

BAB II

TINJAUAN PUSTAKA

2.1. Kajian Pustaka

Pada era teknologi pemanfaatan teknologi *android* sebagai OS perangkat seluler sudah banyak diterapkan oleh berbagai manufaktur perangkat ponsel pintar, dengan hal tersebut mayoritas masyarakat Indonesia dewasa ini sudah menggunakan perangkat ponsel pintar *android* digunakan sebagai media komunikasi dan informasi elektronik, salah satu pemanfaatan ponsel pintar *android* sebagai media komunikasi dan informasi adalah digitalisasi dan pengarsipan transaksi jual beli melalui perangkat lunak kasir, kemudian dilakukan kajian pustaka dengan tujuan memudahkan dalam menganalisis topik-topik yang pernah dibuat untuk kemudian dapat dijadikan referensi pada penelitian ini.

Penelitian yang berjudul “Rancang Bangun Aplikasi Kasir Berbasis *Android* (Studi Kasus: Toko Abah Pancing)” memaparkan rancang bangun aplikasi kasir pada platform *android* untuk studi kasus Toko Abah Pancing menggunakan metode *waterfall*. Berdasarkan hasil penerapan dan pengujian aplikasi tersebut dapat berjalan sesuai dengan kegunaannya [10].

Penelitian yang berjudul “Rancang Bangun Aplikasi Penjualan dan Persediaan Obat pada Apotek Berbasis *Android*” menjelaskan rancang bangun aplikasi penjualan dan persediaan obat dengan metode Extreme Programming (XP) untuk platform *android*, yang menghasilkan aplikasi penjualan berbasis *android* dapat memantau penjualan dan persediaan obat di apotek. Pengujian aplikasi ini menunjukkan bahwa fitur yang dijalankan sudah sesuai dengan hasil yang diinginkan sehingga diharapkan dapat meningkatkan proses bisnis apotek [11].

Penelitian dengan judul “Perancangan Aplikasi Pembukuan Menggunakan Metode *Agile Scrum*” menjelaskan perancangan aplikasi pembukuan menggunakan metode *agile scrum* berbasis *android*

menghasilkan aplikasi pembukuan berbasis android yang dapat mengatur pengeluaran dan pemasukan barang, melakukan transaksi serta mencetak struk secara otomatis dan membuat laporan penjualan otomatis, hasil tes menunjukkan bahwa integrasi dengan bluetooth printer berjalan dengan baik, serta aplikasi dapat terhubung/terputus dengan lancar [12].

Penelitian dengan judul “Rancang Bangun Aplikasi Kasir Usaha Mikro Kecil Menengah Menggunakan Metode Block Programming (Studi Kasus: Pelangi Store)” yang memaparkan rancang bangun aplikasi kasir untuk Usaha Mikro Kecil Menengah dengan menggunakan metode block programming berbasis *android*, dalam pembangunan aplikasi dengan metode block programming yang menjelaskan bagaimana alur proses yang didapat dalam bisnis sehingga mudah dibuat kedalam bentuk program, dan diharapkan proses bisnis dapat lebih efisien dan efektif [3].

Penelitian yang berjudul “Rancang Bangun Aplikasi Kasir Penjualan pada Usaha Mikro Kecil Menengah 3Manstore Berbasis Web” yang menjelaskan rancang bangun aplikasi kasir pada UMKM 3Manstore dengan platform aplikasi website, pada penelitian yang telah dilakukan mendapatkan luaran berupa sebuah Aplikasi kasir untuk UMKM 3Manstore yang mengutamakan pada proses transaksi dan pencetakan nota, pada aplikasi ini juga mempunyai 2 pengguna dengan peran sebagai admin dan user, pada aplikasi juga memiliki fitur yang dapat memantau secara langsung tentang untung rugi dan juga terdapat fitur pembayaran hutang [13].

Tabel 2.1 Kajian pustaka

No	Penulis, Tahun dan Judul	Masalah	Hasil	Metode	Persamaan	Perbedaan
1	Muhamad Nurdiansyah (2020), Rancang Bangun Aplikasi Kasir Berbasis <i>Android</i> (Studi Kasus: Toko Abah Pancing)	Toko Abah Pancing memerlukan sebuah media yang dapat memudahkan pemilik toko dalam melihat harga dan stock alat-alat pancing	Hasil dari penelitian ini didapat perhitungan vector V dengan menggunakan metode Weighthed Product dengan hasil nilai terbesar sebagai alternatif terbaik.	Penelitian ini menggunakan metode waterfall dengan melakukan pengembangan sistem membuat pembaruan yang dimulai dari analisis, desain, pengkodean, pengujian dan pendukung.	Persamaan dari penelitian yang dilakukan ialah merancang sebuah aplikasi kasir berbasis android dan digunakan untuk suatu usaha tertentu	Perbedaan dengan penelitian ialah metodologi yang digunakan pada perancangan peneliti sebelumnya menggunakan metode <i>waterfall</i> sedang penulis menggunakan metode <i>extreme programming</i>
2	Asri Mulyani, Yosep Septiana dan Rizky Helmi pada (2022), Rancang Bangun Aplikasi Penjualan dan Persediaan Obat pada Apotek Berbasis <i>Android</i>	Tidak adanya dokumentasi tentang persediaan obat yang terdapat pada apotek sehingga menyebabkan para pegawai harus melakukan pendataan secara manual sehingga	Hasil dari penelitian ini berupa aplikasi yang dapat membantu proses pengolahan obat seperti menjual dan pendataan obat yang terdapat pada Gudang apotek. Selain itu aplikasi ini dapat	Penelitian ini menggunakan metode Extreme Programming yang berfungsi untuk mempermudah para pegawai untuk melakukan pengolahan data.	Persamaan dengan peneliti sebelumnya ialah metode yang diterapkan dalam pembangunan aplikasi	Perbedaan dari peneliti sebelumnya adalah permasalahan yang diselesaikan berbeda pada bidang usaha yang dikembangkan

