

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

MyDoc

untuk:

Layanan Kesehatan

Dipersiapkan oleh:

Akmal Muhamad Firdaus – 1301204188

Dilia Fadilah Mutmainah – 1301204383

Adinda Arwa Salsabil – 1301204011

M. Naufal Zaidan – 1301204338

IF-44-01

Program Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

| Program Studi S1 Teknik Informatika  -  Fakultas Informatika | Nomor Dokumen | | Halaman |
| --- | --- | --- | --- |
| *SKPL-MyDoc* | | 36 |
| Revisi | *0* | 05 - 01 - 2022 |

# Daftar Perubahan

| Revisi | Deskripsi |
| --- | --- |
| A |  |
| B |  |
| C |  |
| D |  |
| E |  |
| F |  |
| G |  |

| INDEX | - | A | B | C | D | E | F | G |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TGL |  |  |  |  |  |  |  |  |
| Ditulis oleh |  |  |  |  |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

# Daftar Halaman Perubahan

| Halaman | Revisi | Halaman | Revisi |
| --- | --- | --- | --- |
|  |  |  |  |

# Daftar Isi

[**Daftar Perubahan**](#_heading=h.1fob9te) **2**

[**Daftar Halaman Perubahan**](#_heading=h.3znysh7) **3**

[**Daftar Isi**](#_heading=h.2et92p0) **4**

[**1.**](#_heading=h.tyjcwt) **Pendahuluan 8**

[1.1](#_heading=h.3dy6vkm) Tujuan Penulisan Dokumen 8

[1.2](#_heading=h.1t3h5sf) Cakupan Produk 8

[1.3](#_heading=h.4d34og8) Konvensi Dokumen 8

[1.4](#_heading=h.2s8eyo1) Referensi 8

[1.5](#_heading=h.17dp8vu) Gambaran Umum 9

[**2.**](#_heading=h.3rdcrjn) **Overall Description 9**

[2.1](#_heading=h.26in1rg) Perspektif Produk 9

[2.2](#_heading=h.lnxbz9) Fungsi Produk 9

[2.3](#_heading=h.35nkun2) Kelas dan Karakteristik Pengguna 10

[2.4](#_heading=h.1ksv4uv) Lingkungan Operasi 10

[2.5](#_heading=h.44sinio) Batasan Perancangan dan Implementasi 10

[2.6](#_heading=h.2jxsxqh) Dokumentasi Pengguna 11

[2.7](#_heading=h.z337ya) Asumsi dan Dependensi 12

[**3.**](#_heading=h.3j2qqm3) **Requirements Antarmuka Eksternal 13**

[3.1](#_heading=h.1y810tw) Antarmuka Pengguna 13

[3.2](#_heading=h.4i7ojhp) Antarmuka Perangkat Keras 13

[3.3](#_heading=h.2xcytpi) Antarmuka Perangkat Lunak 13

[3.4](#_heading=h.1ci93xb) Antarmuka Komunikasi 13

**4**[**.**](#_heading=h.23ckvvd) **Fitur Sistem (Use Cases) 14**

4[.1](#_heading=h.ihv636) Use Case 1 15

4[.1.1](#_heading=h.32hioqz) Nama Use Case 15

4[.1.2](#_heading=h.1hmsyys) Tujuan 15

4[.1.3](#_heading=h.41mghml) Input 15

4[.1.4](#_heading=h.2grqrue) Output 15

4[.1.5](#_heading=h.vx1227) Skenario Utama 16

4[.1.6](#_heading=h.3fwokq0) Prakondisi 16

4[.1.7](#_heading=h.1v1yuxt) Langkah-langkah 16

4[.1.8](#_heading=h.4f1mdlm) Pascakondisi16

4[.2](#_heading=h.19c6y18) Use Case 17

4[.2.1](#_heading=h.32hioqz) Nama Use Case 17

4[.2.2](#_heading=h.1hmsyys) Tujuan 17

4[.2.3](#_heading=h.41mghml) Input 17

4[.2.4](#_heading=h.2grqrue) Output 17

4[.2.5](#_heading=h.vx1227) Skenario Utama 17

4[.2.6](#_heading=h.3fwokq0) Prakondisi 18

4[.2.7](#_heading=h.1v1yuxt) Langkah-langkah 18

4[.2.8](#_heading=h.4f1mdlm) Pascakondisi 18

4[.3](#_heading=h.ihv636) Use Case 3 19

4[.3.1](#_heading=h.32hioqz) Nama Use Case 19

4[.3.2](#_heading=h.1hmsyys) Tujuan 19

4[.3.3](#_heading=h.41mghml) Input 19

4[.3.4](#_heading=h.2grqrue) Output 19

4[.3.5](#_heading=h.vx1227) Skenario Utama 19

4[.3.6](#_heading=h.3fwokq0) Prakondisi 20

4[.3.7](#_heading=h.1v1yuxt) Langkah-langkah 20

4[.3.8](#_heading=h.4f1mdlm) Pascakondisi20

4[.4](#_heading=h.19c6y18) Use Case 4 21

4[.4.1](#_heading=h.32hioqz) Nama Use Case 21

4[.4.2](#_heading=h.1hmsyys) Tujuan 21

4[.4.3](#_heading=h.41mghml) Input 21

4[.4.4](#_heading=h.2grqrue) Output 21

4[.4.5](#_heading=h.vx1227) Skenario Utama 22

4[.4.6](#_heading=h.3fwokq0) Prakondisi 22

4[.4.7](#_heading=h.1v1yuxt) Langkah-langkah 22

4[.4.8](#_heading=h.4f1mdlm) Pascakondisi 22

4[.5](#_heading=h.ihv636) Use Case 5 23

4[.5.1](#_heading=h.32hioqz) Nama Use Case 23

4[.5.2](#_heading=h.1hmsyys) Tujuan 23

4[.5.3](#_heading=h.41mghml) Input 23

4[.5.4](#_heading=h.2grqrue) Output 23

4[.5.5](#_heading=h.vx1227) Skenario Utama 24

4[.5.6](#_heading=h.3fwokq0) Prakondisi 24

4[.5.7](#_heading=h.1v1yuxt) Langkah-langkah 24

4[.5.8](#_heading=h.4f1mdlm) Pascakondisi24

4[.6](#_heading=h.19c6y18) Use Case 6 25

4[.6.1](#_heading=h.32hioqz) Nama Use Case 25

4[.6.2](#_heading=h.1hmsyys) Tujuan 25

4[.6.3](#_heading=h.41mghml) Input 25

4[.6.4](#_heading=h.2grqrue) Output 25

4[.6.5](#_heading=h.vx1227) Skenario Utama 26

4[.6.6](#_heading=h.3fwokq0) Prakondisi 26

4[.6.7](#_heading=h.1v1yuxt) Langkah-langkah 26

4[.6.8](#_heading=h.4f1mdlm) Pascakondisi 27

4[.7](#_heading=h.ihv636) Use Case 7 27

4[.7.1](#_heading=h.32hioqz) Nama Use Case 27

4[.7.2](#_heading=h.1hmsyys) Tujuan 27

4[.7.3](#_heading=h.41mghml) Input 28

4[.7.4](#_heading=h.2grqrue) Output 28

4[.7.5](#_heading=h.vx1227) Skenario Utama 28

4[.7.6](#_heading=h.3fwokq0) Prakondisi 28

4[.7.7](#_heading=h.1v1yuxt) Langkah-langkah 28

4[.7.8](#_heading=h.4f1mdlm) Pascakondisi29

4[.8](#_heading=h.19c6y18) Use Case 8 29

4[.8.1](#_heading=h.32hioqz) Nama Use Case 29

4[.8.2](#_heading=h.1hmsyys) Tujuan 29

4[.8.3](#_heading=h.41mghml) Input 29

4[.8.4](#_heading=h.2grqrue) Output 29

4[.8.5](#_heading=h.vx1227) Skenario Utama 30

4[.8.6](#_heading=h.3fwokq0) Prakondisi 30

4[.8.7](#_heading=h.1v1yuxt) Langkah-langkah 30

4[.8.8](#_heading=h.4f1mdlm) Pascakondisi 29

4[.9](#_heading=h.19c6y18) Use Case 9 30

4[.9.1](#_heading=h.32hioqz) Nama Use Case 30

