BAB II
LANDASAN TEORI

2.1 *Mobile*

Arfida, Amnah & Wibowo (2018:13) menjelaskan bahwa *mobile* adalah teknologi komunikasi yang memiliki keunggulan dimana data dapat dibawa kemana mana karena ukurannya yang kurang lebih sebesar genggaman tangan. *Mobile* memiliki beberapa penjelasan, salah satunya *mobile* merupakan kemajuan teknologi komputer sehingga dapat berkomukasi menggunakan jaringan tanpa menggunakan kabel dan dibawa atau berpindah tempat.

Kata *mobile* mempunyai arti bergerak atau berpindah, sehingga aplikasi *mobile* adalah sebutan untuk aplikasi yang berjalan di *mobile device* . Dengan menggunakan aplikasi *mobile,* dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, berjualan, belajar, mengerjakan pekerjaan kantor, *browsing* dan lain sebagainya.

2.2 *Android*

Arfida, Amnah & Wibowo (2018:52) menjelaskan bahwa *Android* merupakan sebuah sistem operasi yang berbasis *linux* untuk perangkat portable seperti *semartphone* dan komputer tablet. Sistem operasi dapat diilustrasikan sebagai jembatan antara piranti (*device*) dan penggunanya, sehingga pengguna bisa berinteraksi dengan *device*-nya dan menjalankan aplikasi – aplikasi yang tersedia pada *device*. *Android* juga menyediakan *platform* terbuka (*open source*) bagi para pengembang untuk mengembangkan aplikasi pada berbagai perangkat dengan sistem operasi *android*.

Firly (2018:2) menjelaskan bahwa *android* pertama kali dirilis pada bulan oktober 2003 oleh Andy Rubin, Rich Miner, Nich Sears dan Chris White di bawah sebuah perusahaan bernama *android Inc* di Palo Antom, California. Dan akhirnya *android* diakuisisi oleh *google* pada tahun 2005. 5 november 2007 adalah kali pertama *android* meluncurkan versi *beta* yang bersamaan dengan berdirinya *open*

handset alliance atau OHA. *Android* sendiri memiliki beberapa kelebihan yaitu adalah sebagai berikut :

a. *Open Source* alias Gratis
b. Cepat dan *Responsive*
c. *User Friendly*
d. Variasi harga produk yang beragam
e. *Google* sebagai pengembang
f. *Hardware* pendukung yang beragam

2.3 Perangkat Lunak Pengembangan Sistem

Pengembangan sistem untuk membangun aplikasi pemesanan berbasis *mobile androi* diperlukan beberapa perangkat lunak yang digunakan dalam membangun aplikasi tersebut. Beberapa perangkat lunak yang digunakan adalah sebagai berikut :

2.3.1 *Android studio*

Firly (2018:13) menjelaskan bahwa *Android studio* merupakan *integrated development environment* (*IDE*) atau dalam artian lain adalah sebuah lingkungan pengenbangan terintegrasi resmi yang memang merancang khusus untuk pengembangan system operasi *Google Android*. Aplikasi ini dibangun di atas sebuah perangkat lunak yang dinamakan *IntelliJ IDEA* milik *JetBrains*. Bisa juga dibilang bahwa *android* studio merupakan pengganti dari *Eclipse android development tool* atau *ADT* sebagai *IDE* utama dalam pengembangan aplikasi *android* yang asli.

Android studio diluncurkan pada tanggal 16 mei 2013 dalam *konferensi google I/O* yang pada saat itu masih dalam tahap pratinjau akses *versi* 0.1 sebagai perintis. Hingga pada akhirnya *versi* stabil 3.0 yang *liris* pada pertengahan bulan oktober 2017 dan menjadi *software* terlaris dikalangan *developer* muda. Aplikasi ini dapat digunakan diberbagai sistem operasi yaitu *windows,linux* dan *macOS*.