No	Penulis, Tahun dan Judul	Masalah	Hasil	Metode	Persamaan	Perbedaan
		memungkinkan terjadinya ketidakakuratan pada data yang di dokumentasikan.	memudahkan pegawai untuk mencetak kuitansi jika ada pembeli yang datang dan juga pegawai dapat melakukan pengecekan lewat smartphone.			
3	Dhandy Joenathan Kurnia Putra dan Penidas Fiodinggo Tanaem (2022), Perancangan Aplikasi Pembukuan Menggunakan Metode <i>Agile Scrum</i>	Toko masih menggunakan pendataan secara manual yang menyebabkan toko mengalami daya produksi yang semakin menurun. Pendataan manual juga berdampak negatif bagi toko karena tidak efektif dan sering terjadi kesalahan pendataan.	Penelitian ini memiliki luaran berupa aplikasi yang dapat membantu toko untuk mengelola barang yang masuk dan keluar, dapat mencetak kuitansi otomatis jika terdapat pembelian barang, dan dapat membuat catatan penjualan secara spontan. Aplikasi yang dibuat juga	Pada penelitian ini menggunakan metode agile yang dapat mempermudah peneliti dalam mengembangkan aplikasi yang dibuat selain itu peneliti juga menggunakan framework scrum yang dapat menyelesaikan masalah dan dapat membantu	Persamaan dalam permasalahan yang diselesaikan berupa pembukuan dari hasil pencatatan penjualan	Perbedaan metode penelitian yang digunakan, pada peneliti sebelumnya menggunakan scrum sebagai metode penelitian sedangkan penelitian ini menggunakan metode extreme programming yang kedua metode ini merupakan cabang dari metode agile

No	Penulis, Tahun dan Judul	Masalah	Hasil	Metode	Persamaan	Perbedaan
			dapat terkoneksi dengan printer Bluetooth sehingga pegawai dapat melakukan request cetak kuitansi satu persatu.	peneliti membuat aplikasi yang sesuai dengan keinginannya.		
4	Agit Amrullah dan Fata Aulia (2022), Rancang Bangun Aplikasi Kasir Usaha Mikro Kecil Menengah Menggunakan Metode Block Programming (Studi Kasus: Pelangi Store)	Toko masih menggunakan cara manual untuk mendata suatu barang yang dikarenakan kurangnya pengetahuan tentang perkembangan teknologi sehingga terjadi banyak masalah yang dihadapi toko	Aplikasi berhasil dibuat serta berdasarkan testing menggunakan blkbox testing pada fitur fitur yang telah ditentukan semua fitur terkait berhasil berjalan sesuai apa yang di harapkan.	Pada penelitian ini peneliti menerapkan metode waterfall yang dibagi menjadi beberapa tahapan yaitu analisis permasalahan, desain sistem, implementasi, integrasi, dan perawatan.	Persamaan dari penelitian sebelumnya ialah mengembangkan aplikasi untuk usaha UMKM	Peneliti sebelumnya menerapkan metode block programming sedangkan penelitian yang dikembangkan menggunakan metode extreme programming
5	Asri Mulyani, Ridwan Setiawan dan Ruli Ahmad Rusmana	Toko masih menggunakan cara manual untuk menghitung dan	Penelitian ini menghasilkan luaran berupa aplikasi kasir yang	Peneliti menggunakan metode RUP untuk	Persamaan dari penelitian sebelumnya ialah mengembangkan	Perbedaan dari peneliti sebelumnya adalah framework yang

No	Penulis, Tahun dan Judul	Masalah	Hasil	Metode	Persamaan	Perbedaan
	(2022), Rancang Bangun Aplikasi Kasir Penjualan pada Usaha Mikro Kecil Menengah 3Manstore Berbasis Web	melakukan pendataan barang sehingga sering terjadi kesalahan perhitungan yang menyebabkan kerugian. Selain itu toko masih mencetak kuitansi secara manual.	dapat membantu kinerja UMKM 3Manstore. Aplikasi yang dibuat dapat membantu pegawai dengan cara membantu melakukan perhitungan dan pendataan barang yang terdapat pada toko, selain itu aplikasi ini juga dapat membantu mencetak kuitansi secara otomatis.	mempermudah pembuatan aplikasi yang dibantu dengan penggunaan framework code igniter yang berguna untuk membantu membuat rancangan aplikasi sehingga aplikasi dapat dibuat dengan cepat.	aplikasi untuk usaha UMKM	digunakan untuk pengembangan aplikasi serta platform yang dipilih, dan proses pencetakan kwitansi pada penjualan

