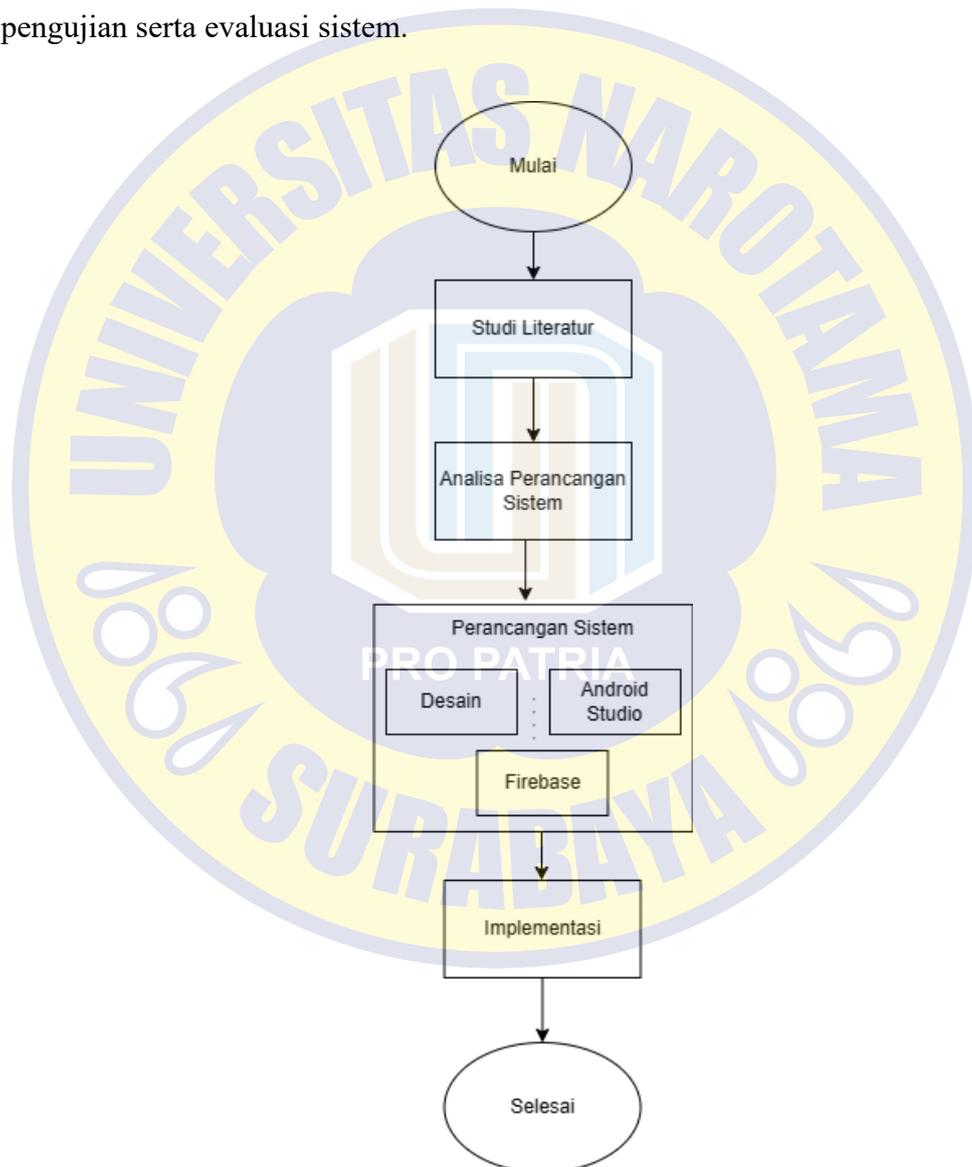


## BAB III

### METODOLOGI PENELITIAN

Pada proses penelitian kali ini, peneliti menggunakan metode Studi Literatur, Analisa Perancangan Sistem, Perancangan sistem, Implementasi dengan pengujian serta evaluasi sistem.



Gambar 3. 1 Alur Proses Penelitian

### 3.1 Studi Literatur

Pada tahap studi literatur penulis akan melakukan eksplorasi dan meneliti teori-teori yang berkaitan dengan penelitian sebelumnya atau penelitian terdahulu yang sesuai dan berhubungan dengan penelitian yang sedang peneliti lakukan.

