

BAB III

METODE PENELITIAN

3.1 Diagram Alir Penelitian

Metode penelitian pada perancangan aplikasi pemantau pedagang keliling adalah dengan menggunakan metode *waterfall*. Metode *waterfall* adalah metode klasik yang sistematis. Tahapan yang digunakan untuk membangun aplikasi yaitu studi literatur, analisis kebutuhan, perencanaan aplikasi, pembuatan model dan desain, pembuatan kontruksi dan algoritma, pengujian dan analisis program.

Studi literatur, diperlukan untuk menentukan apa yang akan dibuat dan konsep apa yang akan digunakan. Studi literatur berfungsi untuk menambah wawasan dalam membuat perancangan alat. Studi literatur dilakukan dengan mencari penelitian penelitian terdahulu yang bertema seperti android, GPS, *tracking* dan sistem telemetri

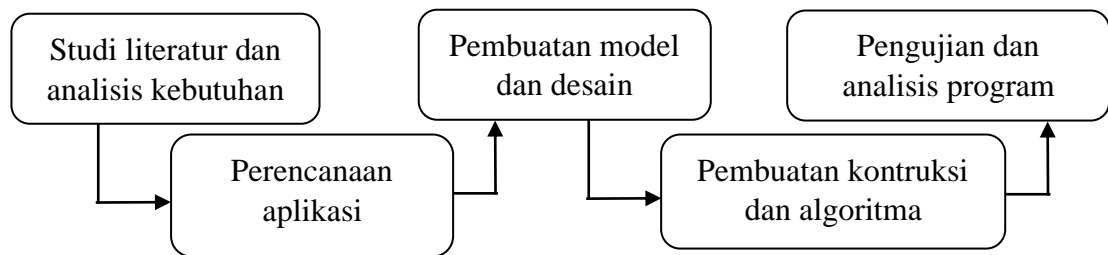
Analisis kebutuhan, diperlukan untuk menentukan apa saja yang dibutuhkan untuk membuat sebuah aplikasi pemantauan pedagang keliling. Analisis kebutuhan dilakukan untuk menganalisis apa saja yang dibutuhkan aplikasi pemantau pedagang keliling setelah membaca penelitian-penelitian sebelumnya.

Perencanaan aplikasi, dilakukan untuk merencanakan aplikasi pemantau pedagang keliling. Perencanaan aplikasi bertujuan untuk mematangkan rencana aplikasi yang akan dibuat dan merencanakan apa saja yang akan dilakukan sebelum melakukan pembuatan aplikasi.

Pembuatan model dan desain, dilakukan untuk menentukan model dan desain yang akan diterapkan pada aplikasi.

Pembuatan kontruksi dan algoritma, dilakukan untuk mengaplikasikan model dan desain yang telah dibuat kepada algoritma dan coding. Pembuatan kontruksi dan algoritma bertujuan untuk mengaplikasikan model dan desain kedalam bahasa pemrograman.

Pengujian dan analisis digunakan sebagai tahapan akhir dalam mencari kesimpulan dan mencapai tujuan dari aplikasi yang telah dibuat. Tahapan tersebut direpresentasikan menjadi kerangka tahapan menggunakan diagram alir seperti Gambar 3.1.



Gambar 3.1 Diagram alir perancangan Aplikasi

3.1.1 Instrumen penelitian dan kebutuhan

Instrumen yang digunakan untuk menunjang perancangan aplikasi pada tugas akhir ini meliputi perangkat keras maupun perangkat lunak. Perangkat keras digunakan untuk membuat dan menjalankan sebuah aplikasi. Perangkat keras yang digunakan yaitu *smartphone* Android. *Smartphone* android digunakan sebagai alat uji coba aplikasi yang dibuat kemudian dilakukan analisis. Selain *smartphone* adapun laptop yang digunakan sebagai perancangan dan pembuatan aplikasi. Laptop juga digunakan sebagai monitor untuk *Smartphone* Android dan monitor data *real-time* pada database.

Tabel 3. 1 Perangkat keras yang digunakan

No	Alat	Keterangan
1	<i>Smartphone Android</i>	3 <i>Smartphone Android</i> , module GPS location simcard dan Internet
2	Laptop	2 Laptop, dengan OS Windows, Ram 16Gb

Tabel 3. 2 Perangkat lunak yang digunakan

No	<i>Software</i>	Keterangan
1	Android Studio IDE	Version 3.5.2
2	Mircrosoft Office	Version 2019
3	Akun Google <i>Maps</i>	Mendapatkan <i>Maps</i> SDK Android
4	Firestore API	<i>Real-time</i> Database
5	JAVA dan Kotlin	Bahasa Pemrograman
6	Adobe Photoshop	Version CC 2019

Perangkat lunak digunakan sebagai pembangun konstruksi dalam pembuatan aplikasi. Perangkat lunak yang digunakan dapat dilihat pada Tabel 3.2. Rincian alat perangkat keras dapat dilihat pada Tabel 3.1.