2.2. Landasan Teori

2.2.1. Usaha Mikro Kecil dan Menengah (UMKM)

Suatu usaha dagang yang berdiri sendiri milik pribadi dan bukan merupakan cabang dari perusahaan lain sesuai dengan kriteria modal usaha yang sudah ditentukan tolak ukurnya merupakan pengertian dari Usaha Mikro Kecil dan Menengah (UMKM) [14]. UMKM memiliki beberapa ciri yaitu jenis barang dalam suatu usaha tidak tetap dan dapat berubah sewaktu-waktu, lokasi usaha tidak menetap dan dapat berubah, belum menerapkan manajemen yang baik dalam usahanya seperti uang pengusaha yang masih tercampur dengan uang hasil perusahaan, masih banyak masyarakat yang belum mempunyai tekad untuk menjadi seorang pengusaha yang mumpuni, tingkat pendidikan manajemen sumber daya manusia masih rendah, pelaku UMKM terkadang masih belum menggunakan layanan peminjaman dari bank dan lebih memilih meminjam ke lembaga non-bank seperti pinjaman online yang tidak berizin atau legal.

2.2.2. Kasir

Point of Sales (POS) atau biasa dikenal dengan mesin kasir online merupakan sistem gabungan dari *hardware* dan *software* yang memiliki banyak fungsi selain melakukan pendataan transaksi, Point of Sales juga memiliki beragam fitur yang digunakan. Umumnya Point of Sales berbentuk mesin, smartphone, tablet dan berbagai macam mesin yang digunakan untuk melakukan transaksi penjualan. Keunggulan sistem Point of Sales dari mesin kasir biasa adalah POS memiliki beragam fitur sedangkan mesin kasir memiliki fitur yang terbatas.

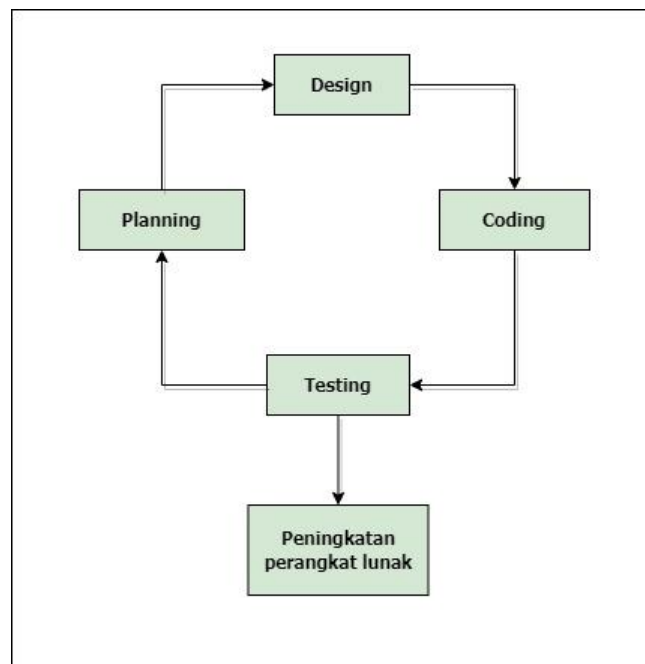
Point of Sales telah berkembang dan juga sering digunakan sebagai tempat menyimpan informasi pelanggan, menghitung laba rugi bahkan meringkas laporan penjualan. Point of Sales dapat digunakan pada non-internet atau memanfaatkan internet dengan fitur canggih untuk membantu pemilik usaha. Keuntungan dalam memiliki POS adalah lebih mudah digunakan karena tidak perlu membeli alat khusus atau mesin kasir dengan

harga yang mahal cukup dengan menggunakan tablet atau smatphone yang digunakan dengan praktis[15].

Kasir merupakan sebuah pekerjaan yang bertujuan untuk melayani pembeli atau pelanggan. Kualitas suatu usaha dapat ditentukan dari tingkat pelayanan yang diberikan, jika pelayanan memuaskan maka memungkinkan pelanggan untuk kembali lagi ke toko tersebut. Mesin kasir merupakan sebuah alat yang berfungsi untuk membantu kasir untuk melakukan sebuah transaksi, selain itu dengan menggunakan mesin kasir pegawai dapat lebih mudah untuk melakukan perhitungan transaksi yang sudah dilaksanakan.

