

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Teori Dasar**

##### **2.1.1. *Android***

*Android* dibuat sebagai sistem operasi sumber terbuka berbasis Linux yang disesuaikan untuk ponsel pintar. Setelah mengakuisisi perusahaan *Android Inc.* pada tahun 2005, *Google* mendanai pengembangan lebih lanjut pada tahun 2009. Pada tahun 2007, *smartphone* pertama yang diberdayakan *Android* dirilis, dan sejak saat itu, berkembang pesat. (Tendra & Suwarti, 2020).



**Gambar 2.1** Logo *Android*

##### **2.1.2. Aplikasi**

Aplikasi adalah perangkat lunak terkompilasi yang dirancang untuk melakukan tugas tertentu bagi pengguna akhir layanan aplikasi atau aplikasi lain. Menurut kamus komputer eksekutif menggambarkan aplikasi sebagai memperbaiki masalah menggunakan salah satu dari banyak metodologi pemrosesan data untuk melakukan beberapa perhitungan atau pemrosesan data yang diprediksi. (Juansyah, 2015).

### **2.1.3. Implementasi**

Implementasi adalah langkah desain berdasarkan temuan analisis diterjemahkan ke dalam bahasa pemrograman tertentu dan perangkat lunak yang dihasilkan dengan nyata diterapkan sehingga siap untuk digunakan. Tujuan implementasi adalah untuk memeriksa modul desain sehingga pengguna dapat menawarkan umpan balik kepada pembuat sistem.(Murtadho et al., 2016).

### **2.1.4. Aplikasi *Mobile***

Istilah "aplikasi *mobile*" mengacu pada perangkat lunak yang dapat digunakan saat pengguna bepergian bebas di antara lokasi yang berbeda dan tidak mengalami gangguan koneksi atau penundaan. Aplikasi ini tersedia untuk digunakan pada ponsel dan PDA, di antara perangkat nirkabel lainnya. (Roihan et al., 2019).

## **2.2. Teori Khusus**

### **2.2.1. *E-Wallet***

*E-wallet* atau *electronic wallet* disebut sebagai “dompet digital” sebagai wahana transaksi jual beli yang menjadi kebutuhan di masa wabah COVID-19 saat ini. Hal ini karena individu berusaha untuk menghindari kontak fisik (jarak fisik) dalam semua aspek kehidupan mereka, termasuk perdagangan, dan sebagai akibatnya, transaksi berbasis elektronik menjadi lebih umum. (Nuha et al., 2020)

### 2.2.2. UMKM (Usaha Mikro Kecil dan Menengah)

UMKM adalah badan usaha yang dimiliki oleh orang atau kelompok dan dinilai berdasarkan pendapatan dan jumlah tenaga kerja. UU No. 20 Tahun 2008 menyebutkan bahwa UMKM harus memiliki siklus usaha, asas, tujuan, pemberdayaan yang terkoordinasi, dan menyadari konsekuensi administratif. UU No. 20 Tahun 2008 membahas sektor industri, pertanian, perdagangan, jasa, dan transportasi. (Sihaloho et al., 2020).

### 2.2.3. Metode *Agile*

Metode *agile* adalah strategi pengembangan perangkat lunak yang paling efektif dan cepat. Meskipun ada metode untuk menjadi modular dengan efektif menggunakan teknologi ini, itu tidak menjelaskan proses eksplisit tentang bagaimana mengembangkan tipe model tertentu. Metode *agile* juga memiliki istilah yang disebut iterasi, yaitu durasi waktu pembuatan perangkat lunak. Setiap iterasi yang dilakukan memiliki durasi yang berbeda, tergantung dari fitur yang dikembangkan dan metode yang digunakan. Umumnya, jangka waktunya adalah 1-4 minggu. Dikarenakan metode ini pengembangan *software* lebih diprioritaskan daripada dokumentasi. Tujuan metode *agile* adalah ini memfasilitasi proses pengembangan perangkat lunak menjadi lebih dinamis dan tidak harus menunggu *stakeholder* untuk menyelesaikan tugasnya terlebih dahulu. Metode *agile* ini ada delapan jenis yaitu:

1. *Extreme Programming (XP)*

*Extreme Programming* awalnya diperkenalkan pada 6 Maret 1996. Ini adalah salah satu metodologi *Agile* paling populer karena keberhasilannya yang

meluas dalam bisnis dari berbagai industri. Kolaborasi adalah fokus dari *Extreme Programming*. Dalam tim kolaboratif, manajer, pelanggan, dan pengembang semuanya memainkan peran yang berbeda.

2. *Adaptive Software Development (ASD)*

*Jim Highsmith* memperkenalkan *Adaptive Software Development (ASD)* sebagai metode untuk mengembangkan perangkat lunak dan sistem yang rumit. ASD dan dasar-dasar kerja sama manusia dan tim yang mengatur diri sendiri.

3. *Dynamic Systems Development Methods (DSDM)*

*Dynamic Systems Development Methods (DSDM)* adalah metode untuk membuat dan memelihara sistem terbatas waktu yang memanfaatkan pembuatan prototipe inkremental dalam lingkungan yang terkondisi.

4. *Scrum Methodology*

Jeff Sutherland menciptakan *Scrum* pada awal 1990-an, dan selanjutnya disempurnakan oleh *Ken Schwaber* dan *Ken Beedle*. *Scrum* adalah teknik tangkas untuk pengembangan perangkat lunak yang rumit. *Scrum* membagi pengembangan perangkat lunak menjadi tahapan yang lebih kecil yang dikenal sebagai sprint.

5. *Crystal*

*Cockburn* dan *Highsmith* mengembangkan *Crystal*. *Crystal Methodology* adalah metodologi *Agile* yang menekankan kondisi tim di atas prosedur dan teknologi. Keterlibatan tim, komunikasi, umpan balik, dan dokumentasi

adalah langkah pertama. Teknik *Crystal* menguntungkan untuk pengembangan perangkat lunak karena disesuaikan dengan keadaan tim.

6. *Feature Driven Development*

Metode *agile* yang disebut "*Feature Driven Development*" digunakan untuk menyelesaikan satu proyek. Biasanya, setiap putaran FDD hanya berlangsung antara 2 hingga 10 hari. Sebagian besar iterasi dalam metode *Scrum* berlangsung antara dua dan empat minggu. Karena itu, fitur-fitur di FDD lebih kecil dan lebih spesifik, sehingga bisa selesai tepat waktu.

7. *Agile Modelling*

*Agile Modeling* adalah metode untuk mendokumentasikan dan memodelkan perangkat lunak secara realistis. Untuk memodelkan perangkat lunak dengan sukses untuk proyek pengembangan perangkat lunak, sekumpulan ide, konsep, dan praktik yang dikenal.

8. *Rational Unified Process*

*Rational Unified Process* adalah paradigma pengembangan perangkat lunak yang diproduksi oleh *Rational Software*, anak perusahaan IBM sejak tahun 2003. Pendekatan ini membagi sistem aplikasi menjadi banyak komponen yang dapat diulang (analisis, desain, implementasi, dan pengujian).

Jenis metode *agile* yang di atas yang penulis menggunakan *Agile Modelling*, karena metode ini praktis untuk mendokumentasi dan pemodelan perangkat lunak. Metode *agile* ini memiliki prinsip utama yaitu seorang pengembang memiliki keunggulan teknis adalah fokus utama pengembangan perangkat lunak menggunakan metode *Agile*. (Yusril et al., 2021).