Untuk kegiatan yang dilakukan pada tahap literatur ini adalah mengumpulkan berbagai macam sumber tentang teori-teori pendukung mengenai aplikasi yang akan dibangun. Teori-teori tersebut dapat dicari melalui buku, jurnal, internet, dan juga penelitian yang terkait dengan topik penelitian ini.

### 3.2 Analisis Perancangan Sistem

Pada saat perancangan sistem pada penelitian ini ada beberapa hal yang perlu diperhatikan, dimana desain mempunyai 2 (dua) tahap yaitu *low fidelity* dan *high fidelity* yang akan dibuat menggunakan *Figma*, lalu pada *database* akan menggunakan fitur *Firestore Database*, *Realtime Database*, *Authentication* yang terdapat pada *Firebase* yang dimana akan menjadi penghubung data antara aplikasi dan *database* itu sendiri. Pada bagian registrasi, data input pada aplikasi mengirim data kepada *database* dan *database* akan memunculkan antrean pada aplikasi secara otomatis. Lalu pada fitur konsultasi dimana *database* sebagai penghubung *chat* dari pasien yang sedang melakukan konsultasi dan di munculkan pada kolom *chat* dengan dokter tersebut.

### 3.2.1 Variabel Pembuatan Perangkat Lunak

#### A. Metode Yang Digunakan

Merupakan alur dari antrean *FIFO (First In First Out)*, pasien pertama yang tiba atau mendaftar dalam sistem akan dilayani terlebih dahulu, diikuti oleh pasien berikutnya dalam urutan kedatangan dalam pendaftaran mereka,. *FIFO* juga disebut dengan *FCFS (First Come, First Served)*. Metode ini diambil sebagai metode penelitian dikarenakan dapat menghindari penumpukan pasien saat mengantri.

#### B. Kebutuhan Sistem

Untuk spesifikasi perangkat keras dan perangkat lunak yang akan digunakan oleh peneliti untuk melakukan penelitian ini adalah seperti pada tabel berikut.

Table 3 1 Spesifikasi Laptop Yang Digunakan

Perangkat	Spesifikasi
<i>Prosesor</i>	<i>AMD Ryzen 5 3500U</i>
<i>RAM</i>	<i>8 GB</i>
<i>VGA</i>	<i>Raden Vega 8 Graphics</i>
<i>SSD</i>	<i>256 GB</i>
Sistem Operasi	<i>Windows 10 64-bit</i>

#### C. Konfigurasi *Android Studio*

Pada *android studio* digunakan *coding* untuk pembuatan aplikasi itu sendiri dan *coding* aplikasi membutuhkan *tool-tool* yang mendukung

seperti *Java SDK* versi 11. Pada *Java SDK* seharusnya diinstal terlebih dahulu sebelum menginstal *android studio* agar sistem dari *android studio* bisa dengan otomatis mendeteksi bahwa komputer atau laptop yang akan digunakan dalam pembuatan aplikasi tersebut sudah terpasang *SDK* dari *java*, tetapi jika laptop atau komputer belum terpasang *SDK* maka sistem dari *android studio* memunculkan sebuah halaman instalasi untuk pengguna bahwa harus memasang *SDK* terlebih dahulu jika ingin menggunakan aplikasinya.

#### **D. Konfigurasi *Firestore***

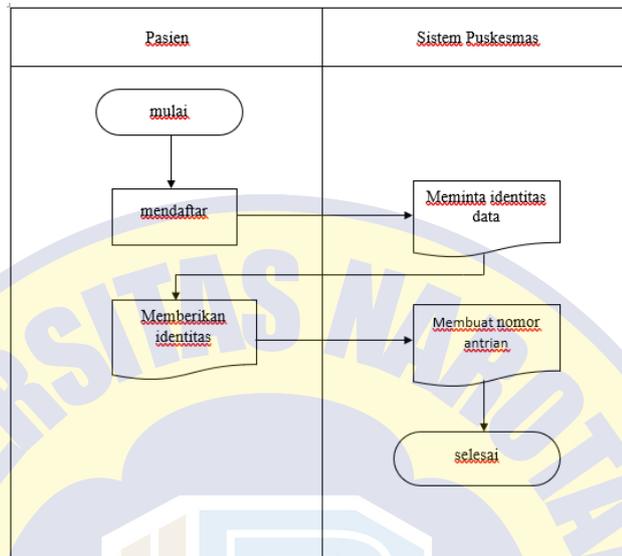
Pada *firebase* akan menggunakan fitur *Firestore database*, *realtime database*, dan *authentication* yang ada pada *firebase* yang dimana akan mengambil *token* untuk akses database aplikasi. Untuk struktur data pada *firebase* peneliti menggunakan *child* dengan *key* bernama “*state*” dan *value* yang bertipe data *auto* berasal dari data yang didapat dari memasukkan pada aplikasi.