4[.9.2](#_heading=h.1hmsyys) Tujuan 30

4[.9.3](#_heading=h.41mghml) Input 30

4[.9.4](#_heading=h.2grqrue) Output 31

4[.9.5](#_heading=h.vx1227) Skenario Utama 31

4[.9.6](#_heading=h.3fwokq0) Prakondisi 31

4[.9.7](#_heading=h.1v1yuxt) Langkah-langkah 31

4[.9.8](#_heading=h.4f1mdlm) Pascakondisi 31

4[.10](#_heading=h.19c6y18) Use Case 10 32

4[.10.1](#_heading=h.32hioqz) Nama Use Case 32

4[.10.2](#_heading=h.1hmsyys) Tujuan 32

4[.10.3](#_heading=h.41mghml) Input 32

4[.10.4](#_heading=h.2grqrue) Output 33

4[.10.5](#_heading=h.vx1227) Skenario Utama 33

4[.10.6](#_heading=h.3fwokq0) Prakondisi 33

4[.10.7](#_heading=h.1v1yuxt) Langkah-langkah 33

4[.10.8](#_heading=h.4f1mdlm) Pascakondisi33

4[.11](#_heading=h.19c6y18) Use Case 11 34

4[.11.1](#_heading=h.32hioqz) Nama Use Case 34

4[.11.2](#_heading=h.1hmsyys) Tujuan 34

4[.11.3](#_heading=h.41mghml) Input 34

4[.11.4](#_heading=h.2grqrue) Output 35

4[.11.5](#_heading=h.vx1227) Skenario Utama 35

4[.11.6](#_heading=h.3fwokq0) Prakondisi 35

4[.11.7](#_heading=h.1v1yuxt) Langkah-langkah 35

4[.11.8](#_heading=h.4f1mdlm) Pascakondisi35

4[.12](#_heading=h.19c6y18) Use Case 12 36

4[.10.1](#_heading=h.32hioqz) Nama Use Case 36

4[.10.2](#_heading=h.1hmsyys) Tujuan 36

4[.10.3](#_heading=h.41mghml) Input 36

4[.10.4](#_heading=h.2grqrue) Output 36

4[.10.5](#_heading=h.vx1227) Skenario Utama 37

4[.10.6](#_heading=h.3fwokq0) Prakondisi 37

4[.10.7](#_heading=h.1v1yuxt) Langkah-langkah 37

4[.10.8](#_heading=h.4f1mdlm) Pascakondisi37

**5**[**.**](#_heading=h.qsh70q) **Requirements Nonfungsional 37**

5[.1](#_heading=h.3as4poj) Requirements Performa 37

5[.2](#_heading=h.3tbugp1) Requirements Keselamatan 38

5[.3](#_heading=h.1pxezwc) Requirements Keamanan 38

5[.4](#_heading=h.49x2ik5) Atribut Kualitas Perangkat Lunak Lainnya 38

**6**[**.**](#_heading=h.2p2csry) **Requirements Lain** 38

[Lampiran A](#_heading=h.49x2ik5) 38

[Lampiran B](#_heading=h.49x2ik5) 39

A. Data Flow Diagrams 39

B. Class Diagrams 40

C. Sequence Diagram 41

1. Sequence Diagram berobat 41

2. Sequence Diagram Login41

3. Sequence Diagram register 42

4. Sequence Diagram riwayat berobat 42

# Pendahuluan

## Tujuan Penulisan Dokumen

Tujuan dokumen *SRS* ini dibuat ialah sebagai bentuk laporan proses pembentukan aplikasi MyDoc. Selain itu, dokumen ini juga berguna untuk mendeskripsikan secara rinci spesifikasi aplikasi yang kami buat yang dilakukan secara bertahap, sehingga pembaca dapat memahami lebih dalam tentang pembuatan aplikasi ini.

## Cakupan Produk

MyDocadalah aplikasi berbasis mobile yang berisi layanan kesehatan, terutama pendaftaran berobat. Aplikasi MyDoc bertujuan utama untuk mempermudah pendaftaran berobat antara pasien dengan layanan kesehatan, hingga mereka tidak perlu datang dan mengantri lagi, tapi dapat melakukannya melalui aplikasi ini. Pasien biasanya tidak suka untuk menunggu waktu lama-lama, sehingga dengan adanya aplikasi ini dapat memudahkan pendaftaran dan efisien dalam waktu.

## Konvensi Dokumen

Dokumen ini menggunakan konvensi berikut :

| Teks Sample | Deskripsi |
| --- | --- |
| *MyDoc* adalah aplikasi berbasis mobile yang berisi layanan kesehatan… | Elemen ditulis miring pada teks :   * Istilah asing * Nama kejadian aplikasi |
| **1.1. Tujuan Penulisan Dokumen** | Judul bab atau sub bab ditampilkan tebal |
| Tekan ***logout*** | Nama elemen antarmuka aplikasi |

## Referensi

* *Slides* Perkuliahan RPL : Analisis Kebutuhan
* *SRS Document “Centralized Monitoring of Professional Drivers”*
* *https://script.id/index.php?r=latihan/download&id=30/Contoh-isi-laporan-SRS.*

## Gambaran Umum

Dokumen SRS ini berisi tentang proses pembuatan aplikasi dari awal, mulai dari *functional requirements*, *non-functional requirements*, hingga *use case diagram.* Dokumen ini juga menjelaskan bagaimana aplikasi kami berjalan dan penjelasan serta fungsi tiap fitur-fitur yang ada di aplikasi kami bekerja.Dokumen disusun berdasarkan template yang telah diberikan oleh dosen kami dan juga rujukan dari *SRS Document “Centralized Monitoring of Professional Drivers”.*

# Overall Description

## Perspektif Produk

MyDoc dibangun berdasarkan kebutuhan antara layanan kesehatan dan pasien. Aplikasi ini dibuat untuk mempermudah penggunaan layanan kesehatan, terutama pendaftaran berobat. Pasien dapat mendaftar berobat tanpa harus pergi dan mengantri lagi seperti biasanya, tapi dapat dilakukan secara *online.* Data pasien dan pendaftarannya juga akan langsung terekam dalam *database* yang telah dibuat dengan bantuan resepsionis (*administrator)* dengan manajemen data, sehingga untuk mengumpulkan, menyimpan, hingga melacak data pasien dapat dilakukan dengan mudah. Dengan begitu, dibuatnya aplikasi ini dapat membantu memudahkan kedua belah pihak.

## Fungsi Produk

Fungsi dari MyDoc ini adalah untuk mempermudah antara layanan kesehatan dengan masyarakat, terutama untuk melakukan pendaftaran berobat secara *mobile*. MyDoc memiliki fungsionalitas seperti petugas kesehatan yang dapat membuat informasi, mengunggah jadwal praktek dokter, pendaftaran berobat, dan pembayaran. Adapun masyarakat juga dapat membuat akun, melihat informasi, melihat jadwal praktek dokter yang tersedia, melakukan pendaftaran, dan melakukan pembayaran ketika mendaftar berobat. Selain itu, ada admin yang bisa mengatur dan merekap data user pasien, mengatur transaksi pembayaran, dan juga menjaga keamanan akun user.