#### 2.2.4. QR Code

*QR Code* merupakan bentuk simbol dua dimensi yang dikembangkan untuk mengirimkan data dan mendapatkan respon dengan cepat oleh *Denso Wave* pada tahun 1994. Pola fungsional dan area mengkodekan diatur dalam susunan persegi di setiap simbol QR-Code. Batas zona meluas ke segala arah di sekitar setiap simbol. Berbeda dengan *barcode*, *QR Code* dapat berisi informasi dalam dimensi horizontal dan vertikal. (Priyambodo et al., 2020).



Gambar 2.2 QR Code

#### 2.2.5. TensorFlow

*TensorFlow* adalah kerangka kerja yang dirancang untuk menunjukkan algoritme pembelajaran mesin dan mengeksekusi instruksi berdasarkan pengetahuan yang dimilikinya tentang target atau objek yang diketahui, dan membedakan antara hal-hal yang berbeda. (Kartika & Pratiwi, 2021).



Gambar 2.3 Logo TensorFlow

### 2.2.6. *Android Studio*

*Android studio* adalah IDE (*Integrated Development Environment*) yang direkomendasikan untuk membuat aplikasi *Android*, dan tersedia gratis. Pada 16 Mei, selama *Google I/O 2013*, *Google* mengumumkan diluncurkan *Android Studio*. Sejak *Eclipse* telah mengambil alih *Android Studio* sebagai IDE masuk untuk membuat aplikasi *Android*. (Mukhtar, 2018).



Gambar 2.4 *Android Studio*

### 2.2.7. *Kotlin*

*Kotlin* adalah bahasa berorientasi objek yang dapat diubah menjadi *bytecode Java* menggunakan infrastruktur kompilasi LLVM (*low-level virtual machine*) dan beroperasi pada *Java Virtual Machine*. Sebagian besar pekerjaan di *Kotlin* telah dilakukan oleh *programmer* di *JetBrains*. *Kotlin* dimaksudkan untuk mendukung kode *Java* ketika sintaks tidak kompatibel dengan *Java*. (Randi et al., 2020).



Gambar 2.5 Bahasa Pemrograman Kotlin

### 2.2.8. *Firebase*

*Firebase* adalah teknologi *database* yang terhubung ke *cloud* milik *Google Inc.*, yang secara teratur menyinkronkan data dalam format *JSON* dengan klien yang terhubung ke internet. Fungsionalitas *realtime database* dari *Firebase*

memungkinkan setiap klien yang terhubung ke Internet untuk menyampaikan informasi dengan segera. *Realtime Database* adalah *database* yang tidak menggunakan SQL karenanya, pengoptimalan dan fiturnya berbeda dari *database* SQL. *Realtime Database* API dimaksudkan untuk memfasilitasi eksekusi operasi yang cepat. (Mulyono et al., 2021).



**Gambar 2.6** Layanan *Firebase*

### **2.2.9. Google Cloud Platform**

*Google Cloud Platform* adalah layanan komputasi berbasis awan yang berbagi perangkat keras dengan penawaran Google yang ditujukan untuk konsumen (seperti Penelusuran dan YouTube). Ini menyediakan berbagai alat manajemen selain layanan *cloud* yang meliputi komputasi, penyimpanan, analisis data, dan pembelajaran mesin. (Ramsari & Ginanjar, 2022).



**Gambar 2.7** Layanan Google Cloud Platform

### **2.2.10. Retrofit API**

*Retrofit* API adalah pustaka untuk aplikasi *Android* dan *Java* yang memungkinkannya terhubung ke server HTTP dengan cara yang aman. *Retrofit* adalah klien untuk REST menghilangkan kebutuhan untuk mengembangkan cara

khusus untuk terhubung ke layanan web REST. API juga digunakan untuk memberi ponsel cerdas akses ke sumber daya, mengelola peralihan antar proses, dan memberi aplikasi lain akses ke data. (Lumba, 2021).

### **2.2.11. Figma**

Pengertian aplikasi *figma* merupakan alat desain yang dapat diakses melalui online atau komputer. Prototipe digital dapat dibuat dan diuji dengan bantuannya. *Plugin Iconify* dan Komunitas *Figma* adalah dua *plugin* dan fitur tambahan yang mungkin membantu dalam proses desain. (Aji Yoga Pratama et al., 2022).