Aplikasi ini menawarkan berbagai fitur canggih yang akan meningkatkan kemampuan *produktivitas* dalam proses pengembangan aplikasi. Berikut ini adalah beberapa hal yang akhirnya banyak mengundang *developer* untuk melirik *android studio* sebagai *software* pengembang :

a. Dukungan dari C++, *NDK* dan sekarang *kotlin*
b. Perkembangan yang *up to date*
c. Sistem berbasis *Gradle* yang dinilai fleksibel
d. Lingkungan yang mencakup seluruh perangkat *android*
e. *Emulator* yang cepat dan kaya akan fitur
f. Alat pengujian dan kerangka yang juga *ekstensif*
g. *Instant Run*
h. Dukungan *google cloud platform*

2.3.2 *Java*

Firly (2018:19) menjelaskan bahwa *Java* adalah bahasa pemrograman *multi platform*. *Java* tidak menyediakan *IDE* khusus seperti halnya bahasa pemrograman yang lain. Pemrogram bisa menggunakan *IDE* yang support ke *java*, misalnya *Netbeans*, *Eclips*, *TexPad*, dan lain-lain. elemen-elemen dasar pemrograman Java terdiri dari Himpunan karakter, Pengenal (*identifier*), Kata Kunci, Tipe Data *Primitif*. Tipe data *primitif* yang didukung oleh bahasa pemrograman *Java* adalah *byte*, *short*, *int*, *long*, *float*, *double*, *Boolean*, *char*.

2.3.3 *Firebase*

Firebase Adalah layanan dari *google* yang digunakan untuk mempermudah para pengembang aplikasi dalam pengembangan aplikasi. Dengan adanya *firebase*, *pengembang* aplikasi bisa fokus mengembangkan aplikasi tanpa harus memberikan usaha yang besar. Dua fitur yang menarik dari *firebase* yaitu *firebase remote config* dan *firebase realtime database*. Selain itu terdapat fitur pendukung untuk aplikasi yang membutuhkan pemberitahuan yaitu *firebase notification*. *Firebase Database* merupakan penyimpanan basis data *non*SQL yang memungkinan untuk menyimpan beberapa tipe data. Tipe data itu antara lain

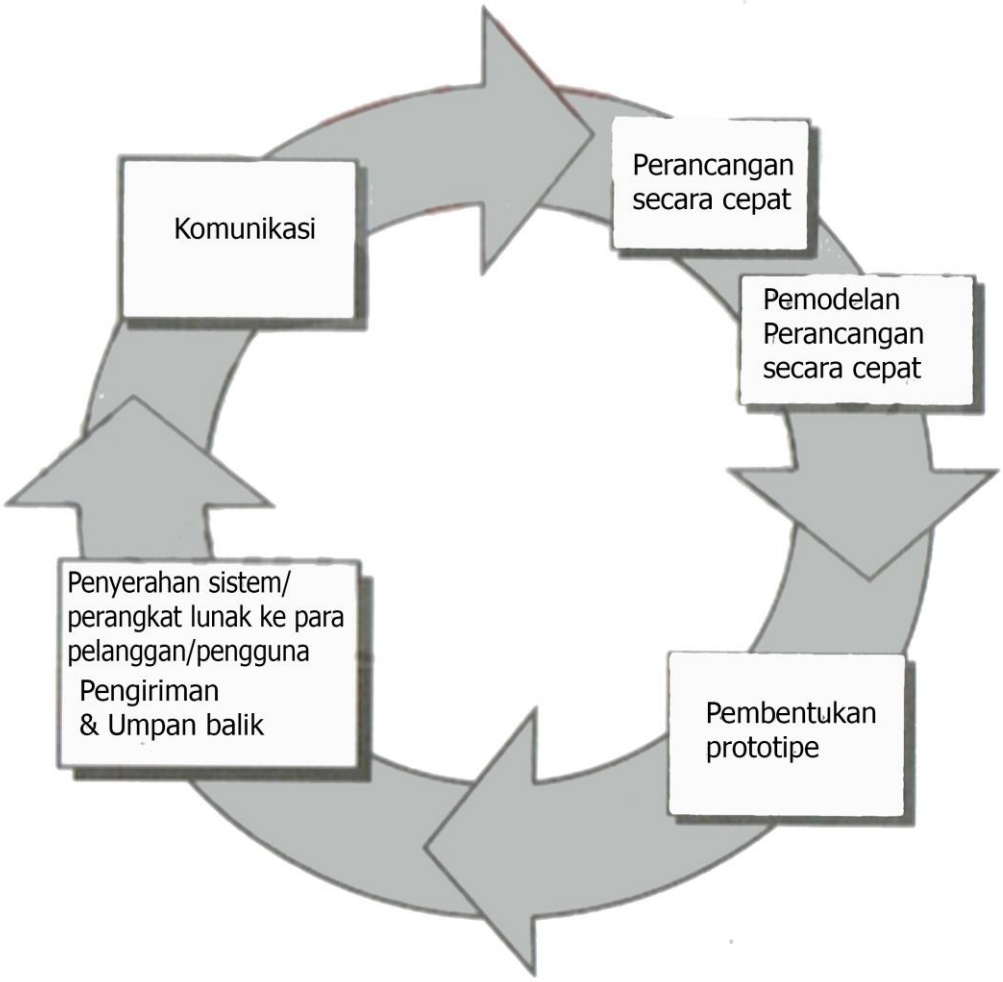
String, *Long*, dan Boolean. Data pada *Firebase Database* disimpan sebagai objek *JSON* tree. Ketika ada penambahan data, data tersebut akan menjadi node pada struktur *JSON*. Pada pengembangan aplikasi, layanan lainnya yang digunakan pada pengembangan aplikasi adalah *Firebase Storage*. Layaknya sebuah penyimpanan awan, *Firebase Storage* memungkin pengembang untuk mengunggah atau mengunduh sebuah berkas. Pada pengembangan aplikas. (*id.m.wikipedia.org/wiki/firebase*)

