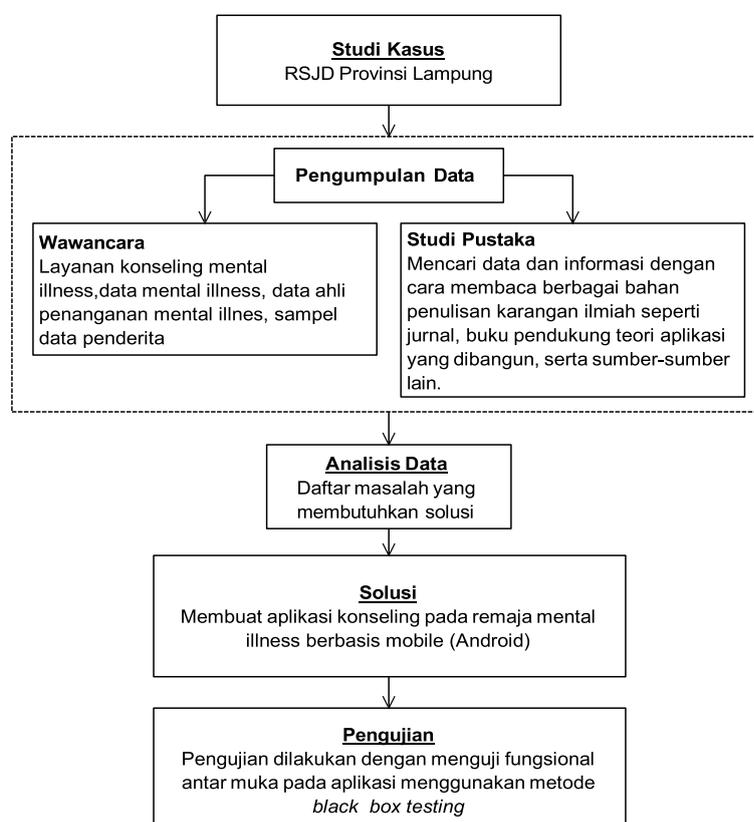


BAB III METODOLOGI PENELITIAN

3.1 Kerangka Penelitian

Penelitian dimulai dengan menentukan tempat penelitian yaitu di RSJD Provinsi Lampung. Setelah itu, langkah berikutnya adalah melakukan pengumpulan data berkaitan dengan konseling mental illness pada remaja. Setelah data didapat, tahap selanjutnya adalah menganalisa data dengan mencari kekurangan dari layanan konseling yang sedang berjalan saat ini dan kemudian ditetapkanlah suatu solusi. Setelah mendapatkan solusi, langkah selanjutnya adalah mengembangkan sistem dari solusi tersebut. Adapun kerangka dari penelitian yang dilakukan adalah seperti pada Gambar 3.1.



Gambar 3.1 Kerangka Kerja Penelitian

3.2 Kebutuhan Perangkat Lunak dan Perangkat Keras

Proses pembuatan aplikasi yang dibangun tidak terlepas dari *tools* pendukung dalam pembuatannya. Adapun *tools* pendukung yang dimaksud adalah perangkat lunak dan perangkat keras. Perangkat lunak dan perangkat keras yang digunakan adalah sebagai berikut:

a. Perangkat Lunak

Perangkat lunak yang digunakan dalam pembuatan aplikasi konseling pada remaja mental illness berbasis *mobile* Android ini adalah sebagai berikut :

1. Sistem operasi Windows 10 64 bit.
2. Aplikasi perancangan/pemodelan sistem : Edraw Max
3. Aplikasi *database* : MySQL Workbench
4. Aplikasi pembuatan program : Android Studio
5. Server : Node.js
6. JDK terbaru
7. Sistem Operasi Android minimal versi 8.0

b. Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan aplikasi konseling pada remaja mental illness berbasis *mobile* Android ini adalah laptop atau *Personal Computer* dengan spesifikasi sebagai berikut :

1. RAM 4 GB jika tidak menggunakan emulator Android (percobaan langsung menggunakan *smartphone* Android)
2. RAM 8 GB atau lebih jika menggunakan emulator Android untuk menjalankan atau *testing* pemograman
3. Hardisk minimal 160 GB jika hanya terpasang sistem operasi, DB *Browser* MySQL, dan Android Studio saja. Disarankan di atas 160 GB, semakin besar kapasitas hardisk semakin baik
4. 1280 x 800 minimum resolusi layar
5. Prosesor *intel core* atau amd atau processor setaranya
6. *Smartphone* Android

3.3 Metode Pengumpulan Data

Metode pengumpulan data dilakukan untuk mendapatkan data terkait dengan konseling mental illness. Adapun langkah yang diambil peneliti untuk mendapatkan data terkait adalah sebagai berikut :

a. Wawancara

Hasil wawancara berupa cara kerja konseling mental illness yang dilakukan pada saat ini.

b. Studi Pustaka

Studi pustaka dilakukan untuk memperoleh data dan informasi dengan cara membaca berbagai bahan penulisan karangan ilmiah seperti jurnal, buku pendukung teori aplikasi berbasis Android yang dibangun, serta sumber-sumber lain mengenai permasalahan yang berhubungan dengan penulisan.

3.4 Metode Pengembangan Sistem *Scrum*

Aplikasi konseling pada remaja mental illness berbasis *mobile* Android dibangun menggunakan model *scrum* berdasarkan penjelasan pada pokok bahasan dua sub 2.9. Pada pokok bahasan tersebut dijelaskan bahwa kerangka kerja *scrum* terdiri dari lima tahapan, yaitu *product backlog*, *sprint planning meeting*, *sprint backlog*, *sprint*, dan *finished work*. Adapun alur tahapan pengembangan sistem menggunakan metode *scrum* adalah seperti pada Gambar 3.2.