**Gambar 2.8** *Figma*

### **2.2.12. Pengujian *Black-Box***

Pengujian perangkat lunak yang disebut pengujian *black-box* berfokus pada persyaratan fungsional aplikasi. Supaya berfungsi pada pengujian *black-box* mengabaikan struktur kontrol dan berkonsentrasi pada keahlian domain. Pengembang perangkat lunak mampu membangun status masukan yang memenuhi persyaratan fungsional aplikasi dengan pengujian *black-box*. (Snadhika Jaya et al., 2018).

### **2.2.13. UML (*Unified Modelling Language*)**

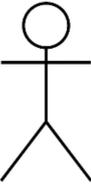
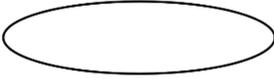
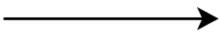
Untuk memvisualisasikan, mendeskripsikan, membuat, dan merekam telah dibuat sistem pengembangan perangkat lunak berorientasi objek disebut *Unified*

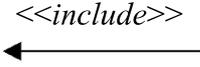
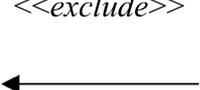
*Modeling Language (UML)*. Dalam penelitian ini memiliki 4 diagram yaitu *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram* dibuat dengan rancang. (Syafitri, 2016).

### 1. *Use Case Diagram*

*Use case Diagram* Dalam diagram *use case*, interaksi antara *use case* dan aktor-aktor termasuk orang, perangkat, atau sistem lain yang berinteraksi dengan sistem yang sedang dikembangkan.(Mulyono et al., 2021).

**Tabel 2.1** Simbol – Simbol pada Use Case Diagram

No.	Gambar	Nama	Keterangan
1		Aktor	Simbol nama aktor sering diawali dengan kata untuk menunjukkan bahwa simbol aktor adalah representasi dari seseorang, aktor tersebut belum tentu merupakan orang. Interaksi antara sistem informasi yang akan dibangun dengan entitas lain, seperti orang, proses, atau sistem lainnya.
2		<i>Use Case Diagram</i>	Fungsionalitas sebagai unit yang berkomunikasi dengan aktor atau unit lain sering ditunjukkan dengan memulai frase nama <i>use case</i> dengan kata kerja.
3		<i>Association</i>	Pertukaran informasi antara aktor dan <i>use case</i> yang digunakan atau dieksploitasi oleh

No.	Gambar	Nama	Keterangan
			aktor.
4		<i>Generalization</i>	Hubungan di mana objek di atas struktur data objek induk dan perilakunya digunakan bersama oleh objek turunan (turunan) di bawahnya,
5		<i>Include/ uses</i>	Menunjukkan bahwa <i>use case</i> sepenuhnya bergantung pada fungsi <i>use case</i> lain.
6		<i>Exclude/ extend</i>	Jika suatu kondisi terpenuhi, menunjukkan bahwa <i>use case</i> adalah tambahan yang berguna untuk <i>use case</i> lainnya.

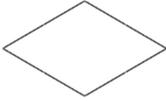
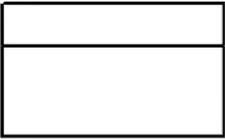
Sumber: Data Penelitian (2023)

## 2. Activity Diagram

Diagram ini digunakan untuk menunjukkan bagaimana alur kontrol sistem antar objek bekerja. Grafik ini juga dapat digunakan untuk menggambarkan aliran aktivitas sistem lainnya ditampilkan aktivitas diagram yang terjadi pada aplikasi *e-wallet*. (Mulyono et al., 2021).

