline*, aplikasi tetap berfungsi menggunakan data yang disimpan dalam *cache browser*.

2.1.8. *WhatsApp Gateway*

WhatsApp Gateway merupakan layanan yang memungkinkan sistem informasi mengirimkan pesan WhatsApp secara otomatis melalui API [25]. Teknologi ini semakin populer sebagai saluran komunikasi bisnis karena tingginya tingkat keterbacaan pesan dibandingkan dengan email atau SMS [26]. *Fonnte WhatsApp Gateway* adalah salah satu penyedia layanan WhatsApp API di Indonesia yang memungkinkan pengiriman pesan WhatsApp secara massal dan

terotomatisasi [27]. Gambar Ilustrasi Alur Kerja WhatsApp Gateway ditunjukkan pada Gambar 2.5 sebagai berikut :



Gambar 2.5 Ilustrasi Alur Kerja *WhatsApp Gateway*

Gambar 2.5 mengilustrasikan alur kerja *WhatsApp Gateway*, di mana sistem mengirimkan *request* melalui API ke server *WhatsApp Gateway* (dalam hal ini Fonnte), yang selanjutnya meneruskan pesan ke pengguna WhatsApp. Sistem dapat mengirim berbagai jenis notifikasi seperti pengingat pembayaran, konfirmasi pembayaran, dan pemberitahuan tagihan baru. Implementasi *WhatsApp Gateway* dalam sistem informasi pembayaran memungkinkan komunikasi yang lebih efektif dengan pelanggan, meningkatkan tingkat pembayaran tepat waktu, dan meningkatkan kepuasan pelanggan.

2.1.9. *Multi-role Access Control*

Sistem dengan *multi-role access control* memungkinkan pengaturan hak akses yang berbeda berdasarkan peran pengguna dalam sistem [28]. Dalam konteks sistem informasi pembayaran, sistem *multi-role* memungkinkan admin memiliki akses penuh untuk mengelola data, sementara pelanggan hanya dapat mengakses data yang relevan dengan akun mereka [29]. *Laravel* menyediakan fitur bawaan untuk mengelola *otentikasi* dan otorisasi berbasis peran [30]. Tabel *Multi-role Access Control* ditunjukkan pada Tabel 2.1 sebagai berikut :

Tabel 2.1 *Multi-role Access Control*

Peran	Akses Data Pelanggan	Membuat Tagihan	Membayar Tagihan	Akses Admin Panel
Admin	√	√	√	√

<i>User</i>	√	x	√	x
-------------	---	---	---	---

Tabel 2.1 menggambarkan hak akses yang dimiliki oleh dua peran pengguna dalam sistem: Admin dan *User*. Admin memiliki akses penuh untuk mengelola data pelanggan, membuat tagihan, membayar tagihan, dan akses panel admin. Sementara itu, *User* hanya memiliki akses untuk mengakses data pelanggan dan membayar tagihan, namun tidak dapat membuat tagihan atau mengakses panel admin.

2.1.10. MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang banyak digunakan dalam pengembangan aplikasi *web* untuk menyimpan dan mengelola data secara terstruktur dan efisien. MySQL mendukung pengolahan data transaksi dan penyimpanan data yang dapat diakses oleh aplikasi berbasis *web* seperti sistem informasi pembayaran. Penggunaan MySQL memberikan keandalan dan kemudahan integrasi dengan bahasa pemrograman PHP dan *framework Laravel* [31].

2.1.11. PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman *server-side* yang digunakan untuk membuat aplikasi *web* dinamis. PHP berperan penting dalam menghubungkan aplikasi *web* dengan *database* seperti MySQL, memproses logika bisnis, serta mengelola interaksi pengguna secara *real-time*. Versi modern PHP menyediakan fitur dan peningkatan performa yang mendukung pengembangan sistem informasi yang responsif dan aman. Dalam konteks pengembangan sistem informasi akademik berbasis *web*, penggunaan PHP terbukti efektif untuk mengelola data secara terstruktur dan mendukung kebutuhan aplikasi *web* [32].

2.1.12. Visual Studio Code

Visual Studio Code adalah *code editor* sumber terbuka yang populer digunakan oleh pengembang aplikasi *web*. *VS Code* menyediakan fitur seperti penyorotan sintaks, *auto-completion*, *debugging*, serta integrasi dengan sistem kontrol versi Git. Penggunaan *VS Code* mempermudah pengembang dalam menulis

dan mengelola kode PHP dan *Laravel*, sehingga meningkatkan produktivitas dan kualitas pengembangan aplikasi [33].