Table 3.2 Config *Firestore*

<i>Key</i>	<i>Value</i>
<i>state</i>	<i>auto</i>

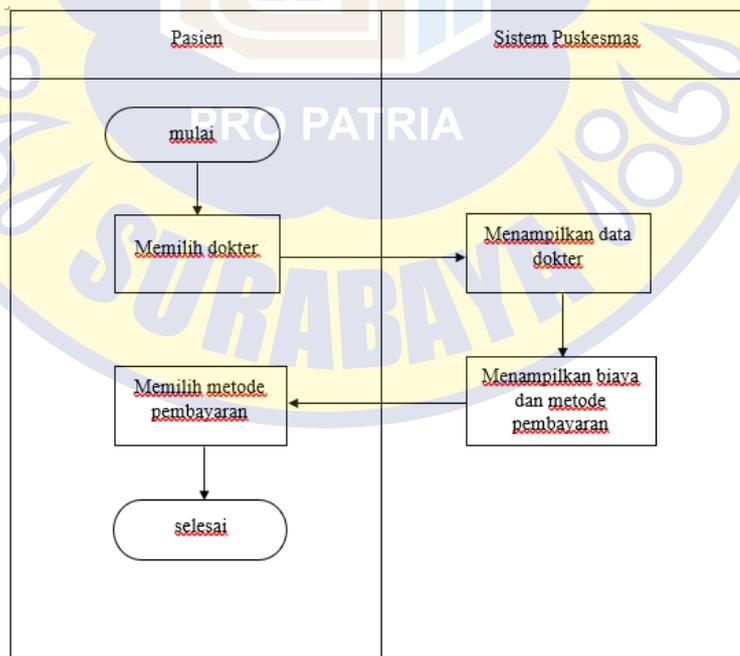
### 3.3 Diagram Alir Sistem

#### 3.3.1 Flowmap Registrasi Antrean



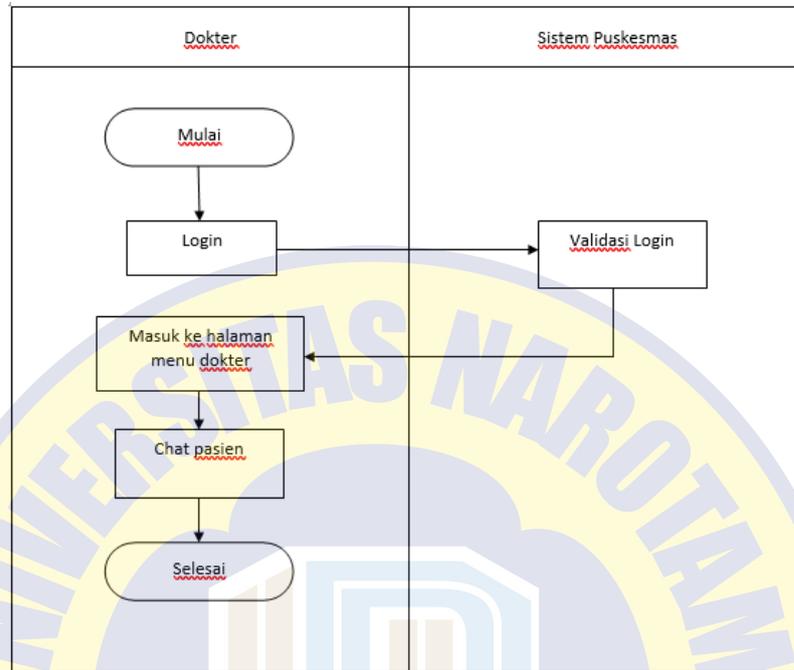
Gambar 3. 2 Flowmap registrasi Antrean

#### 3.2.2 Flowmap Konsultasi Dokter



Gambar 3. 3 Flowmap Konsultasi Dokter

### 3.2.3 Flowmap Dokter Konsultasi



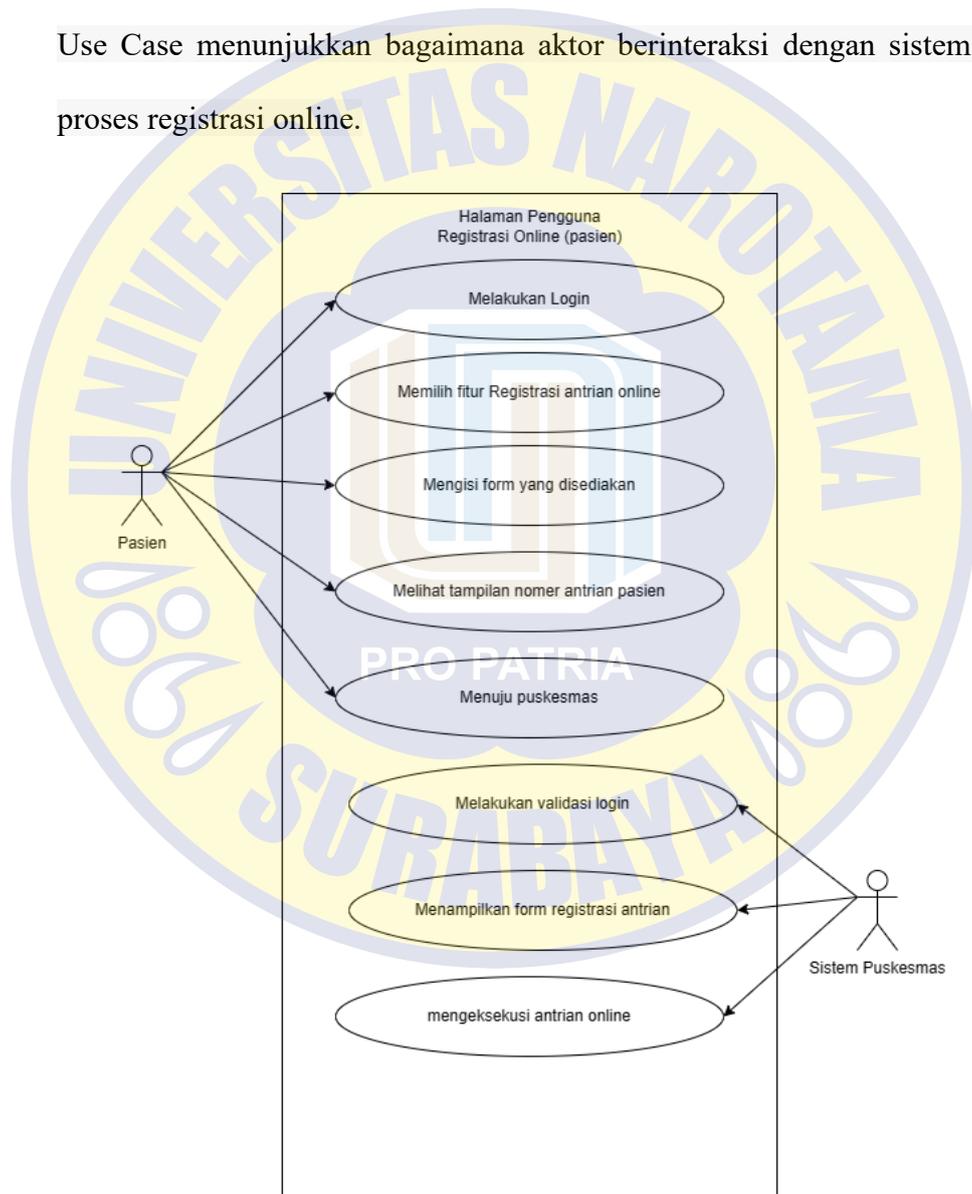
Gambar 3. 4 Flowmap Dokter Konsultasi

### 3.4 UML

Dalam pengembangan sistem perangkat lunak, UML atau Unified Modeling Language