Tabel 2.2 Simbol – Simbol pada *Activity Diagram*

No.	Gambar	Nama	Keterangan
1.		Status Awal	Diagram aktivitas berisi status awal untuk aktivitas sistem.
2.		Aktivitas	Tindakan yang dilakukan sistem sering didahului oleh

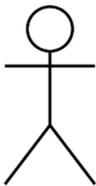
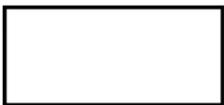
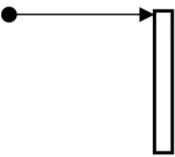
No.	Gambar	Nama	Keterangan
			kata kerja.
3.		Percabangan	Ketika ada beberapa opsi untuk suatu aktivitas pada percabangan.
4.		Penggabungan	Penggabungan, proses menggabungkan banyak aktivitas menjadi satu.
5.		Status Akhir	Activity Diagram mempunyai keadaan akhir yang menggambarkan bagaimana sistem berperilaku.
6.		<i>Swimlane</i>	<i>Swimlane</i> memecahkan entitas perusahaan yang bertanggung jawab atas terjadinya aktivitas tertentu.

Sumber: Data Penelitian (2023)

### 3. *Sequence Diagram*

*Sequence diagram* adalah sejenis diagram yang menunjukkan hubungan antara berbagai item dalam suatu sistem dengan sangat rinci. Hal-hal yang berhubungan dengan operasi biasanya diatur dari kiri ke kanan. (Mulyono et al., 2021).

**Tabel 2.3** Simbol – Simbol pada *Sequence Diagram*

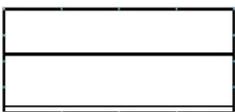
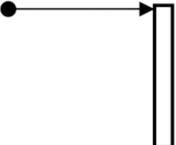
No.	Gambar	Nama	Keterangan
1		Aktor	Simbol nama aktor sering diawali dengan kata untuk menunjukkan bahwa simbol aktor adalah representasi dari seseorang, aktor tersebut belum tentu merupakan orang. Interaksi antara sistem informasi yang akan dibangun dengan entitas lain, seperti orang, proses, atau sistem lainnya.
2		Garis hidup	Mendeklarasikan keberadaan entitas objek, serta antarmukanya.
3		Objek	Mendeklarasikan objek interaksi pesan.
4		Waktu aktif	Mendeklarasikan suatu objek aktif dan dapat menerima dan menanggapi pesan.
5		Pesan	Data informasi dapat berkomunikasi dan bertukar informasi, termasuk data tentang kejadian.

**Sumber:** Data Penelitian (2023)

#### 4. *Class Diagram*

*Class diagram* adalah semacam diagram struktur dalam UML yang dengan jelas menggambarkan struktur dan deskripsi kelas, karakteristik, metode, dan koneksi dari setiap objek. Itu statis dalam arti bahwa diagram kelas tidak menggambarkan apa yang terjadi ketika kelas digabungkan, melainkan koneksi apa yang ada. (Mulyono et al., 2021).

**Tabel 2.4** Simbol – Simbol pada *Class diagram*

No.	Gambar	Nama	Keterangan
1.		Kelas	Himpunan dari objek-objek yang berbagi atribut serta tindakan yang identik.
2.		Antarmuka	Identik dengan ide antarmuka dalam pemrograman berorientasi objek.
3.		<i>Association</i>	Komunikasi antar partisipan yang terlibat dengan aktor memperkerjakan partisipan kasus atau memanfaatkan aktor kasus.
4.		Waktu aktif	Mendeklarasikan bahwa suatu objek aktif dan dapat menerima dan menanggapi pesan.
5.		Pesan	Data informasi item dapat berkomunikasi dan bertukar informasi, termasuk data tentang kejadian.