Gambar 3.2 Alur Tahapan Pengembangan Sistem Menggunakan Metode Scrum

3.4.1 Product Backlog

Tahap pembuatan *product backlog* merupakan pembagian atau pengelompokan proses-proses berdasarkan dari analisis pada langkah sebelumnya. Tahap *product backlog* ini memiliki isi fitur yang akan dibuat dan prioritas dari masing-masing fitur yang ditampilkan pada Tabel 3.1. *Product backlog* ini dibagi menjadi tiga kategori pengguna sistem, yaitu admin, dokter, dan pasien (penderita mental illness). Setiap fitur *backlog* memiliki prioritas, hal ini ditentukan berdasarkan tingkat kepentingan kebutuhan dari pengguna. Prioritas menentukan fitur mana yang harus dikerjakan terlebih dahulu.

Tabel 3.1 *Product Backlog*

No	Fitur	Prioritas
1.	<i>Login multi-user</i>	Prioritas Tinggi
2.	<i>Dashboard</i> pengguna sistem	Prioritas Sedang
Pengguna Level Admin		
1.	Mengelola data dokter	Prioritas Tinggi
2.	Mengelola data praktek	Prioritas Tinggi

3.	Mengelola data pasien	Prioritas Rendah
Pengguna Level Dokter (Ahli Mental Illness)		
1.	Mengelola data konsultasi	Prioritas Tinggi
Pengguna Level Pasien (Penderita Mental Illness)		
1.	Mengelola data konsultasi	Prioritas Tinggi

3.4.2 Sprint Planning Meeting

Pada tahapan ini dilakukan rapat perencanaan *sprint* yang dihadiri oleh *scrum master*, *product owner*, dan *development team*. *Scrum master* dan *product owner* merupakan dokter ahli RSJD penanganan mental illness pada remaja, sedangkan *development team* dalam penelitian ini adalah peneliti.

3.4.3 Sprint Backlog

Sprint backlog disini berisikan penjabaran fitur-fitur *backlog* menjadi tugas-tugas yang lebih detil. Hal ini difungsikan untuk mempermudah tim pengembang dalam menyelesaikan *sprint*. Adapun *sprint backlog* aplikasi yang akan dibangun seperti pada Tabel 3.2.

Tabel 3.2 *Sprint Blocking*

No	Fitur Backlog	Tugas (Task)
1.	<i>Login multi-user</i>	<i>Splash screen</i>
		Semua pengguna sistem masuk ke dalam sistem (<i>login</i>) menggunakan satu <i>form</i>
2.	<i>Dashboard</i> pengguna sistem	<i>Dashboard</i> (menu utama) admin
		<i>Dashboard</i> (menu utama) dokter
		<i>Dashboard</i> (menu utama) pasien
Pengguna sistem level Admin		
1.	Mengelola data dokter	Melihat data dokter
		Menghapus data dokter
		Menambah data dokter

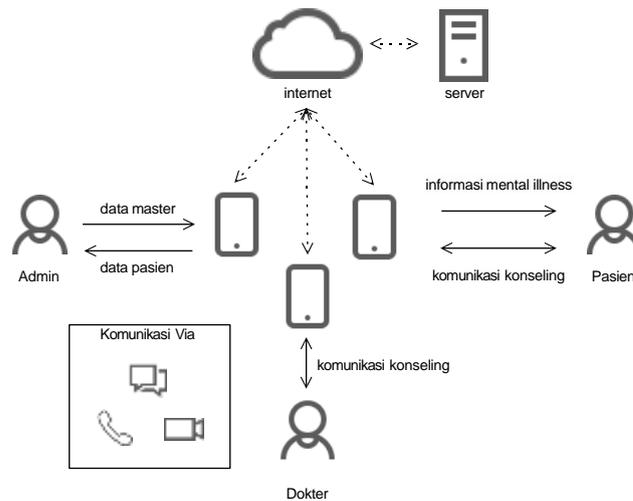
		Mengubah data dokter
2.	Mengelola data praktek	Menambah data praktek
		Mengubah data praktek
		Menghapus data praktek
		Melihat data praktek
3.	Mengelola data pasien	Melihat data pasien
Pengguna sistem level Dokter		
1.	Mengelola data konsultasi	Melakukan layanan konsultasi via telepon, pesan, dan atau video call
		Melihat data pasien
		Melihat data praktek
Pengguna sistem level Pasien		
1.	Mengelola data konsultasi	Melakukan layanan konsultasi via telepon, pesan, dan atau video call
		Melihat data praktek

3.4.4 *Sprint*

Pada tahapan ini development membuat rancangan prototipe sistem atau aplikasi yang terdiri dari arsitektur sistem, alur tahapan pengembangan sistem, *use case diagram*, *activity diagram*, *class diagram*, dan *interface* aplikasi.

3.4.4.1 Arsitektur Sistem

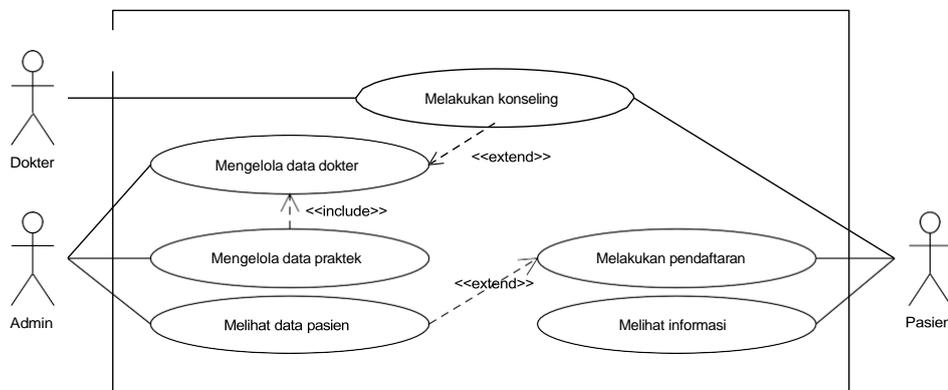
Adapun perancangan arsitektur sistem yang diusulkan adalah seperti yang terlihat pada Gambar 3.3. Aplikasi yang dibangun terdiri dari tiga hak akses yaitu akses admin, dokter, dan pasien. Admin mengelola data dokter dan data praktek yang kemudian terkoneksi dan tersimpan di dalam *database* melalui media *smartphone* Android yang terkoneksi dengan internet. Admin disini adalah orang yang ditugaskan untuk mengelola data master, dokter merupakan ahli penanganan mental illness, sedangkan pasien merupakan penderita mental illness.