## Kelas dan Karakteristik Pengguna

* + 1. User MyDoc

- Memiliki hak akses untuk diri sendiri sesuai dengan yang diberikan

- Dapat menggunakan fitur yang tersedia

- Kemahiran dalam menggunakan aplikasi

* + 1. Administrator

- Memiliki hak untuk mengakses database

- Pengetahuan dalam menggunakan sistem

- Memiliki tanggung jawab dalam mengelola data yang tersedia, terutama data pasien

## Lingkungan Operasi

Karena MyDoc ditujukan untuk masyarakat dan penggunaan secara *mobile*, maka kami memutuskan untuk aplikasi MyDoc dapat berjalan pada *Android Operating System,* dikarenakan mayoritas masyarakat yang memiliki *android handphone* dan dalam mengakses MyDoc dapat dilakukan dimana saja.

## Batasan Perancangan dan Implementasi

Aplikasi *MyDoc* memiliki keterbatasan-keterbatasan yaitu sebagai berikut :

1. Terikat dengan kebijakan atau peraturan tiap instansi layanan kesehatan
2. Biaya yang akan dibutuhkan
3. Personil yang jujur dan memiliki tanggung jawab yang tinggi dalam menjadi *Administrator* dikarenakan data yang berada di database merupakan data pasien dan sangat rahasia

## Dokumentasi Pengguna

1. Login

* Memasukan nomor email dan password jika sudah memiliki akun
* Lalu klik tombol masuk jika sudah mengisi

1. Registrasi

* Memasukan nama, email, ,tempat tanggal lahir, alamat, password dan konfirmasi password
* Klik tombol daftar jika sudah mengisi

1. Dashboard

* Di halaman dashboard berisi berbagai macam menu seperti ; Home, Berobat, dan Profile.
* Di halaman dashboard terdapat fitur navigation bar
* Di halaman dashboard user atau pasien dapat memilih menu apa yang akan dilakukan pada aplikasi tersebut

1. Berobat

* Klik tombol yang memiliki logo -
* User atau pasien memilih dokter tujuan dan tempat tujuan yang ingin dereservasi
* Klik list atau nama dari dokter yang bersangkutan
* Lalu user di pindahkan ke halaman detail dokter untuk melihat apakah jadwal praktik tersedia pada jam dan hari tersebut, jika tersedia klik buat janji dan akan diaharakan ke halaman pembayaran
* Pembayaran dimana berisi dokter yang akan menangani beserta jadwal, metode pembayaran, dan detail pembayaran
* Sistem meminta User atau pasien memilih metode pembayaran yang ada
* Klik tombol buat pesanan yang berada paling bawah halaman

1. Ulasan

* Jika status riwayat berobat telah selesai user / pasien diminta untuk mengisi ulasan
* Sistem akan langsung memindahkan user atau pasien ke halaman riwayat pesanan yang berisikan list pesanan yang sudah pernah dilakukan dan sudah selesai
* Sistem menyediakan text box dimana user atau pasien dapat memberi masukan kepada dokter
* Klik tombol beri ulasan

1. Riwayat

* Klik tombol yang memiliki logo jam pada bagian navigation bar
* Sistem akan menampilkan riwayat pendaftaran yang belum selesai seperti ; menunggu pembayaran, proses, atau selesai
* Klik tombol batalkan pada bagian menunggu pembayaran jika tidak ingin melanjutkan berobat

1. Akun

* Klik tombol yang memiliki logo manusia
* Sistem akan menampilkan data user atau pasien
* Sistem akan menampilkan menu bantuan, kebijakan privasi, dan tentang aplikasi
* Klik tombol logout jika ingin mengeluarkan akun dari aplikasi

1. Navigation Bar

* Fitur ini terdapat pada bagian dashboard
* Fitur ini memiliki beberapa menu pilihan yang nantinya dapat merujuk pada fitur yang ada seperti ; riwayat, home, dan akun

## Asumsi dan Dependensi

- Data user yang diinputkan valid dan tidak ada di database

- Fungsionalitasnya dapat dimodifikasi, karena menyesuaikan kebutuhan sistem yang diinginkan dan kebijakan/peraturan yang ada

- Memerlukan kajian lebih lanjut jika adanya perubahan fungsionalitas

# Requirements Antarmuka Eksternal

## Antarmuka Pengguna

User terbagi menjadi Pasien dan Resepsionis, yang dimana Resepsionis juga bertugas sebagai Administrator untuk mengatur dan mengelola data pasien dalam aplikasi “MyDoc”. User dapat mengakses fitur-fitur yang ada melalui antarmuka berbasis mobile.

## Antarmuka Perangkat Keras

Kebutuhan minimum perangkat keras untuk menjalankan aplikasi “MyDoc” adalah sebagai berikut :

- RAM 4 GB

- Storage device yang memadai

- Mempunyai *wifi* atau *simcard untuk internet*

## Antarmuka Perangkat Lunak

Kebutuhan minimum perangkat lunak untuk menjalankan aplikasi “MyDoc” adalah sebagai berikut :

- OS Android

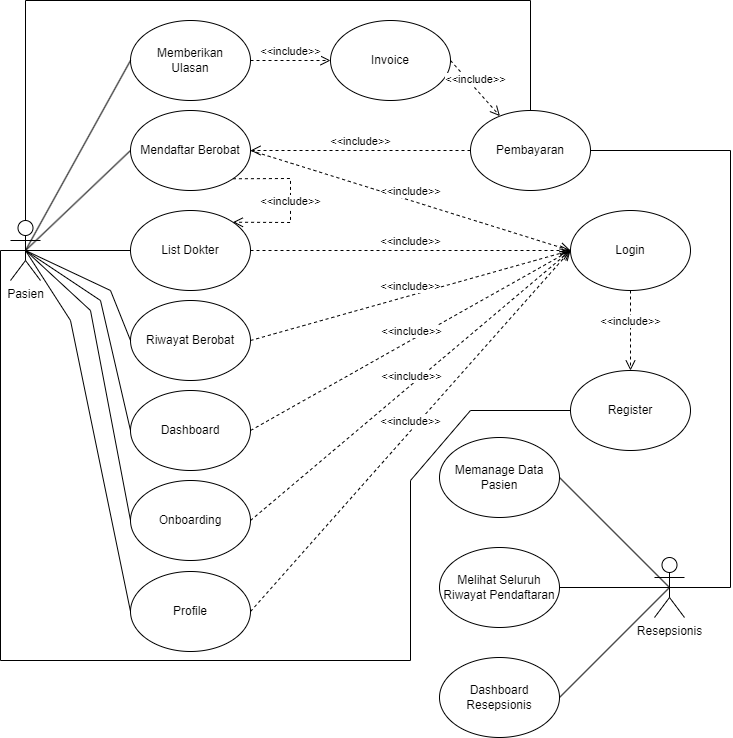
- Mempunyai upstorage sekitar 512 MB

- Mempunyai sambungan internet wifi atau data seluler untuk menggunakannya

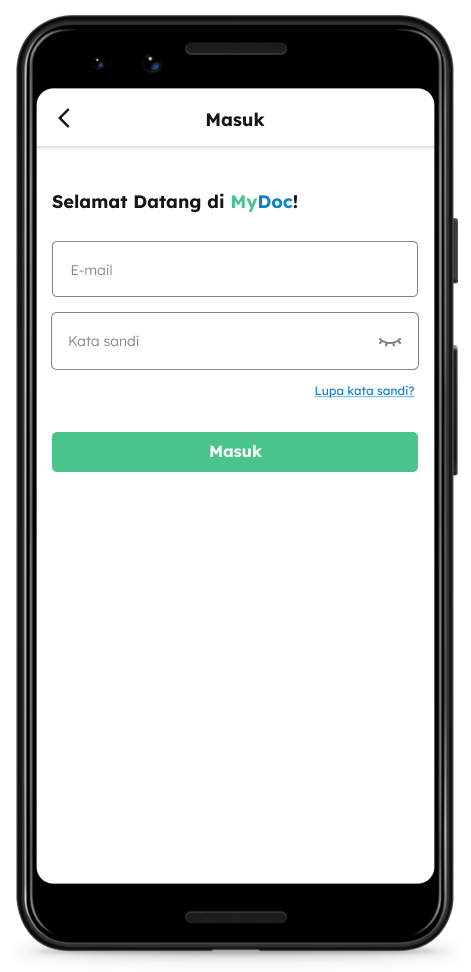
## Antarmuka Komunikasi

Aplikasi “MyDoc” menggunakan protocol HTTP karena aplikasi kami adalah aplikasi mobile.