merupakan bahasa visual yang berguna untuk membuat desain dan model sistem perangkat lunak. Dengan menggunakan UML, pengembang dapat memvisualisasikan berbagai aspek dari sistem perangkat lunak, termasuk fitur, proses bisnis, arsitektur, dan interaksi antara komponen-komponen yang berbeda dalam sistem tersebut. UML juga membantu dalam menentukan hubungan antara komponen, sehingga dapat membantu pengembang dalam memahami dan mengembangkan sistem perangkat lunak dengan lebih efektif. Di seluruh dunia, UML digunakan sebagai alat standar untuk mendesain dan memodelkan sistem perangkat lunak yang kompleks oleh para pengembang perangkat lunak.

### 3.4.1 Use case Diagram (Registrasi Antrian)

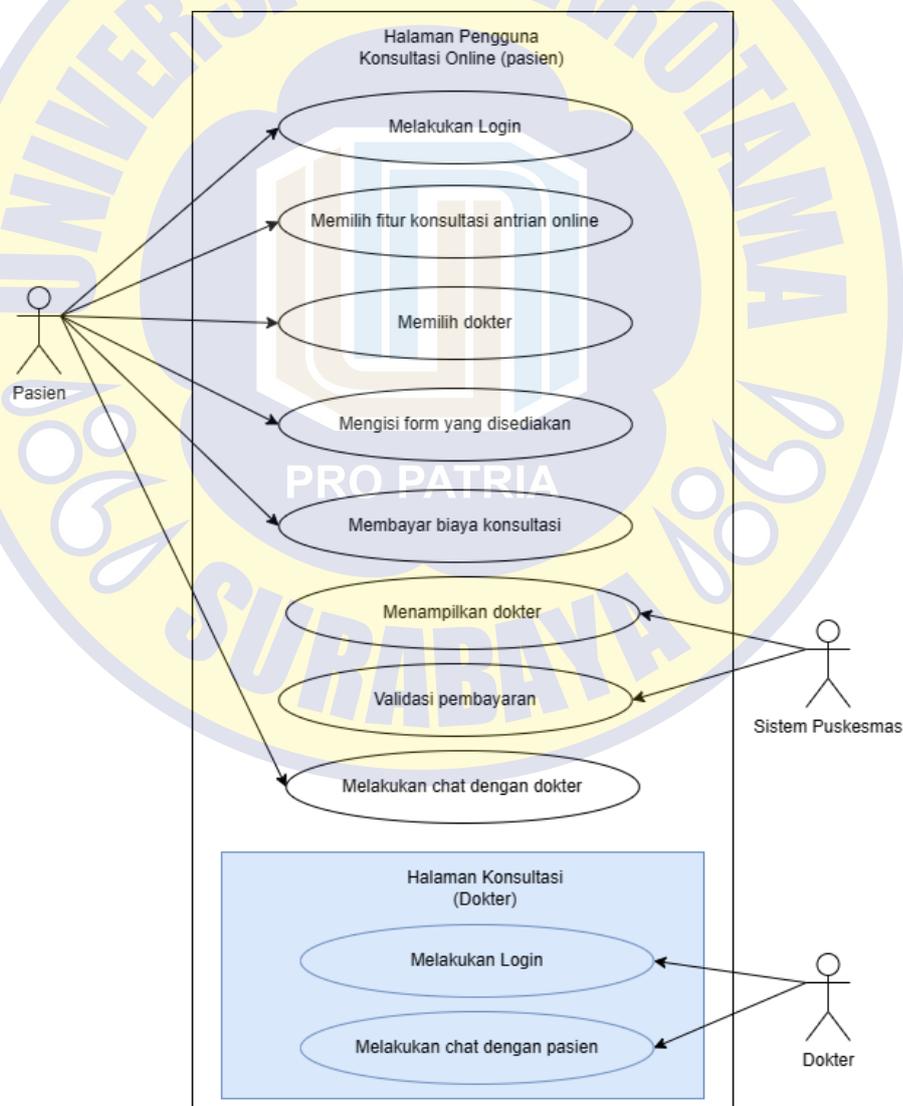
Gambaran proses informasi yang terjadi pada halaman registrasi online dapat dilihat melalui diagram Use Case dibawah yang menunjukkan interaksi antara sistem dan pengguna. Diagram ini dirancang untuk mempresentasikan keseluruhan sistem dengan mempertimbangkan analisis kebutuhan. Diagram Use Case menunjukkan bagaimana aktor berinteraksi dengan sistem dalam proses registrasi online.