Gambar 3.3 Perancangan Arsitektur Sistem Diusulkan

3.4.4.2 Use Case Diagram

Perancangan *use case diagram* pada aplikasi yang dibangun adalah seperti pada Gambar 3.4. Aplikasi yang dibangun memiliki 3 aktor, yaitu admin, dokter, dan pasien. Admin dapat mengelola data praktek, mengelola data dokter, dan melihat data pasien. Dokter dapat melakukan konseling. Sedangkan pasien dapat melakukan pendaftaran dan konseling.



Gambar 3.4 Perancangan Use Case Diagram

a. Definisi Aktor

Deskripsi pendefinisian aktor konseling pada aplikasi konseling remaja mental illness adalah seperti pada Tabel 3.3.

Tabel 3.3 Pendefinisian Aktor

No	Aktor	Deskripsi
1.	Admin	Admin disini bertugas dan bertanggung jawab atas pengelolaan data master (data dokter dan data praktek)
2.	Dokter	Orang merupakan dokter ahli penanganan mental illness yang dapat berkomunikasi dengan pasien dalam melakukan konseling.
3.	Pasien	Pasien merupakan orang yang menderita penyakit mental illness dan dapat melakukan konseling dengan dokter

b. Definisi *Use Case*

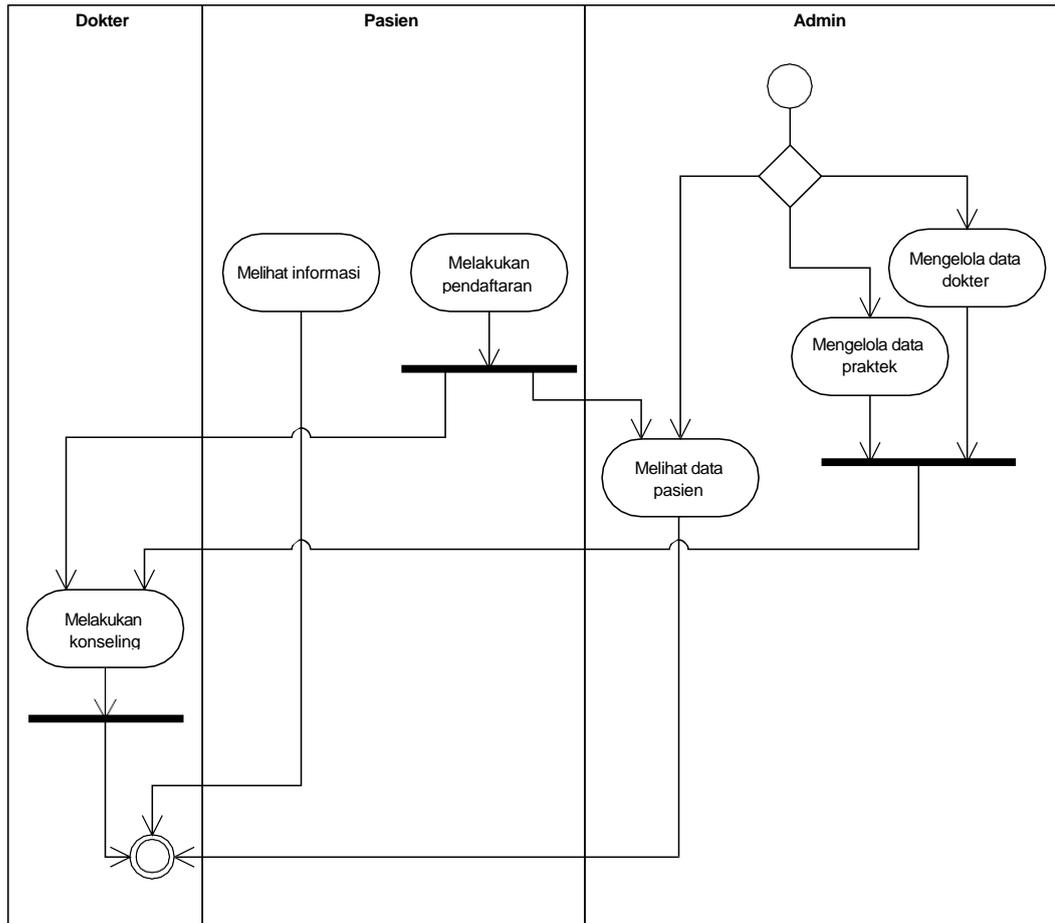
Deskripsi pendefinisian *use case* pada aplikasi konseling remaja mental illness adalah seperti pada Tabel 3.4.

Tabel 3.4 Pendefinisian *Use Case*

No	<i>Use Case</i>	Deskripsi
1.	Mengelola data dokter	Merupakan proses dimana admin mengelola data dokter (CRUD)
2.	Mengelola data praktek	Merupakan proses dimana admin mengelola data praktek (CRUD)
3.	Melihat data pasien	Merupakan proses dimana admin hanya dapat melihat data pasien
4.	Melakukan pendaftaran	Merupakan proses dimana pasien melakukan pendaftaran pasien
5.	Melakukan konseling	Merupakan proses dimana dokter dan pasien melakukankomunikasi konseling
6.	Melihat informasi	Merupakan proses dimana pasien hanya dapat melihat informasi