# Fitur Sistem (Use Cases)



## Use Case 1



* + 1. ***Nama Use Case***

Login

* + 1. ***Tujuan***

Masuk ke dalam aplikasi

* + 1. ***Input***

Data email dan password user/pasien

* + 1. ***Output***

Dashboard

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. Membuka menu login |  |
|  | 2. Sistem menampilkan menu login |
| 3. Memasukkan nomor handphone dan password |  |
|  | 4. Sistem mengecek dan memvalidasi data yang dimasukkan |
| 5. User berhasil login dan masuk ke halaman utama aplikasi |  |

Skenario Alternatif

- Step 4 : email dan password yang dimasukkan User salah. Menampilkan tampilan “nomor email atau password yang dimasukkan salah”. Kembali ke step 2.

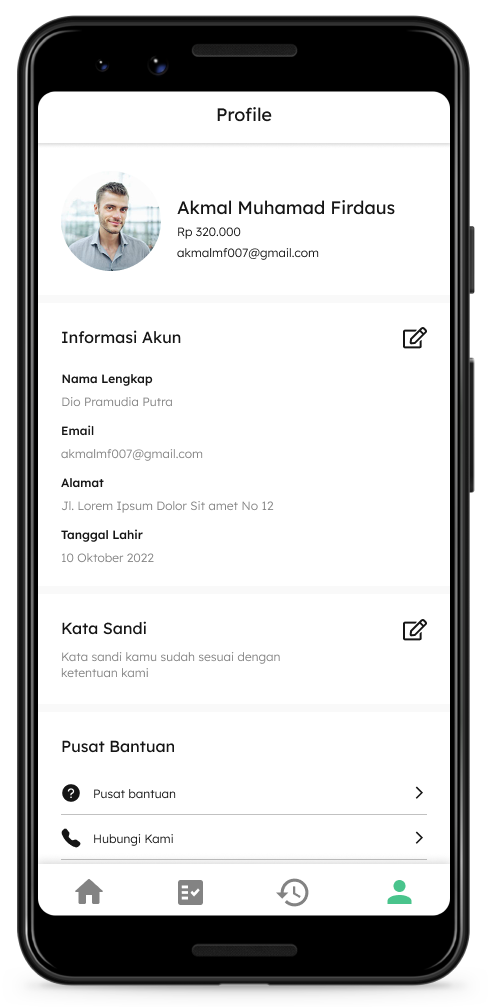
* + 1. ***Prakondisi***

Akun user sudah teregistrasi

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1*** : User memasukkan email dan password yang sudah teregister
       2. ***Langkah 2*** : User memencet tombol masuk
       3. ***Langkah 3*** : User berhasil masuk ke aplikasi
    2. ***Pascakondisi***

User memasuki halaman utama aplikasi dan berhak menggunakan layanan yang ada

* 1. **Use Case 2**

****

* + 1. ***Nama Use Case***

Profile

* + 1. ***Tujuan***

Menampilkan informasi akun user/pasien

* + 1. ***Input***

-

* + 1. ***Output***

Menampilkan informasi akun user/pasien

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. Mengklik tombol ***Profile*** |  |
|  | 2. Sistem menampilkan informasi user/pasien |
| 3. User kembali ke menu login |  |

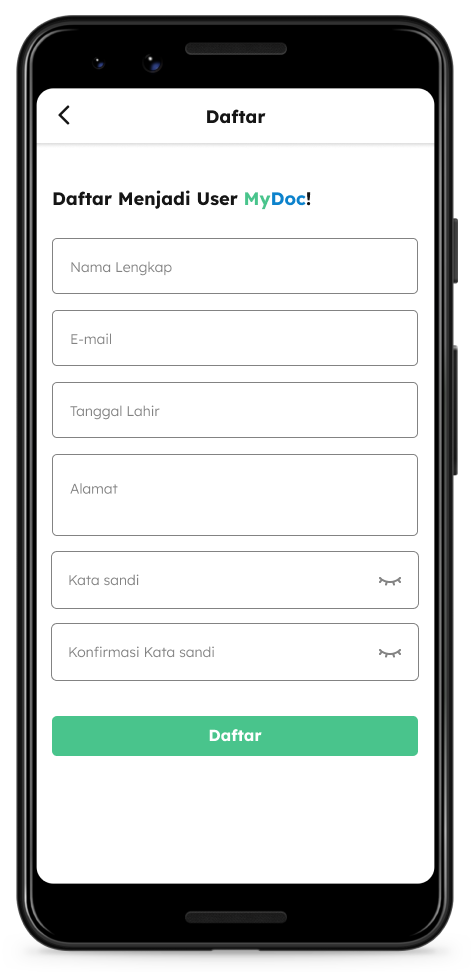
* + 1. ***Prakondisi***

Akun user sudah login

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1*** : User memencet tombol ***Profile***
    2. ***Pascakondisi***

Sistem menampilkan halaman ***Profile***

* 1. **Use Case 3**

****

* + 1. ***Nama Use Case***

Registrasi

* + 1. ***Tujuan***

Akun dapat terdaftar dan datanya tervalidasi dengan benar supaya dapat menggunakan layanan yang ada

* + 1. ***Input***

Data nama lengkap, tanggal lahir, alamat, email, password, dan konfirmasi password

* + 1. ***Output***

Dashboard

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. Menekan menu ***register*** |  |
|  | 2. Sistem menampilkan menu ***register*** |
| 3. Memasukkan nama lengkap, tanggal lahir, alamat, email, password, dan konfirmasi password |  |
|  | 4. Sistem memvalidasi data yang dimasukkan |
|  | 5. Sistem menambahkan data pasien ke dalam database |
| 6. User berhasil teregistrasi dan masuk ke halaman utama |  |

Skenario Alternatif :

- Step 4 : Data email yang dimasukkan sudah ada dalam database. Menampilkan tampilan “Anda sudah terdaftar dengan email tersebut”. Kembali ke step 2.