2.2.3. Metode Agile Extreme Programming (XP)

Extreme programming (XP) adalah metode pengembangan agile yang menggunakan prinsip pengulangan pada tahap pengembangannya. Penerapan metode extreme programming cukup ringkas dan sederhana serta meningkatkan efisiensi dalam pengerjaan aplikasi yang ingin dicapai oleh sebab itu metode ini sangat bagus digunakan pada pengembangan aplikasi berskala kecil [16]. Beberapa tahapan-tahapan *Extreme programming* dalam pengembangan sistem yang dilakukan [8].



Gambar 2.1 Tahapan Metode Agile Extreme Programming

Langkah pada *Extreme programming* memiliki penulisan seperti gambar diatas dengan langkah – langkah tahapan sebagai berikut:

1. *Planning* (Perencanaan)

Langkah awal dari metode *Extreme programming* ialah perencanaan dimana pada tahap ini peneliti mengidentifikasi permasalahan yang muncul dengan melakukan identifikasi masalah dan menganalisa kebutuhan sistem[17]. Sehingga didapat alur proses dari perangkat lunak, fitur - fitur yang ada pada perangkat lunak, fungsi dari perangkat, serta alur proses pengembangan perangkat lunak.

2. *Design* (Perancangan)

Tahap ini melakukan rancangan pemodelan terhadap aplikasi yang akan dibangun berdasarkan dari kebutuhan sistem yang telah dianalisis[17].

3. *Coding* (Pengkodean)

Tahap pengkodean adalah tahapan dimana programmer mengimplementasi desain yang sebelumnya sudah dirancang kedalam bentuk bahasa pemrograman yang dapat dipahami oleh *computer*[17].

4. *Testing* (Pengujian)

Agar dapat memastikan perangkat lunak yang telah dibuat dapat berjalan sesuai dengan yang diinginkan maka dilakukan pengujian agar dapat menemukan *bug* atau *error* pada perangkat lunak[17].







2.2.4. UML (*Unified Modelling Language*)

Unified Modelling Language atau UML adalah bahasa visual yang digunakan untuk menentukan, membangun, dan mendokumentasikan artefak dari sebuah sistem[18]. Selain bahasa visual UML juga terhubung secara langsung dengan berbagai bahasa pemrograman seperti *java* dan C++. UML digunakan untuk menghindari penggunaan kata yang terlalu banyak dan penjelasan detail yang tidak penting sehingga dapat menyebabkan kesalahpahaman[18]. Keuntungan dengan menggunakan UML yaitu setiap elemen yang digunakan memiliki arti dan makna tersendiri sehingga tidak ada kesalahpahaman, notasi yang digunakan mudah dipahami dan tidak berbelit belit dan dapat melakukan penggambaran aspek aspek penting pada

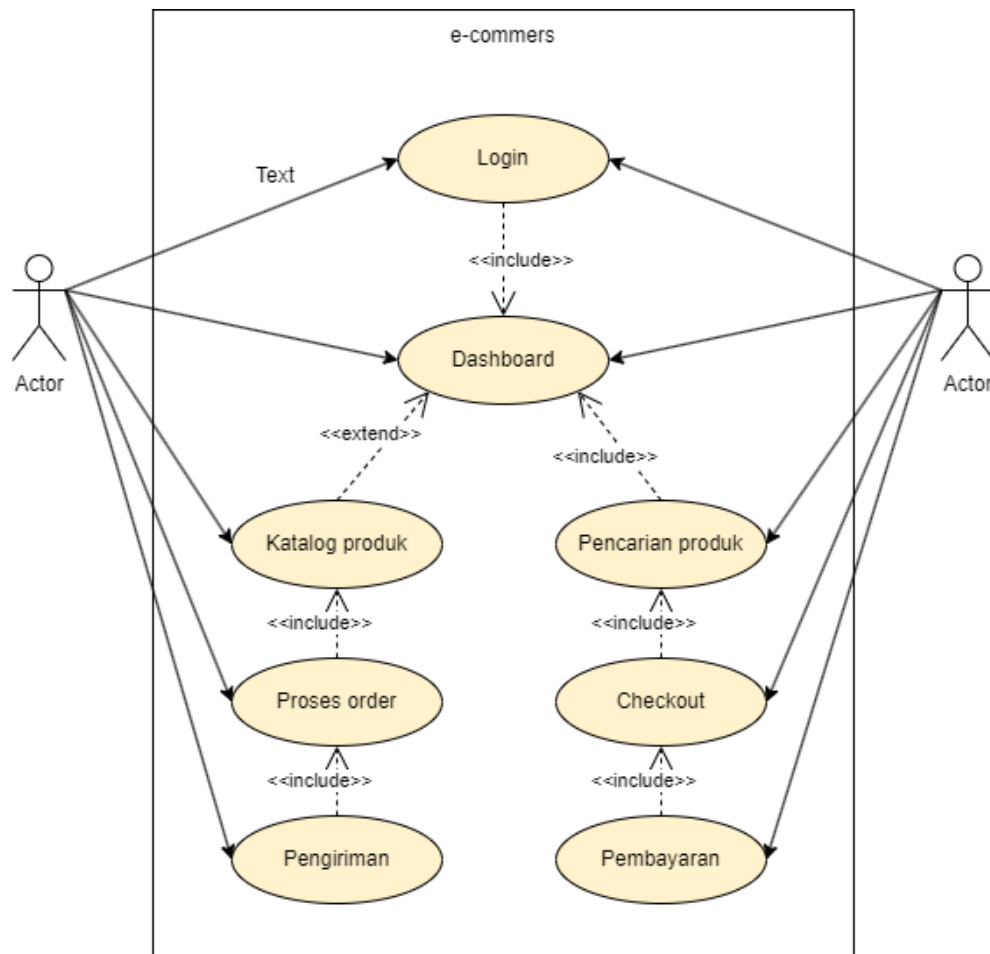
sistem[18]. UML memiliki beberapa daftar diagram yang akan dijelaskan sebagai berikut[18]:

2.2.4.1 Use-Case Diagram

Use-case merupakan gambaran interaksi yang dilakukan antar aktor dan sistem secara keseluruhan serta relasi antar *use case*[18]. *Usecase* berfungsi sebagai penghubung antar developer dan konsumen untuk mendeskripsikan sebuah sistem, selain sebagai penghubung usecase juga dapat menggambarkan proses yang akan dilakukan sistem dan juga mampu menampilkan urutan aktivitas dari sebuah sistem. Manfaat dari use case sendiri adalah dapat mmeberi gambaran interface dari sebuah sistem dan juga dapat menentukan siapa yang dapat beinteraksi dengna sistem dan apa saja yang dapat dilakukan terhadap sistem. Komponen – komponen usecase diagram terdiri dari beberapa item yaitu:

Simbol	Keterangan
 <p>Actor</p>	<p>Aktor: mewakili peran yang diambil orang luar ketika berinteraksi dengan sistem. Misalnya pelanggan atau admin.</p>
 <p>Use Case</p>	<p>Use case: menggambarkan interaksi aktor dengan sistem atau menjelaskan fungsionalitas sistem yang digunakan aktor.</p>
	<p>Association: penghubung antar aktor dan use case, hal ini menunjukkan bahwa seorang aktor dapat menggunakan fungsi tertentu dari sistem.</p>
	<p>Generalisasi: menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case</p>
	<p>Include: menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya.</p>
	<p>Extend: menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi.</p>







Gambar 2.2 Simbol Use Case Diagram



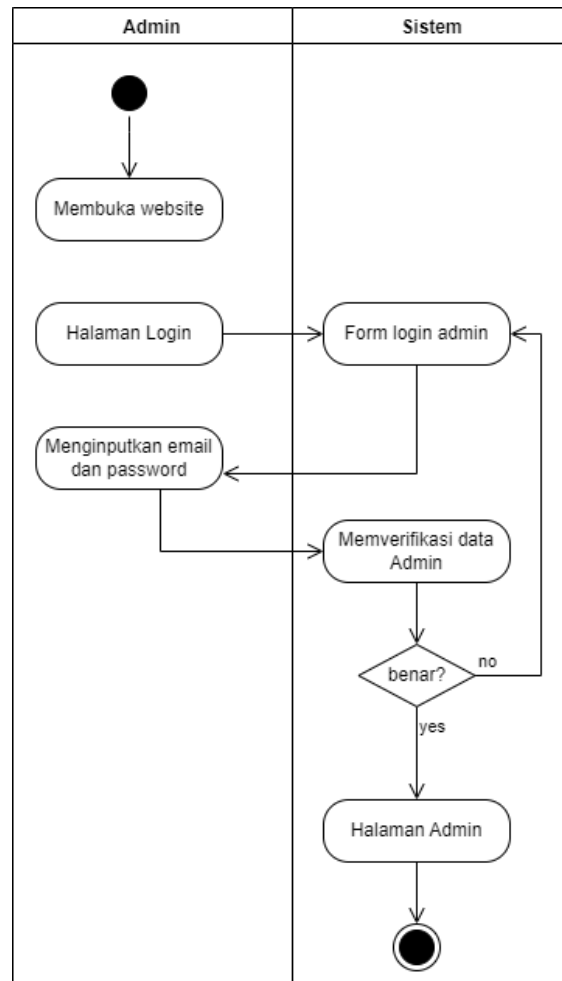
Gambar 2.3 Contoh Use-case Diagram

2.2.4.2 Activity Diagram

Activity diagram adalah sebuah diagram yang menunjukkan aktivitas yang terjadi pada sebuah sistem yang akan dibuat. Activity diagram digunakan untuk mendeskripsikan proses yang menggambarkan fungsionalitas sistem[19]. Activity diagram memiliki tujuan agar mempermudah orang awam untuk memahami aktivitas yang ada pada sistem yang telah dirancang sedetail mungkin, selain itu activity diagram juga berfungsi untuk melihat aktivitas user yang berlangsung berdasarkan rancangan use case diagram yang dirancang sebelumnya. Komponen – komponen Activity diagram terdiri dari beberapa item yaitu:

Simbol	Keterangan
	<p>Initial node: digunakan untuk menandakan status awal, tindakan awal, atau titik awal aktivitas untuk setiap activity diagram</p>
	<p>Final state: notasi akhir digunakan untuk menandakan proses tersebut berakhir</p>
	<p>Activity: aktivitas yang dilakukan atau sedang terjadi dalam sistem. Biasanya diawali dengan “kata kerja” dari aktivitas yang dilakukan.</p>
	<p>Decision: digunakan untuk menggambar suatu keputusan atau Tindakan yang harus diambil pada kondisi tertentu.</p>
	<p>Control flow: menunjukkan rangkaian pelaksanaan.</p>
	<p>Swimlane: memecah activity diagram menjadi kolom dan baris untuk membagi tanggung jawab objek yang melakukan suatu aktivitas.</p>


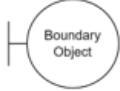





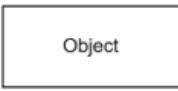
Gambar 2.4 Simbol Activity Diagram



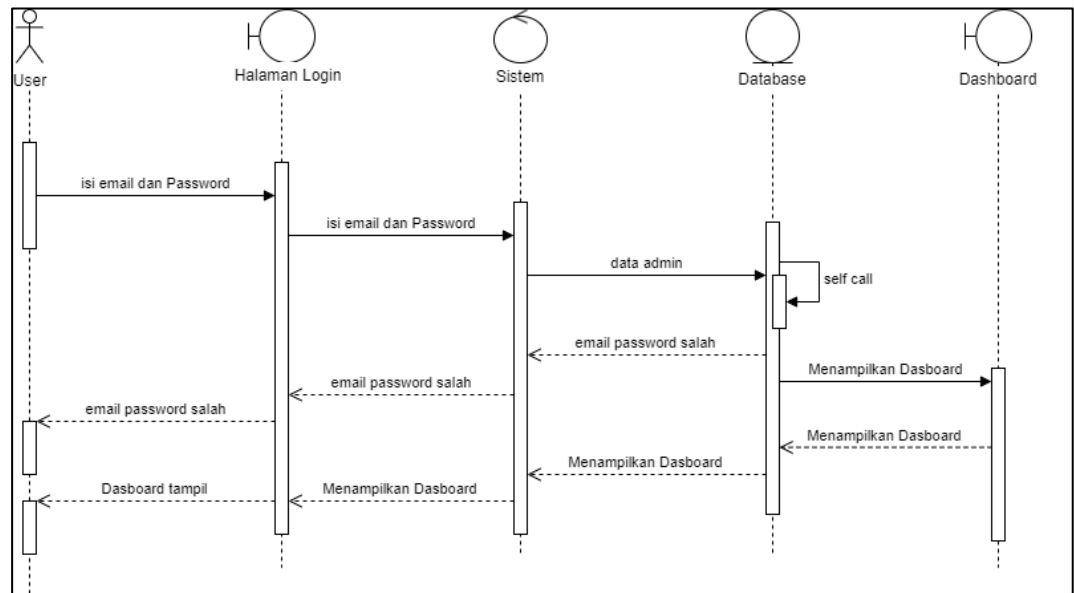
Gambar 2.5 Contoh Activity Diagram

2.2.4.3 Sequence Diagram

Sequence diagram adalah suatu diagram yang berfungsi untuk menjelaskan dan menampilkan ikatan antar objek dengan sebuah sistem secara detail[18]. Sequence diagram dapat digunakan untuk menggambarkan langkah-langkah yang akan dilakukan oleh pengguna yang menghasilkan luaran tertentu. Sequence diagram biasanya dibuat setelah pembuatan activity diagram yang berfungsi untuk memicu proses apa saja yang terjadi didalam sistem. Tujuan dari pembuatan sequence diagram ialah untuk mengetahui runtunan kejadian yang dapat menghasilkan output yang diinginkan[18]. Berikut merupakan komponen – komponen yang digunakan untuk membuat sequence diagram:

Simbol	Keterangan
 <p>Actor</p>	Komponen ini menggambarkan seorang pengguna (user) yang berada di luar sistem dan sedang berinteraksi dengan sistem. Dalam
 <p>Boundary Object</p>	Boundary Object menggambarkan sebuah penggambaran dari form.
 <p>Entity Object</p>	Entity Object adalah elemen yang bertanggung jawab menyimpan data atau informasi, dapat berupa beans atau model object.
 <p>Control Object</p>	Control Object memiliki tanggungjawab terhadap objek yang memiliki logika
	Lifeline ini biasanya memiliki kotak yang berisi objek yang memiliki fungsi untuk menggambarkan.
	<i>activation box</i> ini merepresentasikan waktu yang dibutuhkan suatu objek untuk menyelesaikan tugasnya.
 <p>self call</p>	Message to self menggambarkan hubungan dengan objek itu sendiri yang menunjukkan kejadian itu sendiri.
 <p>Object</p>	Objek biasa digunakan untuk mendokumentasikan perilaku sebuah oboek pada sebuah sistem.