3.4.4.3 Activity Diagram

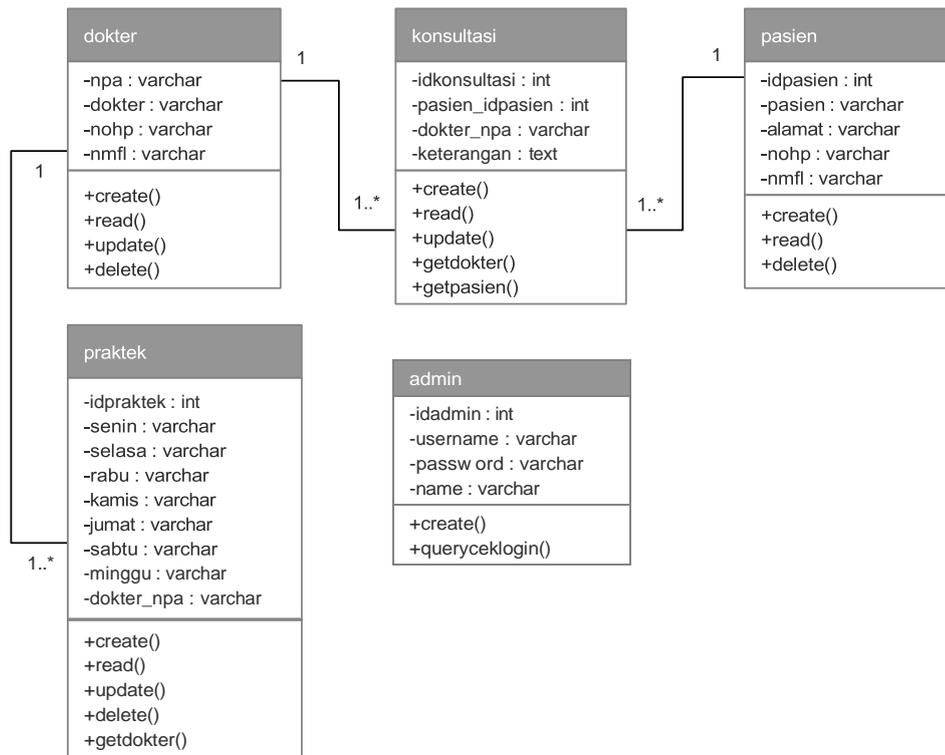
Adapun perancangan *activity diagram* aplikasi konseling remaja mental illness adalah seperti pada Gambar 3.5. Diagram tersebut menggambarkan aktifitas kegiatan aktor.



Gambar 3.5 Perancangan Activity Diagram

3.4.4.4 Class Diagram

Adapun perancangan *class diagram* sistem diusulkan adalah seperti pada Gambar 3.6. Terdapat 5 *class* yang dibuat, yaitu *class* dokter, *class* pasien, *class* praktek, *class* konsultasi, dan *class* admin.



Gambar 3.6 Perancangan *Class Diagram* Sistem Diusulkan

Terdapat 5 tabel yang ada pada *database* konselingmi, yaitu tabel admin, tabel dokter, tabel praktek, tabel pasien, dan tabel konsultasi. Kamus data dari tabel-tabel tersebut adalah sebagai berikut :

a. Kamus Data Tabel Admin

Tabel ini digunakan untuk menyimpan data admin. Adapun perancangan kamus data tabel admin adalah seperti pada Tabel 3.5.

Nama tabel : Admin

Nama *database* : Konselingmi

Primary key : Idadmin

Foreign key : -

Tabel 3.5 Kamus Data Tabel Admin

<i>field_name</i>	<i>type</i>	<i>length</i>	Keterangan
idadmin	int	10	id admin
username	varchar	20	<i>Username</i>
password	varchar	20	<i>Password</i>
name	varchar	50	Nama

b. Kamus Data Tabel Dokter

Tabel ini digunakan untuk menyimpan data dokter. Adapun perancangan kamus data tabel dokter adalah seperti pada Tabel 3.6.

Nama tabel : dokter

Nama *database* : konselingmi

Primary key : npa

Foreign key : -

Tabel 3.6 Kamus Data Tabel Dokter

<i>field_name</i>	<i>type</i>	<i>length</i>	Keterangan
npa	varchar	18	nomor pokok anggota
dokter	varchar	50	nama dokter
nohp	varchar	16	nomor <i>handphone</i>
nmfl	varchar	50	nama <i>file</i>

c. Kamus Data Tabel Praktek

Tabel ini digunakan untuk menyimpan data praktek. Adapun perancangan kamus data tabel praktek adalah seperti pada Tabel 3.7.

Nama tabel : praktek

Nama *database* : konselingmi

Primary key : idpraktek

Foreign key : dokter_npa

Tabel 3.7 Kamus Data Tabel Praktek

<i>field_name</i>	<i>type</i>	<i>length</i>	Keterangan
idpraktek	int	default	id praktek
senin	varchar	13	Senin
selasa	varchar	13	Selasa
rabu	varchar	13	Rabu
kamis	varchar	13	Kamis
jumat	varchar	13	Jumat
sabtu	varchar	13	Sabtu
minggu	varchar	13	Minggu
dokter_npa	varchar	18	npa dokter

d. Kamus Data Tabel Pasien

Tabel ini digunakan untuk menyimpan data pasien. Adapun perancangan kamus data tabel pasien adalah seperti pada Tabel 3.8.

Nama tabel : pasien

Nama *database* : konselingmi

Primary key : idpasien

Foreign key : -

Tabel 3.8 Kamus Data Tabel Pasien

<i>field_name</i>	<i>type</i>	<i>length</i>	Keterangan
Idpasien	int	10	id pasien
Pasien	varchar	50	nama pasien
Alamat	text	default	Alamat
Nohp	varchar	16	nomor <i>handphone</i>
nmfile	varchar	50	nama <i>file</i>

e. Kamus Data Tabel Konsultasi

Tabel ini digunakan untuk menyimpan data konseling/konsultasi. Adapun perancangan kamus data tabel konsultasi adalah seperti pada Tabel 3.9.