2.3.4 *Adobe XD*

Adobe XD adalah perangkat lunak yang bisa digunakan oleh para *desainer* aplikasi *mobile*. *Adobe xd* bisa memudahkan *desainer* aplikasi *mobile* dalam pengembangan *UX/UI*, *adobe xd* ini sudah menyediakan fitur *UI* Desain dan juga *UX* Desain sebagai *prototype* tanpa membutuhkan *third-party* atau aplikasi lain untuk membantu membuat sebuah *prototype*. (*garudapixel.com, dewaweb.com*)

2.4 Metode Pengembangan Perangkat Lunak

Pressman (2010:51) menjelaskan meskipun pembuatan *prototype* dapat di gunakan sebagai model proses yang berdiri sendiri, pembuatan protoipe lebih umum digunakan sebagai teknik yang dapat diimplementasikan di dalam konteks setiap model proses perangkat lunak, dan paradigma pembuatan *prototype* seringkali membantu tim pengembang perangkat lunak dan para *stakeholder* untuk memahami lebih baik apa yang akan dikembangkan saat spesifikasi kebutuhan belum jelas.



Gambar 2.1 Tahapan Pengembangan Aplikasi

Berikut ini adalah penjelasan dari gambar 2.1

1. Komunikasi

   Tahap komunikasi ini adalah tahapan komunikasi antara *developer* dan pelanggan mengenai tujuan pembuatan *software*, mengidentifikasi apakan kebutuhan diketahui.

2. Perancangan secara cepat

   Tahap perancangan secara cepat ini adalah tahapan perancangan cepat setelah terjalin komunikasi.

3. Pemodelan perancangan secara cepat

Tahap pemodelan perancangan secara cepat ini adalah tahapan segera membuat model, dan pemodelan cepat fokus pada gambaran dari segi *software* apakah *visible* menurut pelanggan.

4. Pembentukan prototipe

Tahap pembentukan prototipe ini adalah tahapan pemodelan cepat menuntun pada pembuatan dari prototipe.

5. Penyerahan sistem / perangkat lunak ke para pelanggan / pengguna pengiriman dan umpan balik

Tahap penyerahan sistem / perangkat lunak ke para pelanggan / pengguna pengiriman dan umpan balik ahapan ini adalah prototipe yang dikirimkan kemudian dievaluasi oleh pelangan, umpan balik digunakan untuk menyaring kebutuhan untuk program.