* + 1. ***Prakondisi***

- User berada di menu ***registrasi***

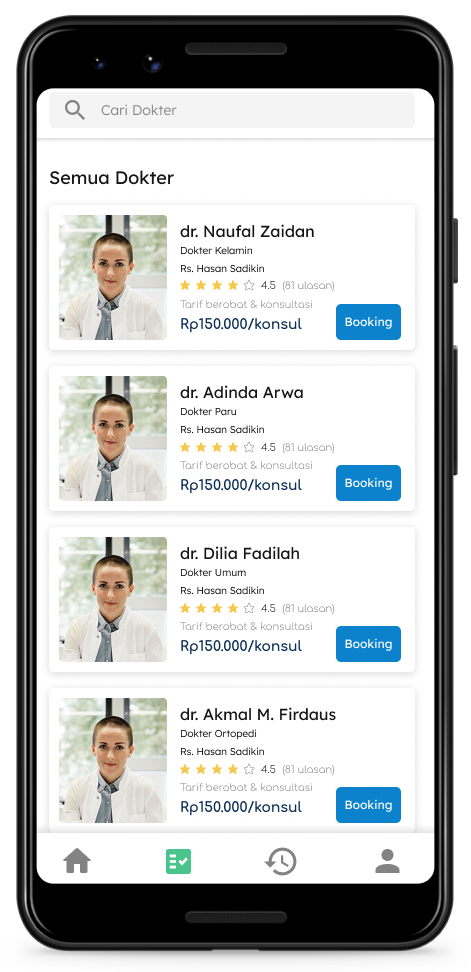
- User sudah menyiapkan data yang dibutuhkan untuk ***registrasi***

* + 1. **Langkah-Langkah**
       1. ***Langkah 1*** : User memasukkan nama lengkap, tanggal lahir, alamat, email, password, dan konfirmasi password
       2. ***Langkah 2*** : User memencet tombol ***daftar***
    2. ***Pascakondisi***

- Akun berhasil teregister

- User memasuki halaman utama

* 1. **Use Case 4**

****

* + 1. ***Nama Use Case***

List Dokter

* + 1. ***Tujuan***

Melihat rumah sakit dan dokter yang tersedia untuk membantu proses pendaftaran berobat

* + 1. ***Input***

-

* + 1. ***Output***

Tiap dokter beserta tempat praktik dengan jadwal dokter yang tersedia

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. Mengklik menu ***List Dokter*** |  |
|  | 2. Sistem menampilkan dokter yang tersedia |
| 3. Mengklik booking |  |
|  | 4. Sistem menampilkan detail dokter tersebut termasuk jadwal praktik yang dipilih |
| 5. User berhasil melihat jadwal yang tersedia dari klinik yang dicari |  |

Skenario Alternatif :

- Step 4 : Jadwal praktik tidak tersedia dikarenakan sedang kosong atau hari libur. Menampilkan pesan “Jadwal kosong” ke layar. Kembali ke step 2.

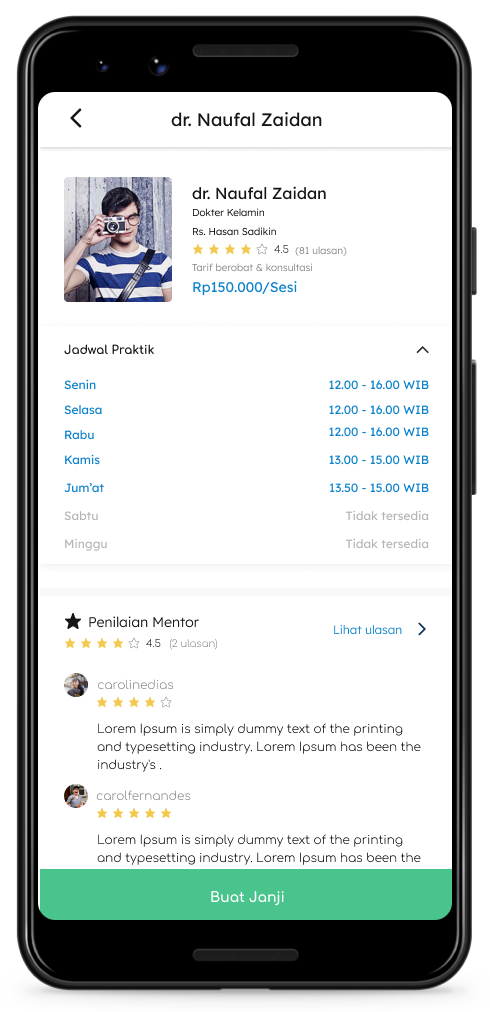
* + 1. ***Prakondisi***

User sudah login

* + 1. ***Langkah-Langkah***
       1. Langkah 1 : User memencet menu ***List Dokter***
       2. Langkah 2 : User mencari dokter yang diinginkan
       3. Langkah 3 : User meng-klik dokter yang dicari
    2. ***Pascakondisi***

Informasi dan jadwal praktik yang cari tertampil di layar user

* 1. **Use Case 5**

****

* + 1. ***Nama Use Case***

Mendaftar Berobat

* + 1. ***Tujuan***

Mendaftar berobat di rumah sakit dan dokter yang dituju

* + 1. ***Input***

Data rumah sakit yang dituju, tanggal dan jam berobat

* + 1. ***Output***

Menu pembayaran

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. Memilih dokter dan rumah sakit yang dituju |  |
|  | 2. Sistem detail informasi dokter |
| 3. User menekan Buat Janji |  |
|  | 4. Sistem memvalidasi data |
|  | 5. Data berhasil tervalidasi dan mengarahkan user ke menu pembayaran |
| 6. User teralihkan ke menu pembayaran |  |

Skenario Alternatif :

- Step 4: Tanggal berobat yang dipilih user tidak terdapat jadwal. Menampilkan pesan “Jadwal Kosong”. Kembali ke step 2.

* + 1. ***Prakondisi***

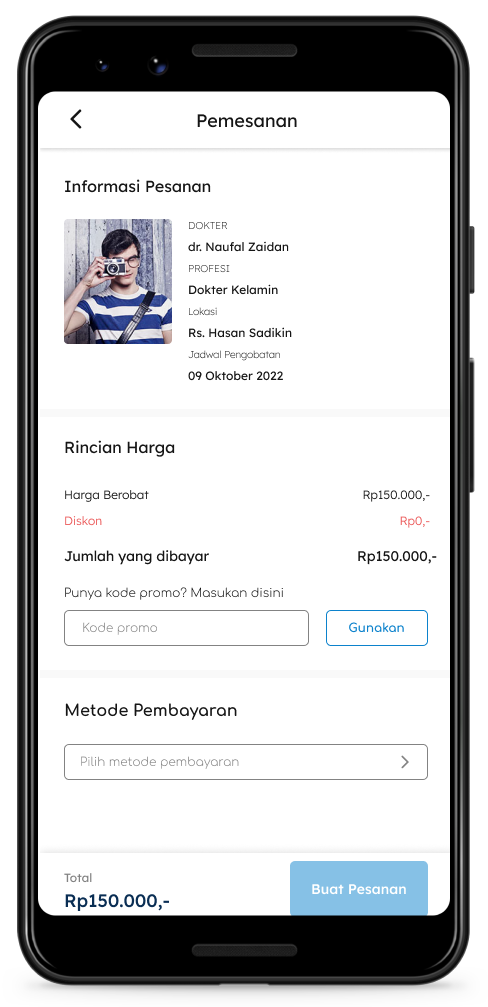
- User sudah login

- User sudah mengecek rumah sakit dan jadwal yang tersedia (opsional)

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1 :*** User memilih dokter dan rumah sakit yang dituju
       2. ***Langkah 2 :*** User menekan Buat Janji
    2. ***Pascakondisi***

User diarahkan ke halaman pembayaran

* 1. **Use Case 6**

****

* + 1. ***Nama Use Case***

Pembayaran

* + 1. ***Tujuan***

Menyelesaikan transaksi pendaftaran berobat dengan membayar sesuai yang nominal yang ditampilkan

* + 1. ***Input***

Mentransferkan uang sesuai nominal

* + 1. ***Output***

Pendaftaran berhasil

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
|  | 1. Sistem menampilkan menu pembayaran, mulai dari nomor rekening dan jumlah uang yang harus dibayar |
| 2. User mentransfer uang ke nomor rekening tersebut dengan nominal sesuai yang harus dibayar |  |
|  | 3. Sistem memverifikasi |
|  | 4. Terverifikasi dan tercatat oleh admin(resepsionis) |
| 5. User menerima pesan bahwa pembayaran telah terverifikasi dan pendaftaran berobat berhasil |  |

Skenario Alternatif :

- Step 3 : Uang yang ditransfer user tidak sesuai dengan nominal yang diberikan.