Gambar 2.6 Simbol Sequence Diagram



Gambar 2.7 Contoh Sequence Diagram

2.2.4.4 Class Diagram

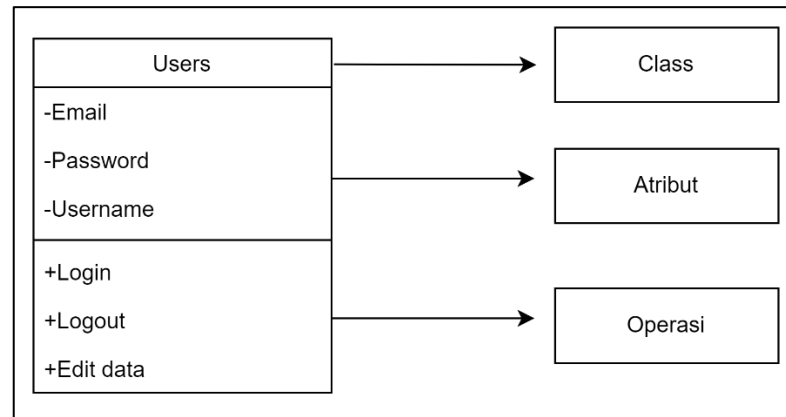
Class diagram termasuk salah satu *UML (Unified Modeling Language)* yang berfungsi untuk menjelaskan suatu model data sebagai gambaran yang diterapkan pada sistem [18]. *Class diagram* juga dapat memberi gambaran struktur dengan jelas. Berikut merupakan komponen-komponen penyusun class diagram:

- **Komponen Atas:** Pada komponen bagian atas merupakan nama class.
- **Komponen Tengah:** Berisikan atribut dari class. Atribut dapat ditulis lebih detail dengan memasukkan tipe data.
- **Komponen Bawah:** Pada bagian bawah menyertakan operasi yang memberi gambaran bagaimana suatu class berinteraksi dengan data.

Selain komponen penyusun class diagram, ada pula hubungan antar kelas yang akan dijelaskan dibawah ini:

- **Asosiatif:** yaitu suatu hubungan antar dua *class* secara statis.
- **Agregasi:** suatu hubungan antar dua *class* yang merupakan bagian dari *class* lain dan keduanya tidak dapat berdiri sendiri.

- **Generalisasi (Pewarisan):** yaitu kemampuan untuk mewarisi seluruh atribut dan metode dari class asalnya (superclass) ke *class* lain (subclass).



Gambar 2. 8 Contoh Class Diagram

2.2.4.5 Entity Relationship Diagram (ERD)

ERD atau *Entity Relationship Diagram* digunakan untuk perancangan database yang terlibat dalam hubungan antar objek atau entitas yang digunakan untuk mengembangkan desain basis data awal. ERD menggunakan tiga prinsip jenis elemen yaitu set entitas, atribut, dan relasi(hubungan)[20].

- Satu-ke-satu (*one to one*)

Suatu entitas di A dikaitkan dengan paling banyak satu entitas di B, dan sebuah entitas di B berhubungan dengan paling banyak satu entitas di A[20].

- Satu-ke-banyak (*one to many*)


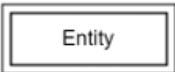





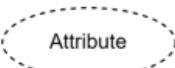
Entitas dalam A dikaitkan dengan beberapa (nol atau lebih) entitas di B. Entitas dalam B, hanya dapat dikaitkan dengan paling banyak satu entitas di A[20].

- Banyak-ke-satu (*Many to one*)

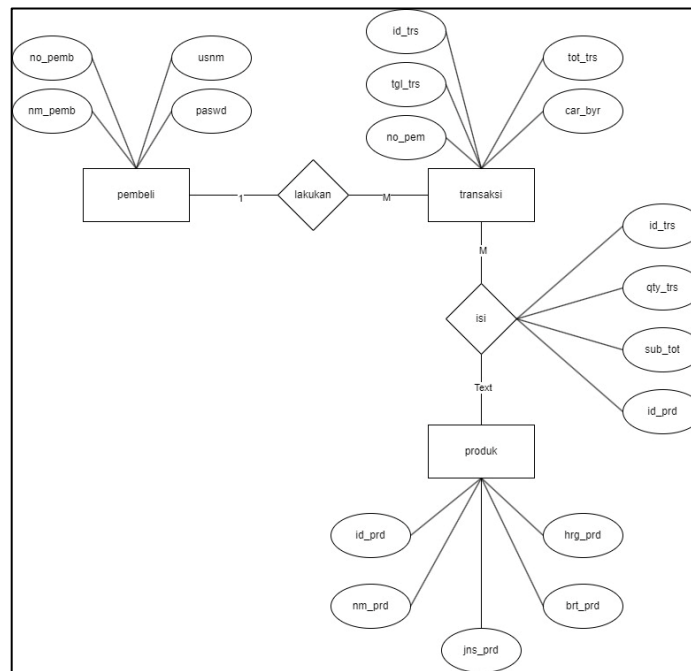
Sebuah entitas di A berhubungan dengan paling banyak satu entitas di B. Sebuah entitas dalam B, dapat dikaitkan dengan beberapa (nol atau lebih) dari entitas di A[20].