2.2. Produk Sejenis atau Terkait

Berbagai penelitian telah dilakukan dalam upaya mengembangkan sistem informasi untuk pembayaran *online*, namun masing-masing memiliki kelebihan dan keterbatasannya sendiri. Penelitian oleh Anam *et al.* (2025) mengembangkan sistem informasi pembayaran internet berbasis *web* yang dirancang untuk memudahkan pengelolaan data pelanggan, pembuatan tagihan, dan pencatatan pembayaran [34]. Meskipun demikian, sistem tersebut belum mengintegrasikan *payment gateway* dan masih mengandalkan proses pembayaran manual. Berbeda dengan penelitian tersebut, sistem yang dikembangkan pada penelitian ini menggunakan *framework Laravel* 10 dan mengintegrasikan *payment gateway* Tripay untuk memfasilitasi pembayaran *online* secara otomatis.

Selain itu, penelitian oleh Nurlaila *et al.* (2023) mengembangkan aplikasi pengelolaan pembayaran berbasis Android untuk membantu memudahkan administrasi pembayaran tagihan dalam bidang pendidikan [35]. Aplikasi ini memungkinkan admin untuk mengelola data pelanggan dan tagihan melalui perangkat *mobile*. Namun, aplikasi ini hanya terbatas pada *platform* Android. Di sisi lain, sistem yang dikembangkan dalam penelitian ini menggunakan teknologi (PWA), yang memungkinkan akses dari berbagai perangkat, baik *desktop* maupun *mobile*, tanpa bergantung pada *platform* tertentu.

Sebuah penelitian lain oleh Rikardo. (2023) mengembangkan sistem informasi pembayaran SPP berbasis *web* dengan integrasi *payment gateway* Midtrans [36]. Meskipun memiliki fitur yang mirip dengan sistem yang akan dikembangkan dalam penelitian ini, penelitian tersebut berfokus pada konteks pendidikan. Sebaliknya, penelitian ini mengadaptasi konsep integrasi *payment gateway* untuk konteks layanan internet RT/RW Net, dengan kebutuhan yang lebih spesifik dan berbeda.

Sementara itu, RL Radius adalah salah satu aplikasi komersial modern yang digunakan untuk mengelola layanan internet berbasis *hotspot* dan PPPoE [37].

Aplikasi ini menyediakan fitur lengkap seperti manajemen pelanggan, pengaturan paket layanan, serta pencatatan pembayaran. Namun, penggunaan RL Radius memiliki ketergantungan pada perangkat Mikrotik. Sebagai pembeda, sistem yang dikembangkan dalam penelitian ini bersifat independen, memungkinkan integrasi dengan berbagai infrastruktur jaringan tanpa ketergantungan pada perangkat tertentu. Tabel Perbandingan Sistem Penelitian Terkait ditunjukkan pada Tabel 2.2 sebagai berikut :

Tabel 2.2 Perbandingan Sistem Penelitian Terkait

Aspek	Penelitian [32]	Penelitian [33]	Penelitian [34]	Sistem Ini
<i>Framework</i>	CodeIgniter	Android	Web+Midtrans	<i>Laravel</i> 10+Tripay
<i>Platform</i>	Web	Android Mobile	Web	Web+PWA
<i>Payment Gateway</i>	Tidak ada	Tidak ada	Ada (Midtrans)	Ada (Tripay)
<i>Responsif Mobile</i>	Tidak ada Terbatas	Tidak ada Hanya Android	Tidak ada Ada	Ada Full PWA

Tabel 2.2 menggambarkan perbandingan antara sistem yang dikembangkan dalam penelitian ini dengan produk sejenis atau terkait yang ditemukan dalam penelitian sebelumnya. Sistem ini menggunakan *Laravel* 10 + *Tripay* sebagai *framework* dan *payment gateway*, dengan dukungan (PWA) serta integrasi *WhatsApp Gateway*. Dibandingkan dengan produk lain, sistem ini lebih lengkap karena mendukung *platform* Web dan *Mobile*, serta mengintegrasikan *payment gateway* untuk pembayaran otomatis.