**Sumber:** Data Penelitian (2023)

### 2.3. Pendahuluan Terdahulu

Penelitian yang telah dilakukan sebelumnya dan dibahas tentang Implementasi Aplikasi *E-Wallet* Untuk UMKM Menggunakan Metode *Agile* Dengan *QR Code* Pada *Tensorflow* Berbasis *Android*, di antara topik-topik lainnya, berikut contohnya:

**Tabel 2.5** Penelitian Terdahulu

No.	Nama Pengarang	Judul Penelitian	Masalah Penelitian	Metode dan Hasil Penelitian
1.	(C. Yosepu, N. Krishna Vardhan, & R.V. Sudhakar, 2018)	<i>A Study on Image Association Using Android Equipment through Machine Learning</i> (ISSN: 2347 – 6710)	<i>The burden of this technology is lessened by using TensorFlow for computer vision and machine learning on Android items. Identifying objects in images involves at least three layers of artificial neural networks.</i>	<i>The method that used in this research was method Agile. The result of this search is the item in the provided picture is recognized with a high success rate, and its characteristics are communicated verbally.</i>
2.	(Syukron Rizqi Ardhi, Bambang Subana, & Warno, 2019)	Aplikasi Sistem Pembayaran Menggunakan	Uang tunai tidak efisien untuk sebagian besar bisnis. Alasan lain mengapa	Metode yang digunakan adalah Metode <i>Waterfall</i> . Hasil penelitian adalah Aplikasi

No.	Nama Pengarang	Judul Penelitian	Masalah Penelitian	Metode dan Hasil Penelitian
		Dompet Elektronik (M-Pay) Berbasis <i>Android</i> Di Toko Irsyad (ISSN: 2355-326X)	bisnis lebih suka menghindari uang tunai adalah karena sulit dilacak. Biaya tinggi untuk pengelolaan dan pengadaan berasal dari kendala. Transaksi tunai memiliki risiko keamanan yang lebih tinggi. Butuh waktu lama untuk memproses pembayaran menggunakan uang tunai.	pembayaran menggunakan dompet elektronik berbasis <i>Android</i> . Metode ini diterapkan melalui studi masalah untuk menciptakan solusi. Kekhawatiran manajemen pembayaran Irsyad sebagian besar terfokus di sana. Sebuah sistem sederhana untuk membeli dan menjual barang harus diterapkan di toko.
3.	(Dwi Prasetyo Noto Kusumo, Sekreningsih Nita, 2019)	Perancangan <i>Game Android</i> Adventure Gajah Mada dengan Metode <i>Agile Development</i>	Pengembangan permainan sebelumnya yang kurang proses atau teknik yang tidak tepat cenderung	Metode yang digunakan adalah Metode <i>Agile</i> . Hasil Penelitian adalah Versi permainan <i>Android</i> saat ini, <i>Adventure</i> Gajah

No.	Nama Pengarang	Judul Penelitian	Masalah Penelitian	Metode dan Hasil Penelitian
		(ISSN: 2685-5615)	memakan waktu lebih lama dan memiliki lebih banyak kesalahan.	Mada, berbeda dengan versi baru ini, yang telah dioptimalkan untuk gameplay yang mulus dengan tetap mempertahankan beberapa gangguan. Bagian selanjutnya dari gim ini memiliki visual 3D, menjadikan keseluruhan estetika lebih modern dan menarik. Dengan menggunakan metode <i>Agile</i> , desain dan konstruksi <i>game</i> yang baik.
4.	(Gusrio Tendra & Suwarti, 2020)	Aplikasi Dompot <i>Digital</i> Berbasis <i>Android</i> Pada Bus Transmetro Kota Pekanbaru	Alat tukar untuk semua transaksi masih berupa uang tunai yang harus dimiliki para penumpang untuk memesan transportasi. Sayangnya,	Metode yang digunakan adalah metode <i>Waterfall</i> . Hasil penelitian adalah untuk Penumpang dan pengguna jasa transportasi Transmetro