Dimas Ramadhan Amrulloh, 2020

PERANCANGAN APLIKASI PEMANTAU PEDAGANG KELILING
BERBASIS ANDROID

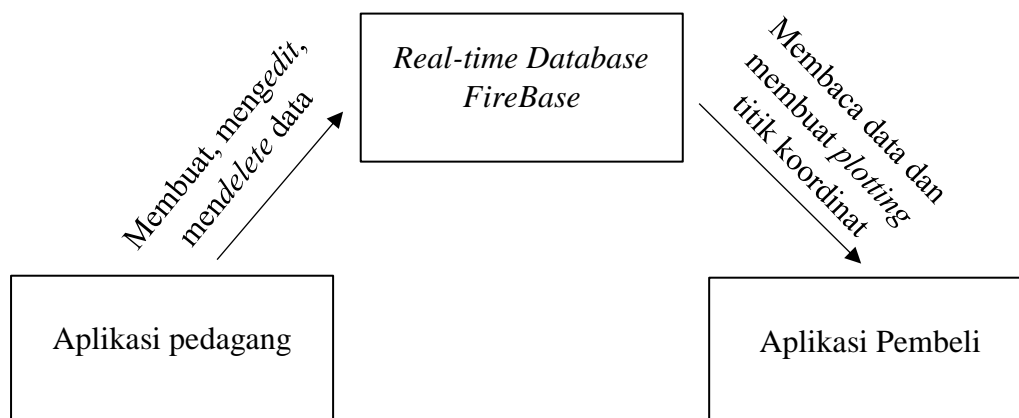
Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

3.2 Studi Literatur dan Analisis Kebutuhan

Studi literatur dilakukan dengan cara membaca beberapa sumber mengenai penelitian yang sudah dilaksanakan sebelumnya. Pada tugas akhir tentang Pemantau Pedagang Keliling menggunakan aplikasi Android. Pencarian dikhususkan pada beberapa sumber mengenai arsitektur aplikasi Android, Google Maps Api, Firebase dan sistem *GPS*.

3.3 Perancangan Aplikasi

Perancangan aplikasi pemantauan pedagang keliling menggunakan *smartphone* Android dilaksanakan dengan menentukan library Android yang akan digunakan agar aplikasi dapat bekerja sesuai dengan desain yang akan direalisasikan. Akan ada 2 (dua) aplikasi yang dibuat. yang pertama adalah aplikasi sebagai penjual yang memberikan koordinat dan aplikasi pembeli yang membaca koordinat. Diagram Blok Aplikasi dapat dilihat pada Gambar 3.2.



Gambar 3.2 Diagram Blok Aplikasi Pemantau pedagang keliling

3.4 Perancangan Desain Aplikasi

Dalam merancang aplikasi pertama dibutuhkan aplikasi Android Studio untuk menulis dan menerapkan algoritma dan desain yang telah dibuat sebelumnya. desain dibuat menggunakan aplikasi *Adobe photoshop* mulai dari *launcher*, *splash screen* dan *marker* yang akan dimunculkan pada titik yang ada di *Google Maps*. Aplikasi yang berperan sebagai penjual akan menggunakan lokasi pada GPS yang ada di *smartphone* mereka. Pada aplikasi telah tersimpan nomor telephone seluler yang kemudian akan dibaca oleh aplikasi yang berperan sebagai pembeli untuk menentukan koordinat penjual dan melakukan panggilan langsung melalui nomor yang tersedia. Spesifikasi pada *Android Studio IDE* dapat dilihat pada Tabel 3.1.