* + 1. ***Prakondisi***

- User sudah menyelesaikan proses pendaftaran

- User mempunyai rekening bank

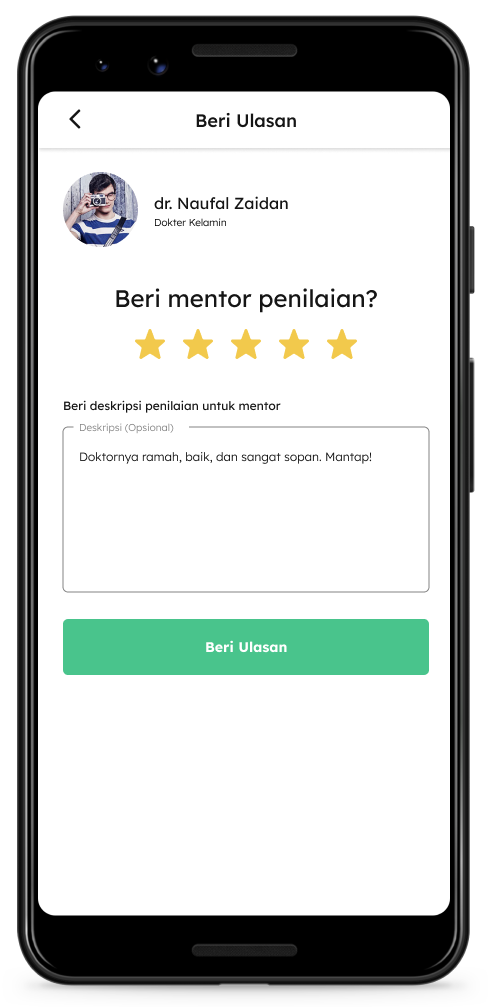
- User mempunyai saldo di rekening bank tersebut

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1 :*** User menyalin/mencatat nomor rekening dan nominal yang harus dibayarkan
       2. ***Langkah 2 :*** User mengisi data untuk transfer ke rekening yang dituju sebesar nominal yang ditampilkan
    2. ***Pascakondisi***

- Pembayaran berhasil diverifikasi

- Pendaftaran berhasil dan telah dicatat oleh resepsionis

* 1. **Use Case 7**

****

* + 1. ***Nama Use Case***

Beri Ulasan

* + 1. ***Tujuan***

Mengulas hasil pengobatan agar dokter/resepsionis dapat menjadikannya sebagai masukan

* + 1. ***Input***

Pasien mengulas hasil berobat

* + 1. ***Output***

Ulasan terkirim, kembali ke dashboard

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. Membuka riwayat berobat |  |
|  | 2. Jika user belum memberi ulasan, sistem otomatis me-redirect ke halaman ulasan |
|  | 3. Sistem menampilkan halaman formulir ulasan |
| 4. User menulis ulasan |  |
| 5. User menekan tombol Beri Ulasan |  |
|  | 6. Sistem memasukkan ulasan |

Skenario Alternatif :

- Step 1 : User belum melakukan pendaftaran berobat sama sekali. Menampilkan pesan “Anda belum mendaftar berobat”.

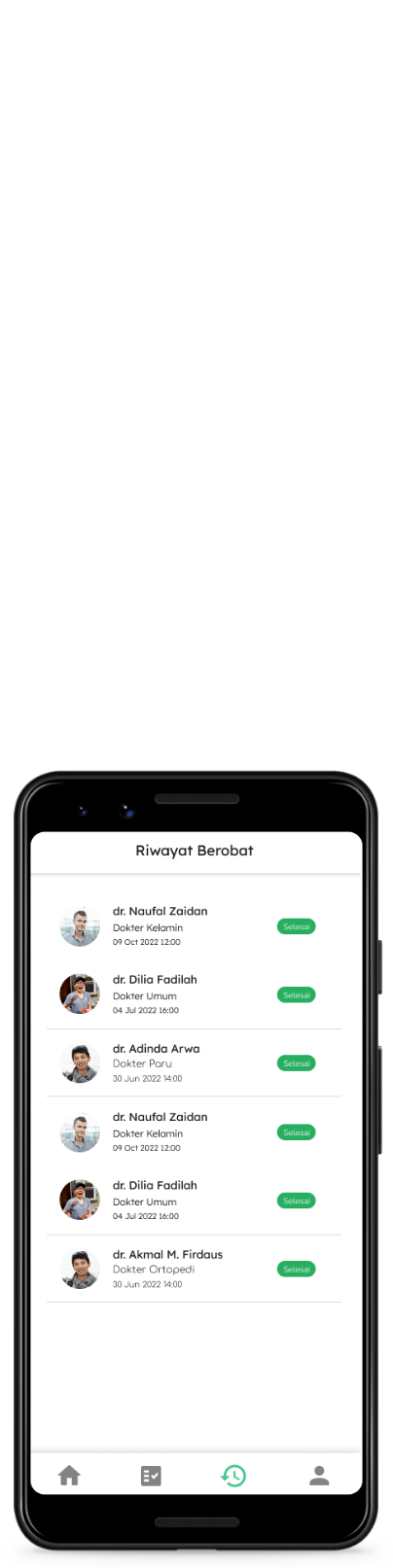
* + 1. ***Prakondisi***

User sudah melakukan pendaftaran berobat dan berobat

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1 :*** User masuk ke menu ***Riwayat Berobat***
       2. ***Langkah 2 :*** User menulis ulasan/pengalaman yang dialami
       3. ***Langkah 3 :*** User memencet tombol Beri Ulasan
    2. ***Pascakondisi***

User kembali ke menu utama

* 1. **Use Case 8**

****

* + 1. ***Nama Use Case***

Riwayat Berobat

* + 1. ***Tujuan***

Mengecek riwayat berobat

* + 1. ***Input***

-

* + 1. ***Output***

Sistem menampilkan riwayat berobat

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. Menekan menu **Riwayat Berobat** |  |
|  | 2. Sistem menampilkan riwayat berobat pasien |

* + 1. ***Prakondisi***

User sudah berhasil melakukan pendaftaran berobat (opsional)

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1 :*** User memencet menu ***Riwayat Berobat***
       2. ***Langkah 2 :*** User melihat informasi dan tiap riwayat yang ada dalam daftar
    2. ***Pascakondisi***

Riwayat beserta informasi yang diinginkan tertampil di layar user (Riwayat mungkin kosong)

* 1. **Use Case 9**
     1. ***Nama Use Case***

Mengelola Data dan Pendaftaran Pasien

* + 1. ***Tujuan***

Data dan hasil pendaftaran pasien tersimpan dengan rapi dan terstruktur di database “MyDoc”