Nama tabel : konsultasi
 Nama *database* : konselingmi
Primary key : idkonsultasi
Foreign key : dokter_npa, pasien_idpasien

Tabel 3.9 Kamus Data Tabel Admin

<i>field_name</i>	<i>type</i>	<i>length</i>	Keterangan
idkonsultasi	int	10	id konsultasi
keterangan	text	default	Keterangan
pasien_idpasien	int	10	id pasien
dokter_npa	varchar	18	npa dokter

3.4.4.5 Interface Aplikasi

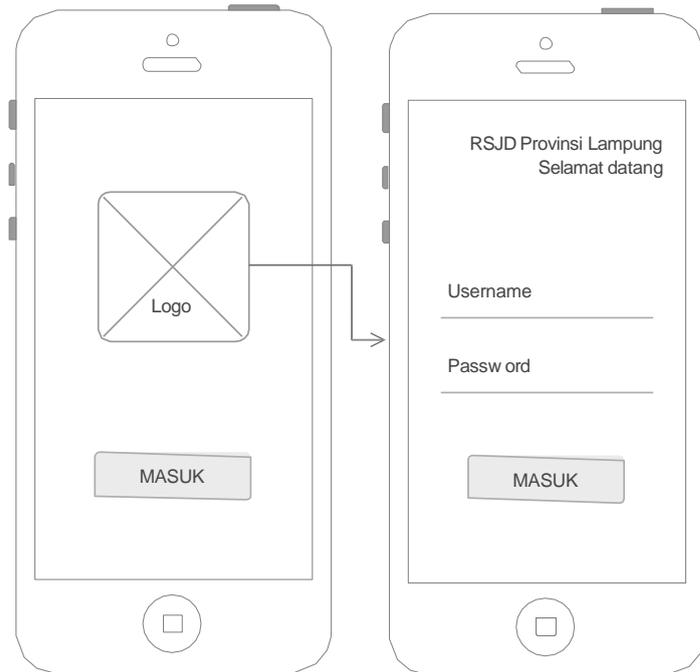
Interface aplikasi dirancang terdiri dari tiga akses, yaitu akses admin, dokter dan akses pasien. Adapun perancangan *interface* aplikasi dengan tiga akses tersebut adalah dijelaskan pada sub pokok bahasan di bawah ini.

3.4.4.5.1 Interface Aplikasi Akses Admin

Interface aplikasi akses admin terdiri dari beberapa menu, yaitu menu *login*, menu utama (*dashboard*), menu dokter, menu praktek, dan menu pasien. Adapun perancangan *interface* aplikasi pada menu-menu tersebut adalah sebagai berikut :

a. Interface Menu Login

Menu ini dirancang sebagai sarana *login*. Adapun perancangan *interface* menu *login* akses admin adalah seperti pada Gambar 3.7.



Gambar 3.7 Perancangan *Interface* Menu *Login* Akses Admin

b. *Interface* Menu Utama (*Dashboard*)

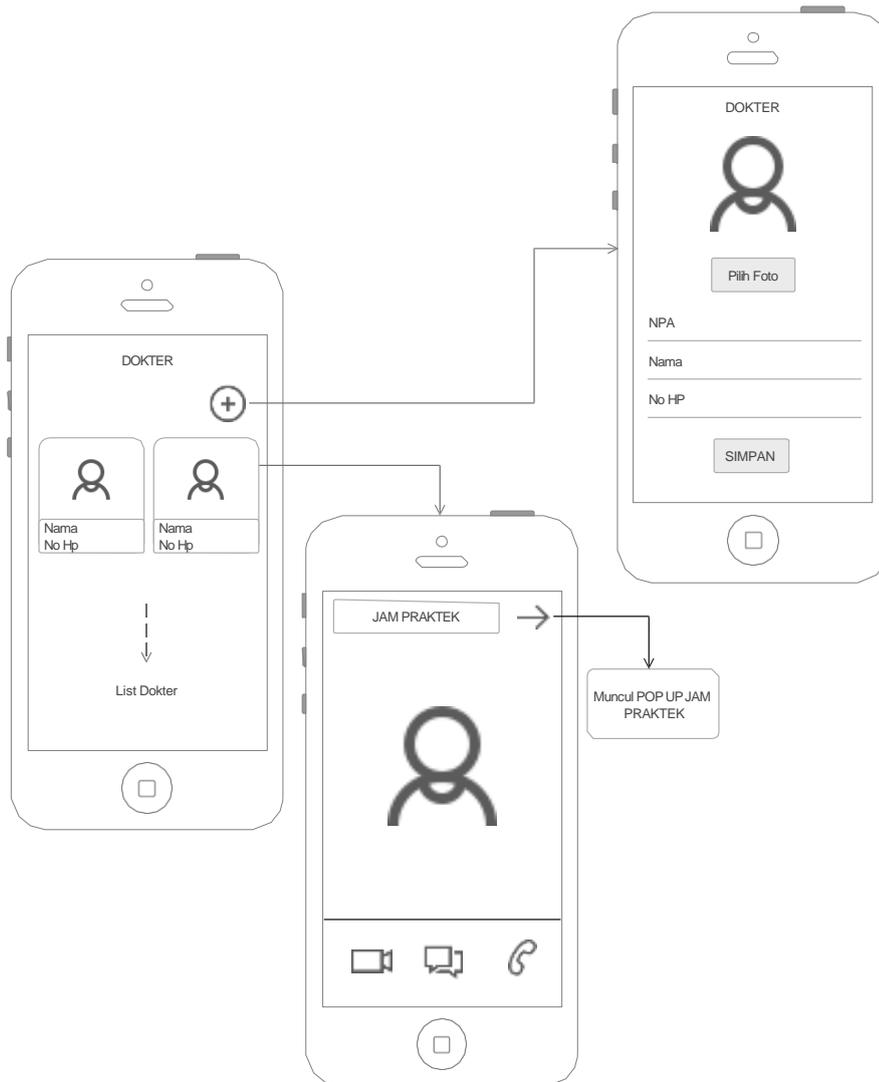
Setelah berhasil *login*, maka sistem akan menampilkan menu utama (*dashboard*) yang berisikan beberapa menu, yaitu menu dokter, praktek, dan pasien. Adapun perancangan *interface* menu utama (*dashboard*) akses admin adalah seperti pada Gambar 3.8.