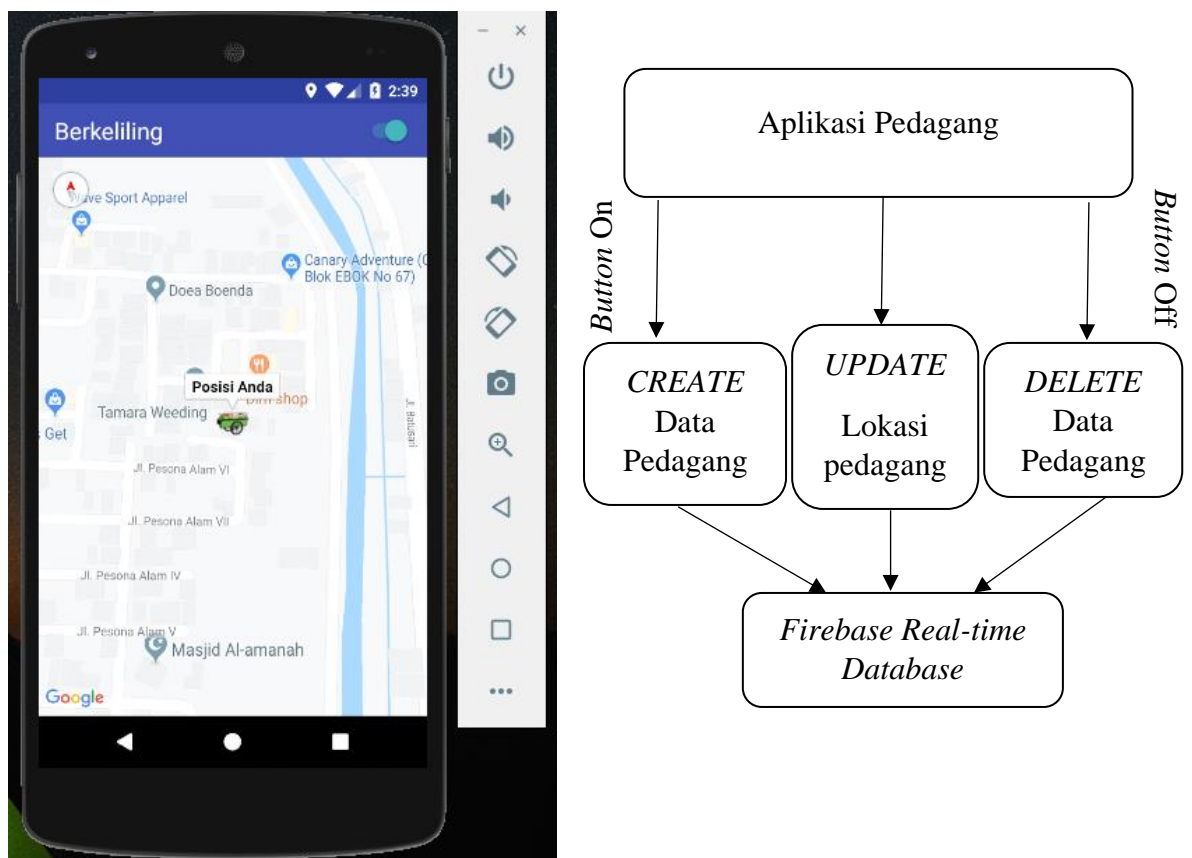
Tabel 3. 3 Spesifikasi plugin pada Android Studio IDE

Plugin	Version
<i>Kotlin</i>	1.2.61
<i>Build Gradle</i>	3.1.4
<i>Google Services</i>	4.0.1
<i>Compile SDK version</i>	27
<i>Constraint layout</i>	1.1.2
<i>Play service Maps</i>	15.0.1
<i>Firestore Database</i>	16.0.1
<i>Play Service Location</i>	15.0.1

Setelah menyusun plugin-plugin kemudian, memulai untuk membuat aplikasi dengan pedagang sebagai *CUD (Create, Update, Delete)* pada database, dan pembeli Sebagai *R (Read)* yaitu pembaca data data yang telah tersimpan pada database dan menambahkan fitur *call Phone* pada aplikasi yang berperan sebagai pembeli. Setelah menentukan pedagang mana yang dipantau maka pembeli dapat menghubungi pedagang yang diinginkan.