Gambar 3. 5 Use case Diagram Registrasi Antrian

### 3.4.2 Use Case Diagram (Konsultasi Online)

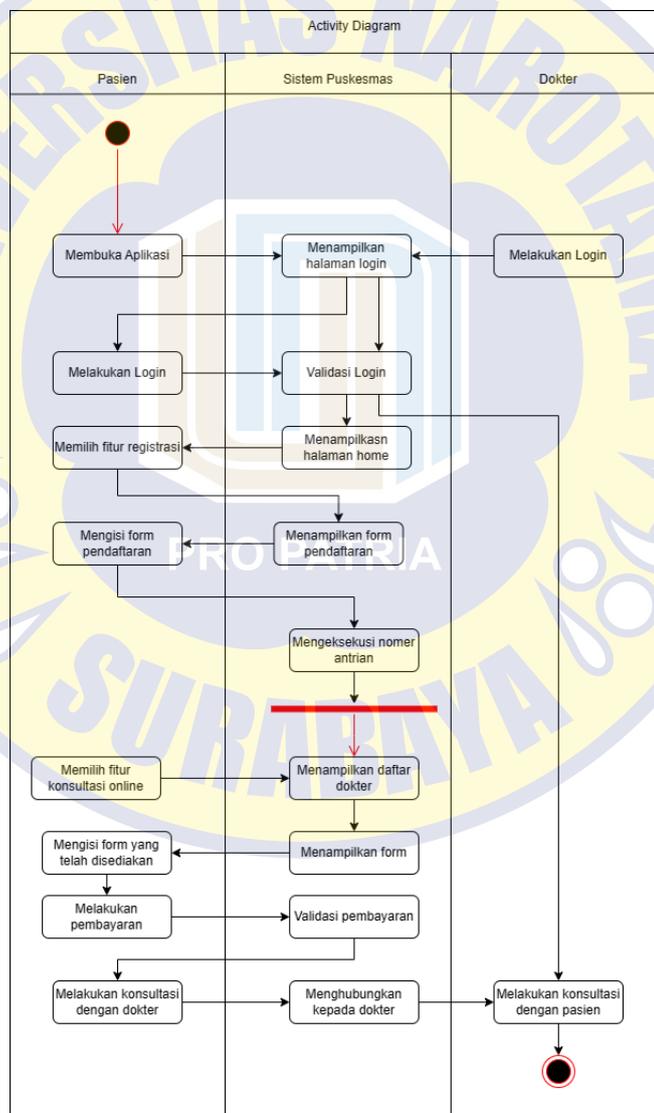
Diagram Use Case berikut ini adalah gambaran dari proses informasi yang berjalan di halaman konsultasi online yang dirancang. Sistem ini berinteraksi dengan aktor dan dokter untuk memenuhi kebutuhan yang telah dianalisis, sehingga menyajikan gambaran keseluruhan sistem. Kasus penggunaan ini berfokus pada interaksi antara aktor dan sistem, serta aktor dengan dokter dalam proses konsultasi.



Gambar 3. 6 Use Case Diagram Konsultasi Online

### 3.4.3 Activity Diagram

Sistem antrian dan konsultasi online menyediakan kenyamanan bagi pasien, dokter, dan puskesmas. Proses berulang dari awal hingga akhir bermula ketika pasien membuka aplikasi dan melakukan pendaftaran. Setelahnya, puskesmas akan memproses informasi yang diberikan oleh pasien dan meneruskan kepada dokter. Dokter akan melakukan konsultasi dan mengeluarkan diagnosis.



Gambar 3. 7 Activity Diagram

### **3.5 Implementasi**

Pada tahap ini untuk memastikan bahwa aplikasi yang dikembangkan layak, dilakukan uji kesesuaian dengan menjalankan program untuk mengidentifikasi error. Jika terjadi ketidaksesuaian, akan dilakukan peninjauan ulang, dan melakukan tahap selanjutnya hingga aplikasi berjalan tanpa error.