Gambar 3.8 Perancangan *Interface* Menu Utama (*Dashboard*) Akses Admin

c. *Interface Menu Dokter*

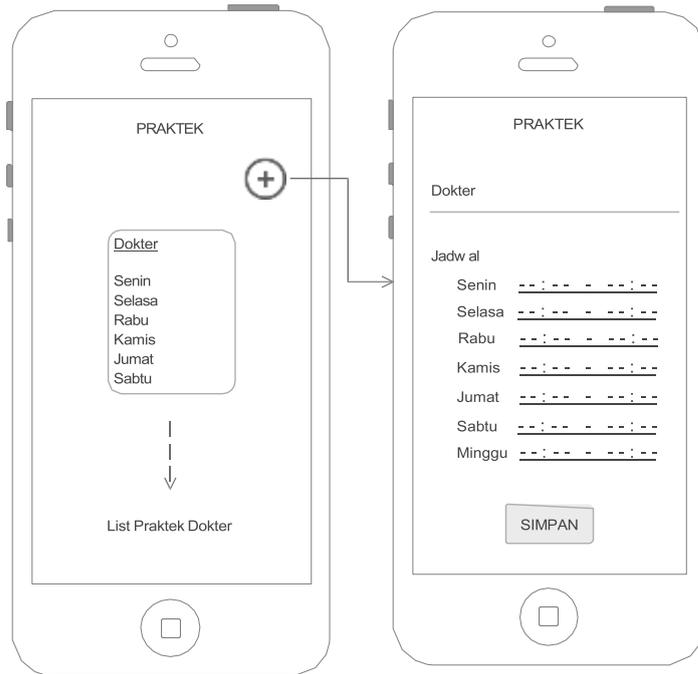
Menu ini dirancang untuk menampilkan informasi data dokter. Pada menu ini, admin dapat mengelola data dokter Adapun perancangan *interface* menu dokter akses admin adalah seperti pada Gambar 3.9.



Gambar 3.9 Perancangan *Interface* Menu Dokter Akses Admin

d. *Interface Menu Praktek*

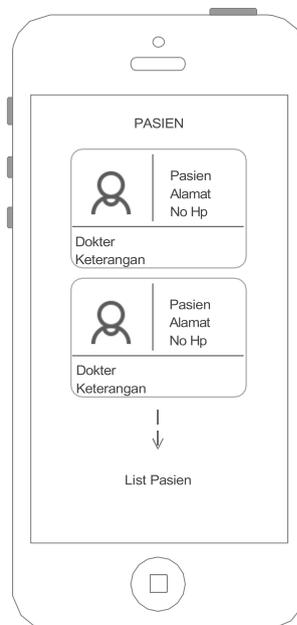
Menu ini dirancang untuk dapat mengelola data praktek dokter. Adapun perancangan *interface* menu praktek akses admin adalah seperti pada Gambar 3.10.



Gambar 3.10 Perancangan *Interface* Menu Praktek Akses Admin

e. *Interface* Menu Pasien

Menu ini dirancang untuk menampilkan informasi data pasien. Adapun perancangan *interface* menu pasien akses admin adalah seperti pada Gambar 3.11.



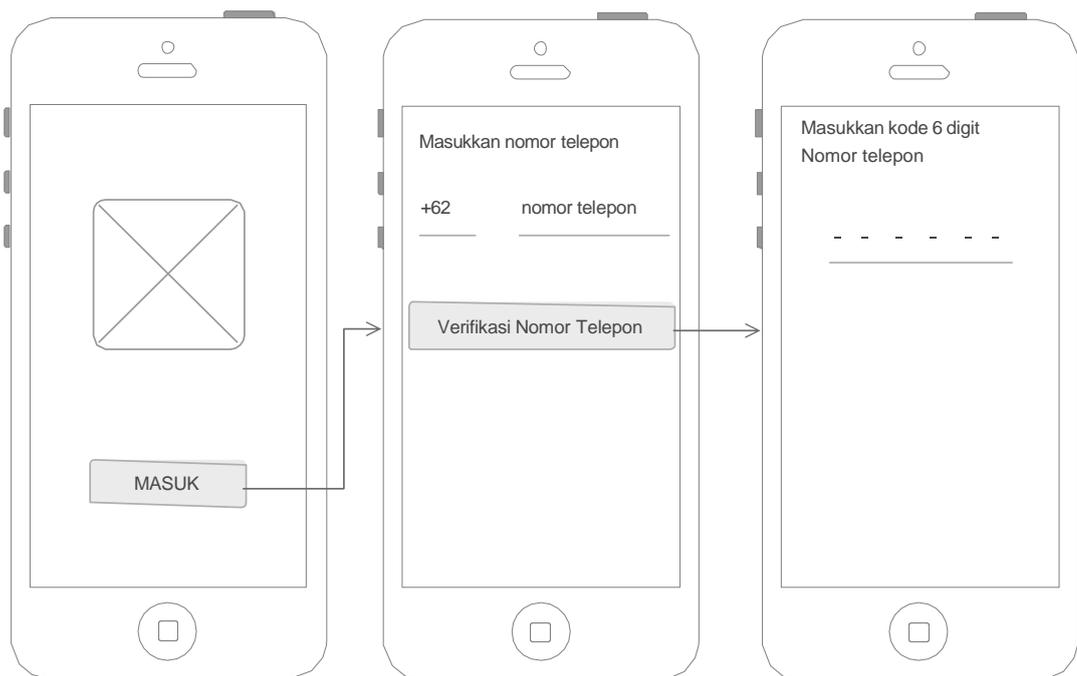
Gambar 3.11 Perancangan *Interface* Menu Pasien Akses Admin

