

**DPPL-01**

## **DESKRIPSI PERANCANGAN PERANGKAT LUNAK**

### **MyDoc**

untuk:

Layanan Kesehatan


Dipersiapkan oleh:

1. Akmal Muhamad Firdaus (1301204188)
2. Dilia Fadilah Mutmainah (1301204383)
3. Adinda Arwa Salsabil (1301204011)
4. M. Naufal Zaidan (1301204338)

Program Studi S1 Informatika

Fakultas Informatika, Universitas Telkom

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

	<b>Prodi S1- Informatika Universitas Telkom</b>	<b>Nomor Dokumen</b>		<b>Halaman</b>
		<b><i>DPPL-MyDoc 01</i></b>		<i>79</i>
		<b>Revisi</b>		<i>05-11-22</i>

## DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	
<b>G</b>	

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

## Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

# Daftar Isi

<b>Daftar Isi</b>	<b>4</b>
Daftar Tabel	<b>11</b>
Daftar Gambar	<b>13</b>
1. Pendahuluan	<b>15</b>
1.1 Tujuan Penulisan Dokumen	15
1.2 Lingkup Masalah	15
1.3 Definisi dan Istilah	15
Tabel 1.1 Definisi dan Istilah	16
1.4 Referensi	16
2 Perancangan Global	<b>17</b>
2.1 Rencana Lingkungan Implementasi	17
2.2 Deskripsi Arsitektur Perangkat Lunak	17
Tabel 2.1 Deskripsi Arsitektur Perangkat Lunak	18
3 Perancangan Rinci	<b>19</b>
Gambar 3.1 Usecase Diagram	19
3.1 Realisasi Use Case	19
Tabel 3.1 Tabel Use Case	20
3.1.1 Use Case #1 Register	21
3.1.1.1 Use Case Scenario #1 Register	21
3.1.1.2 UI Design dan Deskripsi Objek UI #1 Register	21
Gambar 3.2 UI Register	21
Tabel 3.2 Tabel Deskripsi Objek UI Register	22
Tabel 3.3 Page Register - MD - REG	22
3.1.1.3 Identifikasi Object dan Tipe nya #1 Register	22
Tabel 3.4 Tabel Object Perancangan Register	22
3.1.1.4 Sequence Diagram #1 Register	23
Gambar 3.3 Sequence Diagram Register	23
3.1.2 Use Case #2 Login	24
3.1.2.1 Use Case Scenario #2 Login	24
3.1.2.2 UI Design dan Deskripsi Objek UI #2 Login	24
Gambar 3.4 UI Login	24
Tabel 3.5 Tabel Deskripsi Objek UI Login	24
Tabel 3.6 MD-LGN - Page Login	25
3.1.2.3 Identifikasi Object Baru & Tipe Kelas #2 Login	25
Tabel 3.7 Tabel Object Perancangan Login	25
3.1.2.4 Sequence Diagram #2 Login	25
Gambar 3.5 Sequence Diagram Login	25
3.1.3 Use Case #3 Dashboard	26
3.1.3.1 Use Case Scenario #3 Dashboard	26

3.1.3.2 UI Design dan Deskripsi Objek UI #3 Dashboard	26
Gambar 3.6 UI Dashboard	26
Tabel 3.8 Tabel Deskripsi Objek UI Dashboard	26
Tabel 3.9 MD-DSB - Page Dashboard	27
3.1.3.3 Identifikasi Object Baru & Tipe Kelas #3 Dashboard	27
Tabel 3.10 Tabel Object Perancangan Dashboard	27
3.1.3.4 Sequence Diagram #3 Dashboard	28
Gambar 3.7 Sequence Diagram Dashboard	28
3.1.4 Use Case #4 List Dokter	29
3.1.4.1 Use Case Scenario #4 List Dokter	29
3.1.4.2 UI Design dan Deskripsi Objek UI #4 List Dokter	29
Gambar 3.8 UI List Dokter	29
Tabel 3.11 Tabel Deskripsi Objek UI List Dokter	29
Tabel 3.12 Page List Dokter	30
3.1.4.3 Identifikasi Object Baru & Tipe Kelas #4 List Dokter	30
Tabel 3.13 Tabel Object Perancangan List Dokter	30
3.1.4.4 Sequence Diagram #4 List Dokter	31
Gambar 3.9 Sequence Diagram ListDokter	31
3.1.5 Use Case #5 Detail Dokter	32
3.1.5.1 Use Case Scenario #5 Detail Dokter	32
3.1.5.2 UI Design dan Deskripsi Objek UI #5 Detail dokter	32
Gambar 3.10 UI Detail Dokter	32
Tabel 3.14 Tabel Deskripsi Objek UI Detail Dokter	32
Tabel 3.15 MD-DTL - Page Detail Dokter	33
3.1.5.3 Identifikasi Object Baru & Tipe Kelas #5 Detail Dokter	33
Tabel 3.16 Tabel Object Perancangan Detail Dokter	33
3.1.5.4 Sequence Diagram #5 Detail Dokter	34
Gambar 3.11 Sequence Diagram Detail Dokter	34
3.1.6 Use Case #6 Mendaftar Berobat	35
3.1.6.1 Use Case Scenario #6 Mendaftar Berobat	35
3.1.6.2 UI Design dan Deskripsi Objek UI #6 Mendaftar Berobat	35
Gambar 3.12 UI Mendaftar Berobat	35
Tabel 3.17 Tabel Deskripsi Objek UI Mendaftar Berobat	35
Tabel 3.18 Page Mendaftar Berobat	36
3.1.6.3 Identifikasi Object Baru & Tipe Kelas #6 Mendaftar Berobat	36
Tabel 3.19 Tabel Object Perancangan Mendaftar Berobat	36
3.1.6.4 Sequence Diagram #6 Mendaftar Berobat	36
Gambar 3.13 Sequence Diagram Mendaftar Berobat	36
3.1.7 Use Case #7 Riwayat Berobat	37
3.1.7.1 Use Case Scenario #7 Riwayat Berobat	37
3.1.7.2 UI Design dan Deskripsi Objek UI #7 Riwayat berobat	37
Gambar 3.14 UI Riwayat Berobat	37
Tabel 3.20 Tabel Deskripsi Objek UI Riwayat Berobat	37
Tabel 3.21 Page Riwayat Berobat	38
3.1.7.3 Identifikasi Object Baru & Tipe Kelas #7 Riwayat Berobat	38