3.4.1 Aplikasi pemantau keliling dari sisi pedagang

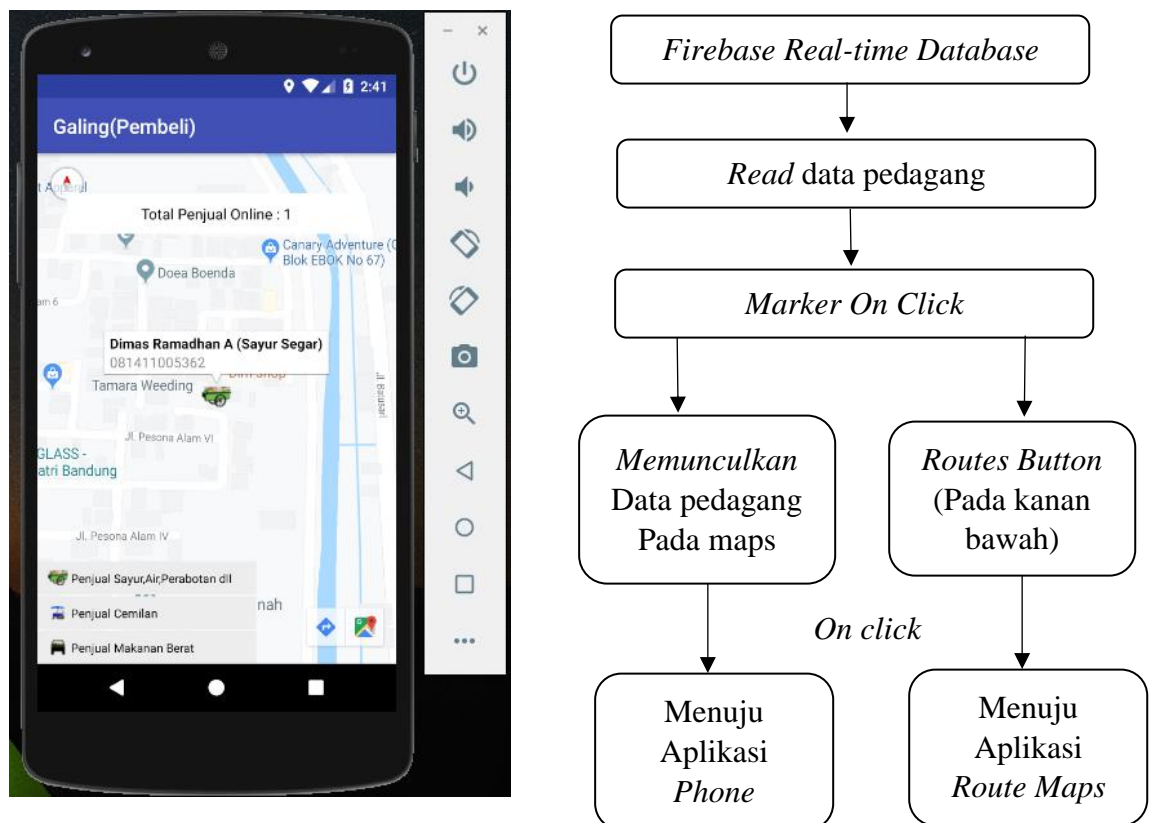
Pada aplikasi penjual hanya terdapat 1 *button* dan *screen* digunakan untuk menampilkan *marker*. Satu *button* digunakan untuk menandakan bahwa pedagang sedang beristirahat atau sedang berjualan. Jika *button* dalam posisi *on* (sedang berjualan), maka *marker* akan dibuat dan data dikirimkan ke database, ketika *button* tetap pada settingan berjualan, maka data akan terus *update latitude* dan *longitude* untuk mendapatkan titik yang sebenarnya. Jika *button* dalam keadaan *off* (sedang beristirahat) maka *marker* akan hilang karena data pada Firebase akan terhapus. Sistem ini berguna untuk menghemat sistem penyimpanan dari database. Desain dibuat seminimalis mungkin untuk memudahkan semua pedagang dalam menggunakan aplikasi. Desain tampilan untuk aplikasi pedagang dapat dilihat pada Gambar 3.3.



Gambar 3.3 Tampilan dan Flow aplikasi sisi penjual

3.4.2 Aplikasi aplikasi pemantau keliling dari sisi pembeli

Pada aplikasi pembeli desain dibuat dengan satu screen yaitu untuk pemantau pedagang pada *Google Maps*. Pada *screen* diberi sebuah legenda untuk memberikan informasi tentang klasifikasi *icon marker* pada pedagang yang sedang berkeliling. *Marker* pada aplikasi pembeli didapat dari data yang dibaca database *Firestore*. Kemudian data dituangkan pada *marker*. Ketika *marker* ditekan maka akan menampilkan nama dan barang yang dijual dan nomor telephone seluler. ketika ditekan lagi maka aplikasi akan berpindah ke aplikasi *phone* yang ada di *Android* untuk mengeksekusi nomor telephone seluler agar dapat berkomunikasi dengan pedagang. Selain itu Ketika *marker* ditekan akan menampilkan tanda panah dan *maps* pada kanan bawah *screen button* yang dapat ditekan kemudian aplikasi akan berpindah ke aplikasi *Google Maps* untuk menentukan *routes* dan jarak antara pembeli dan pedagang. Terdapat field pada atas *screen* untuk melihat berapa pedagang yang sedang berjualan. Desain aplikasi dapat dilihat pada Gambar 3.4.



Gambar 3.4 Tampilan dan Flow aplikasi Sisi Pembeli

Dimas Ramadhan Amrulloh, 2020

PERANCANGAN APLIKASI PEMANTAU PEDAGANG KELILING
BERBASIS ANDROID

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

3.4.3 Splash Screen Aplikasi

Splash Screen berfungsi sebagai opening aplikasi untuk mempercantik aplikasi tersebut. Selain itu splash screen juga dapat berfungsi sebagai loader, sehingga ketika aplikasi terkoneksi maka akan masuk ke menu utama setelah splash screen dapat dilihat pada gambar 3.5.



Gambar 3.5 Tampilan Splash Screen (a) tampilan pada aplikasi pembeli dan (b) tampilan pada aplikasi pedagang