2.5 Pengujian Kotak Hitam

Pressman (2010:597) menjelaskan bahwa pengujian kotak hitam, juga disebut pengujian perilaku, yang berfokus pada persyaratan fungsional perangkat lunak. Pengujian kotak hitam memungkinkan untuk membuat beberapa kumpulan kondisi masukan yang sepenuhnya akan melakukan semua kebutuhan fungsional untuk program. Pengujian kotak hitam berupaya untuk menemukan kesalahan dalam kategori berikut : (1) fungsi yang salah atau hilang, (2) kesalahan antarmuka, (3) kesalahan dalam struktur data atau akses basis data eksternal, (4) kesalahan perilaku atau kinerja, dan (5) kesalahan inisialisasi dan penghentian.

2.6 *Unifed Modelling language* (UML)

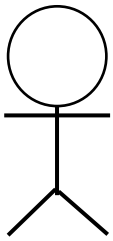
Rosa (2018:133) menjelaskan bahwa *Unifed Modeling Language* (UML) adalah metode perancangan berortasi objek yang memeriksa syarat – syarat dari sudut pandang kelas-kelas dan objek yang ditemui pada ruang lingkup permasalahan dengan tujuan untuk memahami domain masalah dan meningkatkan ketelitian,

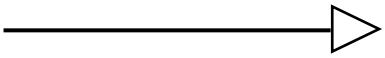
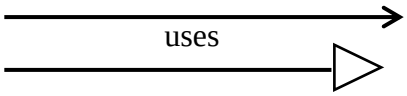
konsistensi, kelengkapan analisis. *Unified Modeling Language* (UML) adalah satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek.

2.6.1 *Use Case Diagram*

Rosa & Shalahuddin (2018:155) menguraikan bahwa *Use Case Diagram* merupakan deskripsi peringkat tinggi bagaimana perangkat lunak (aplikasi) akan digunakan oleh penggunanya. Selanjutnya use case tidak hanya penting pada tahap analisis, tetapi juga sangat penting untuk perancangan, untuk mencari kelas-kelas yang terlibat dalam aplikasi, serta untuk melakukan pengujian. Diagram use case bersifat statis, diagram ini memperlihatkan himpunan use-case dan aktor-aktor (suatu jenis khusus dari kelas). Simbol – simbol *Use Case* dapat dilihat pada tabel 2.1

Tabel 2.1 Simbol *Use Case Diagram*

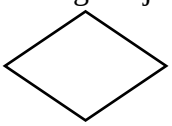
Simbol	Keterangan
Use Case 	Fungsionalitas yang disediakan sistem sebagai unit – unit yang saling bertukar pesan antar unit dan aktor.
Aktor / *Actor* 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi.
Asosiasi / *Association* 	Komunikasi antar aktor dan *Use Case* yang berpartisipasi.
Ekstensi / *extend* <<extend>> 	Relasi Use Case tambahan ke sebuah Use Case dimana Use Case yang ditambah dapat berdiri sendiri walau tanpa *Use Case* tambahan.

Generalisasi / generalization 	Hubungan generalisasi dan spesialisasi antara dua buah *Use Case* yang mana fungsi yang satu lebih umum dari yang lainnya.
Include / Use Case <<include>>  uses	Relasi *Use Case* tambahan ke sebuah *Use Case* dimana *Use Case* yang ditambahkan memerlukan *Use Case* ini untuk menjalankan fungsinya.

2.6.2 *Activity Diagram*

Rosa & Shalahuddin (2018:161) menguraikan bahwa *Activity diagram* menggambarkan workflow (alir kerja) atau aktivitas dari sebuah system atau proses bisnis atau menu yang ada pada perangkat lunak. Tahap perancangan activity diagram menjabarkan masing – masing activity pada perancangan use case. Simbol – simbol *Activity Diagram* dapat dilihat pada tabel 2.2.

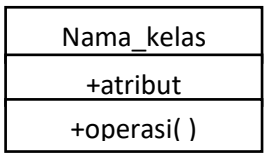
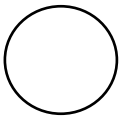
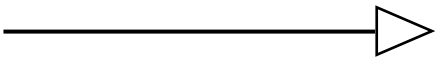
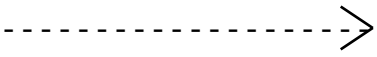
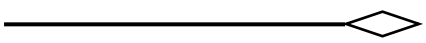
Tabel 2.2. Simbol *Activity Diagram*