Tabel 3.22 Tabel Object Perancangan Riwayat Berobat	38
3.1.7.4 Sequence Diagram #7 Riwayat Berobat	38
Gambar 3.15 Sequence Diagram Riwayat Berobat	38
3.1.8 Use Case #8 Edit Profile	39
3.1.8.1 Use Case Scenario #8 Edit Profile	39
3.1.8.2 UI Design dan Deskripsi Objek UI #8 Edit Profile	39
Gambar 3.16 UI MD-EDT1 Gambar 3.17 UI MD-EDT2	39
Tabel 3.23 Tabel Deskripsi Objek UI Edit Profile	39
Tabel 3.24 Page Edit Profile - MD - EDT2	40
3.1.8.3 Identifikasi Object Baru & Tipe Kelas #8 Edit Profile	40
Tabel 3.25 Tabel Object Perancangan Edit Profile	40
3.1.8.4 Sequence Diagram #8 Edit Profile	40
Gambar 3.18 Sequence Diagram Edit Profile	40
3.1.9 Use Case #9 Pembayaran	41
3.1.9.1 Use Case Scenario #9 Pembayaran	41
3.1.9.2 UI Design dan Deskripsi Objek UI #9 Pembayaran	41
Gambar 3.19 UI MD-CHK	41
Tabel 3.26 Tabel Deskripsi Objek UI Pembayaran	41
Tabel 3.27 Pembayaran - MD - CHK	42
3.1.9.3 Identifikasi Object Baru & Tipe Kelas #9 Pembayaran	42
Tabel 3.28 Tabel Object Perancangan Pembayaran	42
3.1.9.4 Sequence Diagram #9 Pembayaran	42
Gambar 3.20 Sequence Diagram Pembayaran	42
3.1.10 Use Case #10 Ulasan	43
3.1.10.1 Use Case Scenario #10 Ulasan	43
3.1.10.2 UI Design dan Deskripsi Objek UI #6 Ulasan	43
Gambar 3.21 UI Ulasan	43
Tabel 3.29 Tabel Deskripsi Objek UI Ulasan	44
Tabel 3.30 Page Ulasan	44
3.1.10.3 Identifikasi Object Baru & Tipe Kelas #6 Ulasan	44
Tabel 3.31 Tabel Object Perancangan Ulasan	44
3.1.10.4 Sequence Diagram #6 Ulasan	45
Gambar 3.22 Sequence Diagram Ulasan	45
3.1.11 Use Case #11 Onboarding	46
3.1.11.1 Use Case Scenario #11 Onboarding	46
3.1.11.2 UI Design dan Deskripsi Objek UI #11 Onboarding	46
Gambar 3.23 UI Onboarding	46
Tabel 3.32 Tabel Deskripsi Objek UI Onboarding	46
Tabel 3.33 Page Onboarding	47
3.1.11.3 Identifikasi Object Baru & Tipe Kelas #11 Onboarding	47
Tabel 3.34 Tabel Object Perancangan Onboarding	47
3.1.11.4 Sequence Diagram #11 Onboarding	47
Gambar 3.24 Sequence Diagram Onboarding	47
3.1.12 Use Case #12 Invoice	48
3.1.12.1 Use Case Scenario #12 Invoice	48