- Banyak-ke-banyak (*many to many*)

Entitas dalam A dikaitkan dengan beberapa (nol atau lebih) entitas dalam B, dan entitas dalam B juga dapat dikaitkan dengan beberapa (nol atau lebih) entitas di A[20].

Simbol	Keterangan
	Entitas Kuat
	Entitas Lemah
	Entitas Asosatif
	Relasi
	Relasi Pengidentifikasi
	Atribut
	Atribut Bernilai Banyak
	Atribut Turunan

Gambar 2. 9 Simbol ERD



Gambar 2. 10 Contoh ERD

2.2.4.6 Wireframe

Wireframe merupakan rancangan awal pada sebuah website atau aplikasi sebelum didesain. Wireframe merupakan salah satu tahapan yang penting dalam merancang atau mendesain aplikasi yang harus dipahami dengan benar. Wireframe tahapan penting sebelum letak letak fitur pada suatu aplikasi dan desain aplikasi di buat[21].

2.2.5. Android OS

Android OS adalah sebuah sistem operasi perangkat mobile yang menyediakan berbagai macam aplikasi dan juga menjadi alat komunikasi dengan perangkat yang berbeda [22]. Android memiliki beberapa versi dan memiliki nama masing – masing saat pertama kali muncul dari versi awal 1.0 hingga versi 11.0[23]. Android merupakan sistem operasi yang berbasis Linux yang mengelola dan mendukung pengoperasian perangkat keras seperti *tablet* atau *smartphone*. Jadi, *Android* bekerja dengan cara menyentuh, menggeser atau tap layar perangkat.

Android adalah sumber terbuka yang bebas digunakan, serta dapat dimodifikasi dan didistribusikan oleh perangkat lunak atau pengembang apa

pun. Karena bersifat sumber terbuka pengguna dapat membuat aplikasi baru di dalamnya, perusahaan teknologi bebas menggunakan sistem operasi ini di perangkatnya tanpa lisensi. Pembuat aplikasi juga diizinkan membuat aplikasi menggunakan kode sumber yang diterbitkan oleh Google.

2.2.6. Flutter

Flutter adalah *SDK (Software Development Kit)* yang dikembangkan Google yang bertujuan untuk menyederhanakan pengembangan *software* multiplatform dengan satu codebase. Hal ini juga berlaku untuk pemisahan antarmuka pengguna (UI) serta code yang sering terdapat pada bermacam bahasa pemrograman yang lain. Pengimplementasian code pada flutter menggunakan widget yang berupa komponen - komponen penting sebagai pendukung untuk menampilkan tampilan pada aplikasi. flutter memiliki kelebihan diantara multiplatform lain yaitu flutter tidak terlalu membutuhkan penyambung seperti multiplatform lain, bahasa pemrograman dari flutter lebih simple dan lebih mudah dimengerti dibandingkan dengan bahasa pemrograman multiplatform lain.[24]

2.2.7. Firebase

Firebase merupakan suatu layanan google yang menyediakan tools dan layanan yang dapat membantu dalam mengembangkan aplikasi *android* menjadi lebih cepat[25]. Firebase memiliki beberapa fitur yang dapat digunakan dalam pengembangan aplikasi salah satunya fitur *Realtime Database* yang digunakan untuk menyimpan data dan melakukan sinkronisasi secara *realtime* [26].

2.2.8. API (Application Programming Interface)

API merupakan antarmuka yang digunakan buat mengakses aplikasi ataupun layanan dari suatu program. *API* membolehkan pengembang buat mengenakan guna yang telah terdapat dari aplikasi lain sehingga tidak butuh membuat ulang dari dini. Tujuan pemakaian dari *API* merupakan buat silih berbagi informasi antar aplikasi yang berbeda, tujuan pemakaian *API* yang lain ialah buat memusatkan proses pengembangan aplikasi dengan metode

sediakan suatu *function* yang terpisah sehingga pengembang tidak butuh lagi merancang fitur yang seragam[27].

2.2.9. Blackbox testing

Blackbox testing dilakukan untuk menguji perangkat lunak yang berfokus pada fungsi dari system yang telah dibangun seperti kekurangan pada suatu program yang dibuat. Black Box Testing memiliki kelebihan yaitu penguji tidak perlu tau bahasa pemrograman yang digunakan, penguji hanya menuliskan apakah program berfungsi sesuai dengan yang diharapkan [28].

Kelebihan penggunaan *blackbox testing* adalah pengevaluasi program tidak perlu paham tentang Bahasa pemrograman yang digunakan pada program yang dibuat, pengetesan program dilakukan oleh pengguna, yang bertujuan agar dapat mengungkapkan ketidakpastian dalam spesifikasi syarat yang telah ditentukan.

Kekurangan penggunaan *blackbox testing* salah satunya adalah tidak menguji pada bagian *backend* sama sekali dan sering kali terjadi pengulangan pengujian yang dilakukan oleh pembuat program.