Simbol	Keterangan
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Status awal 	Bagaimana objek dibentuk atau diawali.
Status akhir 	Bagaimana objek dibentuk dan diakhiri.
Percabangan / join 	Asosiasi percabangan dimana jika ada pilihan aktifitas lebih dari satu.
Penggabungan / join 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu

2.6.3 *Class Diagram*

Gunawan & Sari (2017:319) menguraikan bahwa *Class diagram* menggambarkan atribut / poperti suatu system, sekaligus menawarkan layanan untuk memanipulasi keadaan metode atau fungsi. Class diagram menggambarkan struktur dan deskripsi class, package, dan objek beserta hubungan satu sama lain. Simbol – simbol yang ada pada *Class Diagram* ditunjukan oleh Tabel 2.3.

Tabel 2.3. Simbol *Class Diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur system.
Antarmuka / *interface*  **Nama_*interface***	Sama dengan konsep *interface* dalam pemograman berorientasi objek.
Asosisasi / *association*	Relasi antarkelas dengan makna umum, asosia biasanya disertai dengan *multiplicity*.
Asosisasi berarah / *directed association* 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity*.
generalisasi 	Relasi antarkelas dengan makna Generalisasi – spesialisasi (umum khusus).
Kebergantungan / *dependency* 	Relasi antarkelas dengan makna kebergantungan antarkelas.
Agregasi / *aggregation* 	Relasi antar kelas dengan makna semua – bagian

2.7 Penelitian Terdahulu

Berikut merupakan jurnal terkait dengan penelitian terdahulu :

No	Nama	Judul	Terbit / Tahun	Keterangan
1	Whildan syaihul fikri, Eka whidi Y dan Yogie indra K	Aplikasi Pemesanan Cetak Foto Online (Studi Kasus: Studio FotoXYZ)	Konferensi Nasional ICT-M Politeknik Telkom (KNIP) / 2012	Penelitian ini membahas mengenai Aplikasi pemesanan cetak foto online yang berada di studio foto XYZ, sehingga mampu membuat proses pemesanan cetak foto lebih cepat tanpa pelanggan harus datang ke studio foto dan dapat membantu perusahaan dalam pengolahan data pemesanan cetak foto.
2	Asilah salma, Irfan darmawan & Faisal mufied	Perancangan Aplikasi Callme Berbasis Android menggunakan metode Prototyping (Modul administrasi Customer dan Admin)	e-Proceeding of Engineering / 2017	Penelitian ini membahas mengenai perancangan aplikasi callme berbasis android untuk mempermudah dalam pemesanan makanan dan menghemat biaya costumer karena tidak melalui pihak ketiga. Metode yang digunakan dalam perancangan aplikasi ini adalah metode Prototyping.
3	Surawijaya & Eko budi	Aplikasi Mobile Driver Online Berbasis Android Untuk Perusahaan Rental Kendaraan	Ultima InfoSys / 2017	Penelitian ini membahas mengenai Aplikasi mobile Driver online berbasis android untuk membantu perusahaan rental kendaraan dalam membagi jadwal driver, dapat memantau driver apabila keluar jalur, dan mempasilitasi bagi para pelanggan dalam melakukan pemesanan.
4	Raharjo, Hidayat dan Yudato	Aplikas Penyedia Layanan Jasa Tukang Bangunan	JTET(Jurnal Teknik Elektro Terapan) Vol.6 No.2 Agustus 2017	Penelitian ini membahas mengenai perlunya disediakan aplikasi mobile yang mampu menghubungkan tukang-tukang bangunan dan masyarakat melalui perangkat *smartphone*, karena sulitnya mencari tenaga tukang

				bangunan yang berpengalaman dan bisa dipercaya
5	Nuzula	System Pelayanan dan Pemesanan Online Pada Toko Bangunan Sumarno Jaya Depok	Jurnal String Vol.2 No.3 April 2018	Penelitian ini membahas mengenai media internet diharapkan mampu memberikan informasi kepada orang banyak untuk mengetahui keberadaan Toko bangunan sumarno jaya, tujuannya untuk mempermudah para konsumen untuk melakukan pelayanan dan pemesanan secara tidak langsung ke toko untuk berbelanja produk bangunan.