* + 1. ***Input***

Data dan pendaftaran pasien

* + 1. ***Output***

Data tersimpan dalam database

* + 1. ***Skenario Utama***
    2. ***Prakondisi***

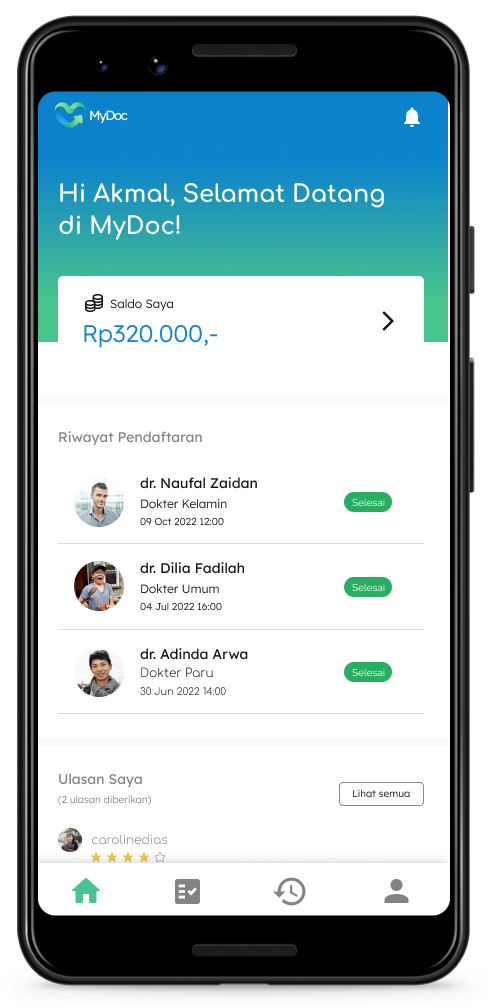
- Akun pasien sudah teregister

- Akun pasien sudah melakukan pendaftaran berobat (opsional)

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1 :*** Resepsionis (administrator) dengan bantuan data manajemen mengumpulkan dan menganalisis informasi tentang data pasien
       2. ***Langkah 2 :*** Resepsionis dan manajemen data menyimpan ke dalam database
    2. ***Pascakondisi***

Data pasien tersimpan dalam database

* 1. **Use Case 10**

****

* + 1. ***Nama Use Case***

Dashboard

* + 1. ***Tujuan***

Memberikan navigation kepada user dan sekilas info pasien (Saldo, ulasan yang diberikan, riwayat pendaftaran, dll)

* + 1. ***Input***

-

* + 1. ***Output***

System menampilkan halaman dashboard

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. User mendownload aplikasi “MyDoc” |  |
| 2. User melakukan login/register |  |
|  | 3. Sistem memvalidasi data |
|  | 4. Sistem menampilkan dashboard |
| 5. User sudah berada di dashboard |  |

* + 1. ***Prakondisi***

- User sudah menginstall aplikasi “MyDoc” di Playstore

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1 :*** User mendownload aplikasi terlebih dahulu
       2. ***Langkah 2 :*** User log in / register
    2. ***Pascakondisi***

User sudah log in / register

* 1. **Use Case 11**

****

* + 1. ***Nama Use Case***

Onboarding

* + 1. ***Tujuan***

Memberikan navigation kepada user untuk memilih masuk atau mendaftar

* + 1. ***Input***

-

* + 1. ***Output***

System menampilkan halaman OnBoarding

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. User mendownload aplikasi “MyDoc” |  |
|  | 2. Sistem menampilkan dashboard |
| 3. User berada pada layout OnBoarding |  |

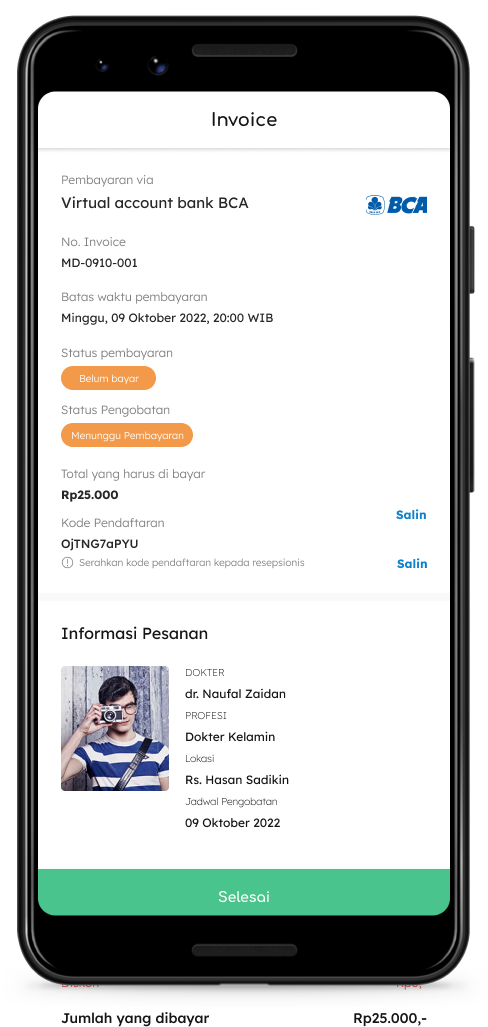
* + 1. ***Prakondisi***

- User sudah menginstall aplikasi “MyDoc” di Playstore

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1 :*** User mendownload aplikasi terlebih dahulu
    2. ***Pascakondisi***

OnBoarding tertampil

* 1. **Use Case 12**

****

* + 1. ***Nama Use Case***

Invoice

* + 1. ***Tujuan***

Informasi mengenai status pembayaran berobat

* + 1. ***Input***

-

* + 1. ***Output***

System menampilkan halaman Invoice

* + 1. ***Skenario Utama***

| Actor Actions | System Actions |
| --- | --- |
| 1. User mendaftar berobat |  |
| 2. User melakukan pembayaran |  |
|  | 3. Sistem menampilkan invoice |

* + 1. ***Prakondisi***

- User mendaftar berobat dan melakukan pembayaran, invoice untuk melakukan pengecekan status pembayaran

* + 1. ***Langkah-Langkah***
       1. ***Langkah 1 :*** User mendaftar berobat pada salahsatu dokter
       2. ***Langkah 2 :*** User melakukan pendaftaran
    2. ***Pascakondisi***

Invoice tertampil

# Requirements Nonfungsional

## Requirements Performa

1. MyDoc tersedia selama 24 jam, tapi tidak berlaku pada saat aplikasi maintenance
2. Transisi pada setiap halaman memiliki waktu yang sama sekitar 1 detik

## Requirements Keamanan

1. User MyDoc diharuskan untuk memverifikasi email yang sudah di cantumkan pada saat membuat registrasi.
2. Seluruh data yang ada di server di amankan oleh jasa pengelola dan penyimpanan data yang bisa dijamin keamanannya.

## Atribut Kualitas Perangkat Lunak Lainnya

**Requirements Kehandalan**

1. Sistem harus bekerja secara semestinya tanpa gangguan apapun.
2. Sistem harus dapat memberikan hasil kinerja yang baik dan hasilnya akan digunakan sebagai acuan untuk perkembangan dikemudian hari.

**Requirements Kehandalan**

1. Sistem harus mudah dipahami dan mudah digunakan dari antarmuka sampai fungsi.
2. Sistem harus memandu user dalam penggunaan aplikasi secara keseluruhan