3.4.4.5.2 Interface Aplikasi Akses Dokter

Interface aplikasi akses dokter terdiri dari beberapa menu, yaitu menu *login* dan menu utama (*dashboard*) konsultasi. Adapun perancangan *interface* aplikasi pada menu tersebut adalah sebagai berikut :

a. Interface Menu Login

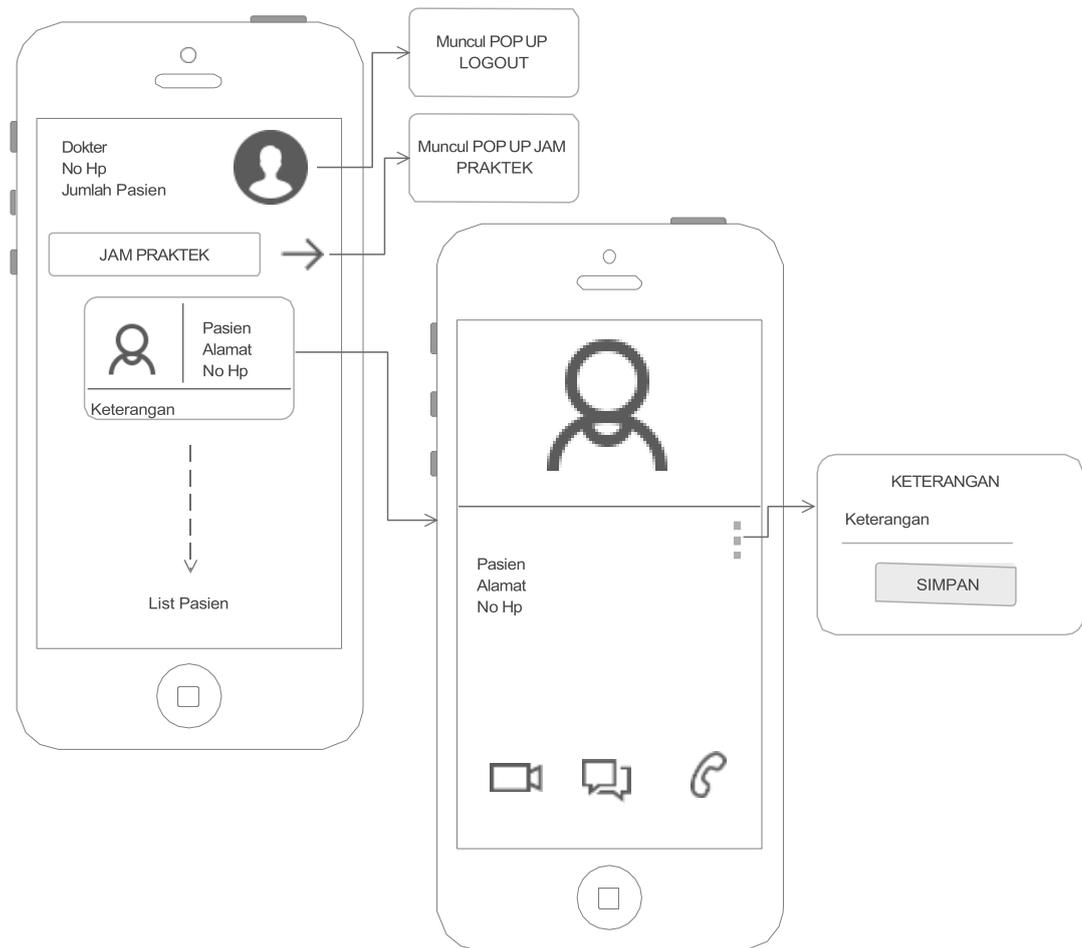
Menu ini dirancang ketika dokter hendak masuk aplikasi dengan menekan tombol “masuk”. Dokter login masuk menggunakan nomor *handphone* yang telah terdaftar pada data master yang kemudian melakukan kode verifikasi seperti pada Gambar 3.12.



Gambar 3.12 Perancangan *Interface* Menu *Login* Akses Dokter

b. Interface Menu Utama (*Dashboard*)

Jika telah berhasil *login*, sistem akan menampilkan menu utama (*dashboard*). Adapun perancangan *interface* menu utama aplikasi akses dokter adalah seperti pada Gambar 3.13.



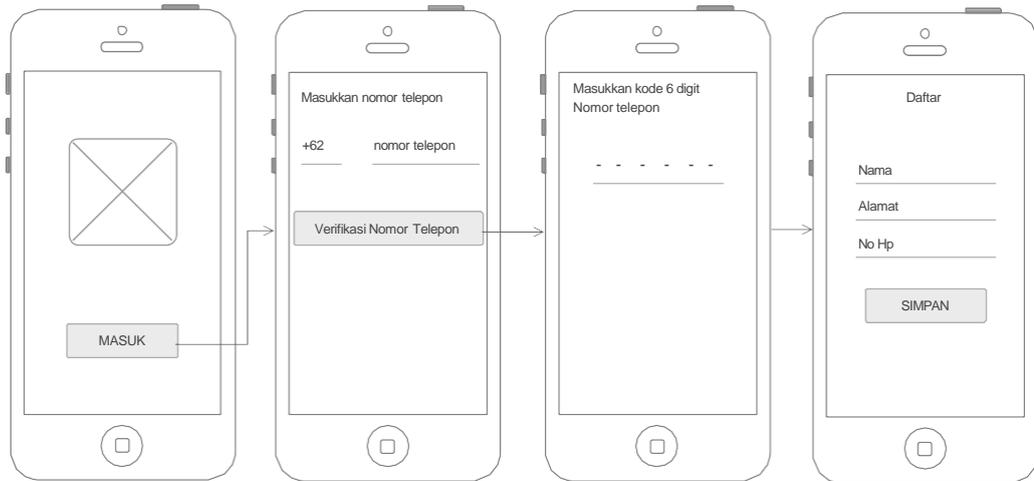
Gambar 3.13 Perancangan *Interface* Menu Utama (*Dashboard*) Akses Dokter