No.	Nama Pengarang	Judul Penelitian	Masalah Penelitian	Metode dan Hasil Penelitian
		(ISSN: 2337-7631)	terkadang biaya perjalanan ditetapkan dan tidak semua penumpang memiliki jumlah uang yang tepat yang dibutuhkan untuk menanggungnya. Untuk mendapatkan uang kembali, yang tidak selalu menjadi pilihan, penumpang harus memberikan sebagian dari uang mereka sendiri.	Pekanbaru akan lebih mudah melakukan pembelian dan transaksi lainnya dengan hadirnya aplikasi berbasis dompet digital. Jika petugas memindai <i>QR Code</i> pada aplikasi penumpang, saldo yang berlaku akan langsung dipotong dari aplikasi. Pelanggan tidak lagi dirugikan jika tidak mendapatkan pengembalian biaya transaksi.
5.	(Rupali B. Kale* , & Ruchika K. Balwade, Vipin B. Gawai, 2020)	<i>Online Food Ordering System for College Canteen</i> (ISSN: 2229-7111)	<i>Since the administrator is unaware of how much supply is available for a certain product or service that is</i>	<i>The method that used in this research was Rapid Application Development (RAD). The result of this search is to expedite the food ordering and</i>

No.	Nama Pengarang	Judul Penelitian	Masalah Penelitian	Metode dan Hasil Penelitian
			<i>being provided at various points throughout their search, users are not successful at developing, producing, or responding rapidly.</i>	<i>delivery processes for hungry students who now must stand in two separate lines.</i>
6.	(Aryo Damar Prasetya, Rio Korio Utoro, & Rickman Roedavan, 2021)	Perancangan <i>Mobile Application</i> Prediksi Nilai Akhir Menggunakan <i>Tensorflow Lite</i> Berbasis <i>Android</i> (ISSN: 2442-5826)	Terjadi peningkatan kesalahan penyampaian dan jenis kesalahan lainnya selama masa karantina karena banyak instruktur memanfaatkan program untuk melaporkan kinerja siswa kepada orang tua secara individu.	Metode yang digunakan adalah <i>Rapid Application Development</i> (RAD). Hasil penelitian adalah aplikasi ini dapat digunakan untuk memprediksi nilai akhir siswa menggunakan kecerdasan buatan dan hasil prediksi nilai akhir dapat dikirimkan ke masing-masing siswa dan orang tua siswa via email.
7.	(Diana Kartika	Implementasi <i>Tensorflow</i>	Pengunjung asing kesulitan	Metode yang digunakan adalah

No.	Nama Pengarang	Judul Penelitian	Masalah Penelitian	Metode dan Hasil Penelitian
	Novia Pratiwi, (2021)	<i>Lite dan Firebase Authentication</i> Pada Aplikasi <i>Android SignBoard Translator</i> (ISSN: 2548-7779)	memahami rambu-rambu lalu lintas, menyebabkan mereka melakukan banyak pelanggaran.	metode <i>Agile</i> . Hasil penelitian adalah Saat diuji, model <i>the.tflite</i> mencapai tingkat akurasi 99% saat mengategorikan rambu lalu lintas dalam kumpulan data yang dipermasalahan. Saat informasi pengguna dicatat dengan benar di <i>Firestore Authentication</i> , desain sistem Daftar dan Masuk untuk aplikasi penerjemah papan nama dapat diterapkan.
8.	(Tri Yuliati, & Tri Handayani, 2021)	Pendampingan Penggunaan Aplikasi Digital QRIS Sebagai Alat Pembayaran Pada UMKM	Banyak pelaku UMKM yang masih belum memahami manfaat dan fakta tentang QRIS.	Metode yang digunakan adalah metode kualitatif. Hasil penelitian adalah secara khusus, pendekatan kualitatif sedang diterapkan. Penelitian