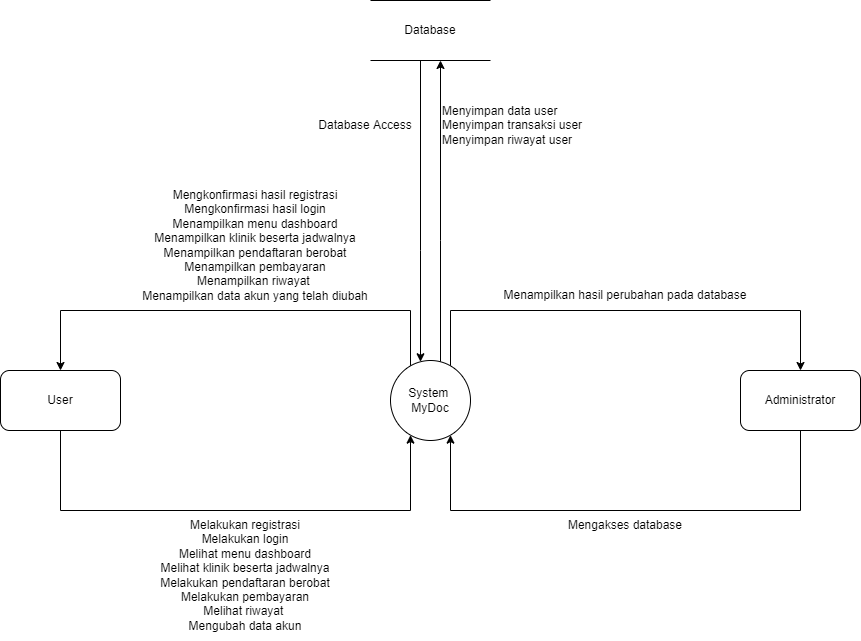
# Requirements Lain

**Lampiran A: Daftar Kata-Kata Sukar**

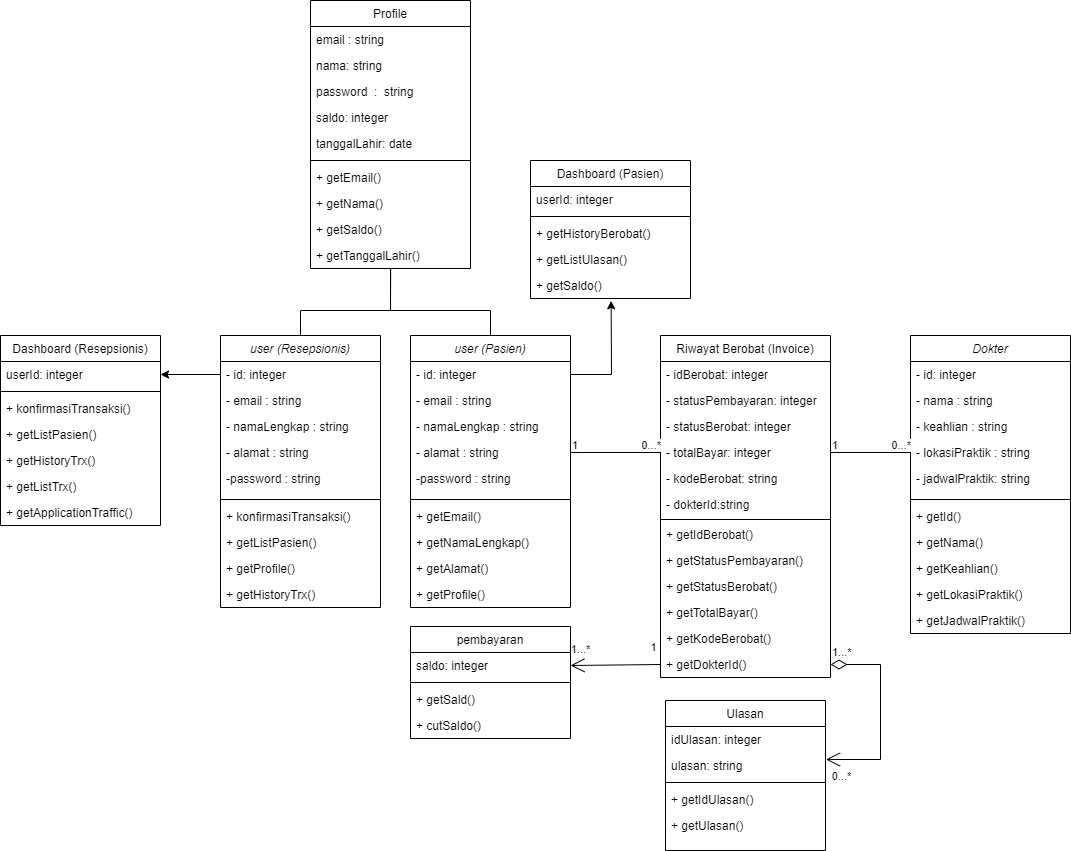
* SRS - Software Requirements System
* Functional Requirements - Fitur fungsional yang terdapat dalam aplikasi
* Non-Functional Requirements - Fitur non-fungsional yang terdapat dalam aplikasi
* User - Pengguna aplikasi

**Lampiran B: Analysis Models**

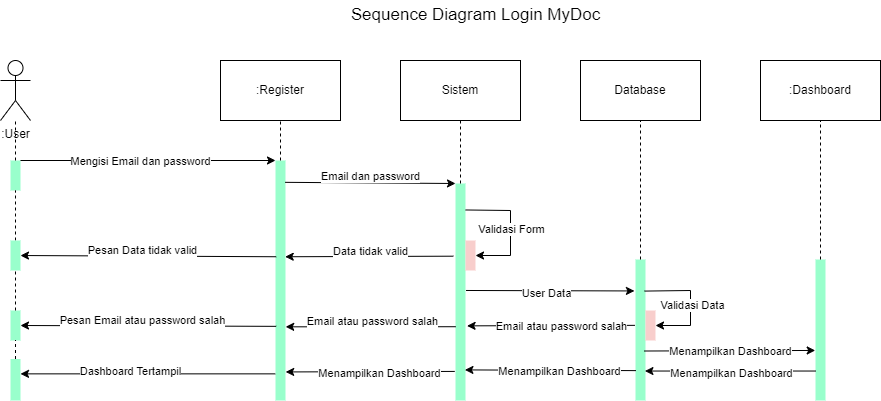
1. **Data Flow Diagrams**

****

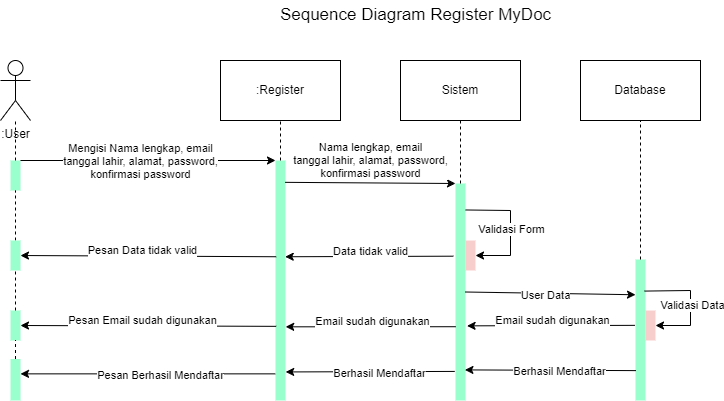
1. **Class Diagrams**

****

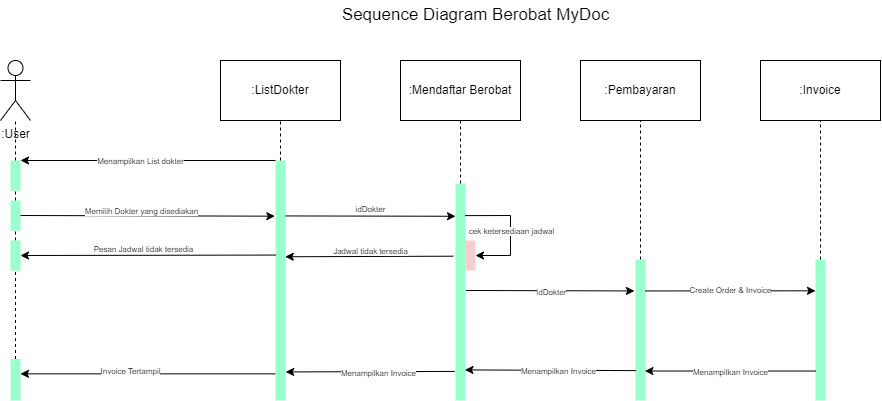
1. **Sequence Diagram**
2. Sequence Diagram berobat
3. Sequence Diagram Login



1. Sequence Diagram Register



1. Sequence Diagram Riiwayat Berobat

****