3.4.4.5.3 *Interface* Aplikasi Akses Pasien

Interface aplikasi akses pasien terdiri dari beberapa menu, yaitu menu *login* dan menu utama (*dashboard*) konsultasi. Adapun perancangan *interface* aplikasi pada menu tersebut adalah sebagai berikut :

a. *Interface* Menu Pendaftaran

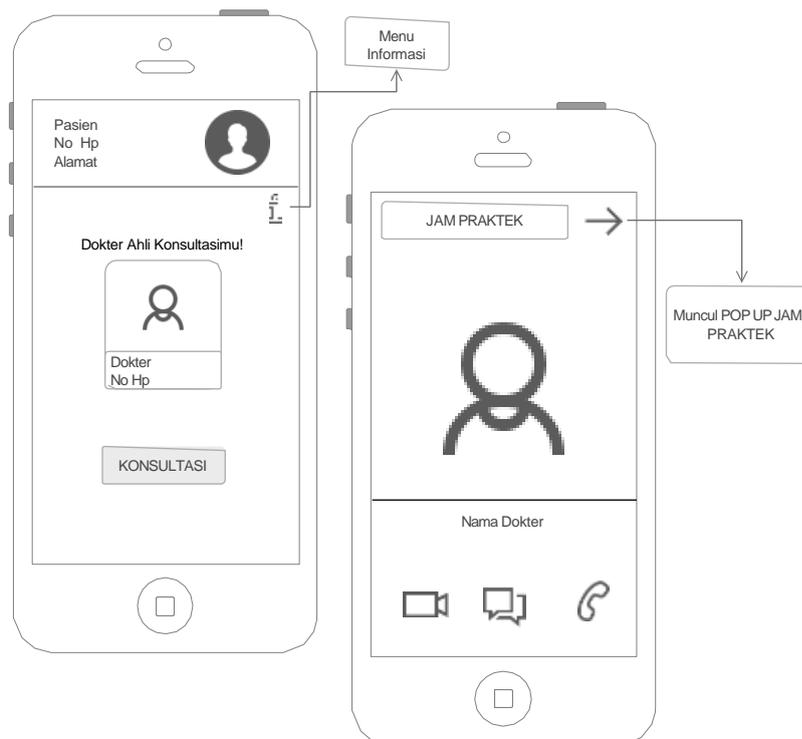
Calon pasien dapat melakukan pendaftaran terlebih dahulu pada menu ini dengan memasukkan nomor *handphone* dan mengisi kode validasi. Setelah itu, pengguna mengisi data diri seperti nama, alamat dan nomor *handphone*. Adapun perancangan *interface* menu pendaftaran pasien adalah seperti pada Gambar 3.14.



Gambar 3.14 Perancangan *Interface* Menu Pendaftaran Akses Pasien

b. *Interface* Menu Utama (*Dashboard*)

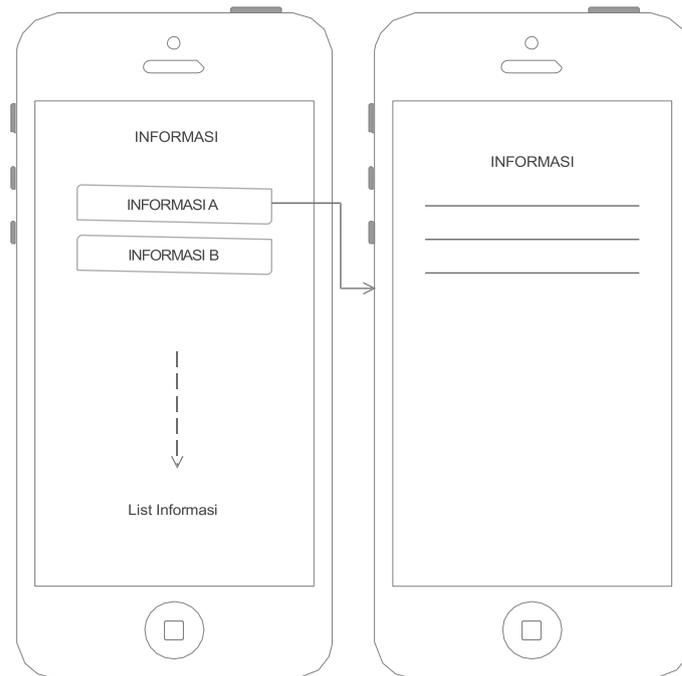
Jika telah berhasil melakukan pendaftaran, sistem akan otomatis memilihkan dokter pada ketika menekan tombol “konsultasi” pada menu utama (*dashboard*) . Adapun perancangan *interface* menu utama aplikasi akses pasien adalah seperti pada Gambar 3.15.



Gambar 3.15 Perancangan *Interface* Menu Utama (*Dashboard*) Akses Pasien

c. *Interface* Menu Informasi

Menu ini berisikan informasi mengenai mental illness. Adapun perancangan *interface* menu informasi akses pasien adalah seperti pada Gambar 3.16.



Gambar 3.16 Perancangan *Interface* Menu Informasi Akses Pasien

3.4.5 Finished Work

Tahap ini melakukan implementasi dari *sprint* yang telah dirancang sebelumnya yang diimbangi dengan pengujian aplikasi sebelum diserahkan kepada *scrum master*. Pengujian aplikasi dilakukan dengan menggunakan metode pengujian *blackbox* yang mengacu pada fungsional sistem. Adapun hasil dari implementasi dan pengujian dijelaskan pada Bab IV.

3.4.6 Perancangan *Black Box Testing*

Perancangan testing mengacu pada *black box* fungsional aplikasi dan pemasangan aplikasi pada versi Android yang berbeda. Adapun perancangan dari *black box testing* yang mengacu pada pengujian pemasangan aplikasi di tiap versi Android yang berbeda adalah seperti pada Tabel 3.10.

Tabel 3.10 Perancangan *Black Box Testing* Pemasangan Aplikasi

No.	Versi Android	Hasil Diharapkan
1.	7.0 ke bawah	Aplikasi tidak dapat terpasang
2.	8.0 – 11	Aplikasi dapat terpasang