No.	Nama Pengarang	Judul Penelitian	Masalah Penelitian	Metode dan Hasil Penelitian
		(ISSN: 2721-5008)		menyimpulkan bahwa penggunaan QRIS mengancam kegunaan QRIS sebagai mekanisme pembayaran non tunai dalam membatasi penyebaran <i>COVID-19</i> di masa pandemi. Penggunaan ponsel pintar saja dapat mempercepat dan mempermudah proses pembayaran.
9.	(Dennis, & Nia Ekawati, 2021)	Perancangan Aplikasi Absensi Karyawan Dengan Menggunakan Kode QR Berbasis <i>Android</i> (UPB)	Kehadiran di Asacom kini dicatat secara manual, di atas kertas. Hal ini dapat menyebabkan inefisiensi di pihak bisnis dan karyawan, serta kesalahan lainnya, seperti gagal	Metode yang digunakan adalah Metode <i>Waterfall</i> . Hasil Penelitian adalah penggunaan <i>QR Code System</i> pada ponsel pintar untuk Membantu Pemilik Toko dalam Melacak Kehadiran Mampu mengembangkan aplikasi pelacakan

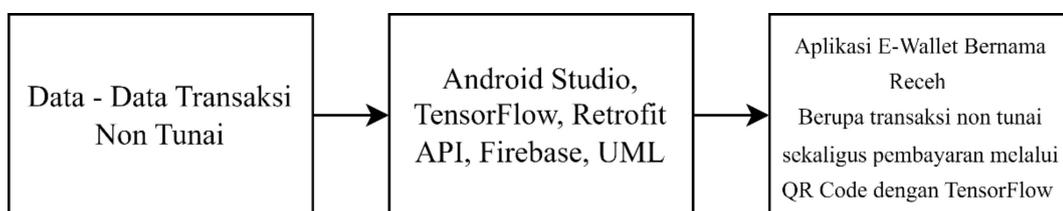
No.	Nama Pengarang	Judul Penelitian	Masalah Penelitian	Metode dan Hasil Penelitian
			mengumpulkan data pada saat laporan bulanan atau kehilangan data jadinya kurang bertanggung jawab.	kehadiran <i>realtime QR Code</i> berbasis <i>Android</i> .
10.	(Anusha B., & Dr. Aravinda T V, 2021)	<i>CNN Based Robust Traffic Sign Detection System</i> (ISSN: 2582-5208)	<i>The ability to recognize and decipher road signs is crucial for riding about and photographing the infrastructure..</i>	<i>The method that used in this research was method Agile. The result of this search is presents the fundamental steps required to recognize and analyze a traffic sign or container captured by a camera. The effectiveness of the aforementioned techniques was increased by using the GPU.</i>

**Sumber:** Data Penelitian (2023)

## 2.4. Kerangka Pemikiran

Kerangka pemikiran adalah persyaratan yang disepakati secara umum untuk penyelidikan ke dalam luasnya masalah untuk menemukan solusi yang bisa diterapkan. Dengan demikian, parameter penyelesaian studi dapat terlihat dengan jelas. (Nuraeni & Bethani Suryawardani, 2017)

Media yang digunakan sebelumnya untuk pembayaran tunai secara manual sehingga membuat pelanggan merasa kecewa dengan waktu kurang efisien prosesnya. Penggunaan media dalam proses transaksi non tunai dengan *QR Code* berbasis *android*, diharapkan dapat meningkatkan minat pelanggan. Untuk kepentingan penelitian ini dibuat dalam bentuk bagan berikut sebagai kerangka kerja:



**Gambar 2.9** Kerangka Pemikiran  
**Sumber:** Data Penelitian (2023)

Gambar di atas pada masukan yang dilakukan yaitu data-data transaksi non tunai seperti kebutuhan harian, cemilan, dan lain-lain. Lalu proses selanjutnya adalah melakukan perancangan dan pembuatan aplikasi *Android* dengan alat *tools* yaitu *Android Studio*, *TensorFlow*, *Retrofit API*, *Firebase*, dan *UML*. Keluaran dari proses di atas adalah Aplikasi *E-Wallet* bernama *Receh* berupa transaksi non tunai sekaligus pembayaran melalui *QR Code* dengan *TensorFlow*.