3.1.12.2 UI Design dan Deskripsi Objek UI #12 Invoice	48
Gambar 3.25 UI MD-INV	48
Tabel 3.35 Tabel Deskripsi Objek UI Invoice	48
Tabel 3.36 Page Invoice - MD - INV	49
3.1.12.3 Identifikasi Object Baru & Tipe Kelas #12 Invoice	49
Tabel 3.37 Tabel Object Perancangan Invoice	49
3.1.12.4 Sequence Diagram #12 Invoice	49
Gambar 3.26 Sequence Diagram Invoice	49
3.1.13 Use Case #13 Dashboard Instansi Kesehatan	50
3.1.13.1 Use Case Scenario #13 Dashboard Instansi Kesehatan	50
3.1.13.2 UI Design dan Deskripsi Objek UI #13 Dashboard Instansi Kesehatan	50
Gambar 3.27 UI DashBoard Instansi Kesehatan	50
Tabel 3.38 Tabel Deskripsi Objek UI Dashboard Instansi Kesehatan	50
Tabel 3.39 Page Dashboard Instansi Kesehatan - MD - DSB	51
3.1.13.3 Identifikasi Object Baru & Tipe Kelas #13 Dashboard Instansi Kesehatan	51
Tabel 3.40 Tabel Object Perancangan Dashboard Instansi Kesehatan	51
3.1.13.4 Sequence Diagram #12 Dashboard Instansi Kesehatan	51
Gambar 3.28 Sequence Dashboard Instansi Kesehatan	51
3.1.14 Use Case #14 Menambah List Dokter	52
3.1.14.1 Use Case Scenario #14 Menambah List Dokter	52
3.1.14.2 UI Design dan Deskripsi Objek UI #14 Menambah List Dokter	52
Gambar 3.29 UI MD-ADD_DCT	52
Tabel 3.41 Tabel Deskripsi Objek UI Menambah List Dokter	52
Tabel 3.42 Page Menambah List Dokter - MD - ADD_DCT	53
3.1.14.3 Identifikasi Object Baru & Tipe Kelas #14 Menambah List Dokter	53
Tabel 3.43 Tabel Object Perancangan	53
3.1.14.4 Sequence Diagram #14 Menambah List Dokter	53
Gambar 3.30 Sequence Diagram Menambah List Dokter	53
3.1.15 Use Case #15 Verifikasi Antrian	54
3.1.15.1 Use Case Scenario #15 Verifikasi Antrian	54
3.1.15.2 UI Design dan Deskripsi Objek UI #15 Verifikasi Antrian	54
Gambar 3.31 UI MD-VRF	54
Tabel 3.44 Tabel Deskripsi Objek UI Verifikasi Antrian	54
Tabel 3.45 Page Verifikasi Antrian - MD - VRF	55
3.1.15.3 Identifikasi Object Baru & Tipe Kelas #15 Verifikasi Antrian	55
Tabel 3.43 Tabel Object Perancangan	55
3.1.15.4 Sequence Diagram #15 Verifikasi Antrian	55
Gambar 3.32 Sequence Diagram Verifikasi Antrian	55
3.1.16 Use Case #16 Dashboard Admin	56
3.1.16.1 Use Case Scenario #16 Dashboard Admin	56
3.1.16.2 UI Design dan Deskripsi Objek UI #15 Dashboard Admin	56
Gambar 3.33 UI Dashboard Admin	56
Tabel 3.46 Tabel Deskripsi Objek UI Dashboard Admin	56
Tabel 3.47 Tabel Deskripsi Objek UI Dashboard Admin	57
3.1.16.3 Identifikasi Object Baru & Tipe Kelas #16 Dashboard Admin	57



Tabel 3.48 Tabel Object Perancangan Dashboard Admin	57
3.1.16.4 Sequence Diagram #15 Dashboard Admin	58
Gambar 3.34 Sequence Diagram Dashboard Admin	58
3.1.17 Use Case #17 Mengelola data Riwayat Berobat	58
3.1.17.1 Use Case Scenario #17 Mengelola data Riwayat Berobat	58
3.1.17.2 UI Design dan Deskripsi Objek UI #17 Mengelola data Riwayat Berobat	59
Tabel 3.49 Tabel Deskripsi Objek UI Mengelola Data Riwayat Berobat	59
Tabel 3.50 Mengelola Data Riwayat Berobat Admin	60
Tabel 3.51 Tabel Object Perancangan Mengelola Data Riwayat Berobat	60
3.1.17.3 Sequence Diagram	60
Gambar 3.36 Sequence Mengelola Riwayat Berobat Instansi Kesehatan	61
3.1.18 Use Case #18 Melihat Data Riwayat Berobat	62
3.1.18.1 Use Case Scenario #18 Melihat Data Riwayat Berobat	62
3.1.18.2 UI Design dan Deskripsi Objek UI #18 Melihat Data Riwayat Berobat	62
Gambar 3.37 UI MD-SHRB	62
Tabel 3.52 Tabel Deskripsi Objek UI Melihat Data Riwayat Berobat	62
Tabel 3.53 Tabel Page Melihat Data Riwayat Berobat	62
3.1.18.3 Identifikasi Object Baru & Tipe Kelas #18 Melihat Data Riwayat Berobat	63
Tabel 3.54 Tabel Object Perancangan Melihat Data Riwayat Berobat	63
3.1.18.4 Sequence Diagram #18 Melihat Data Riwayat Berobat	63
Gambar 3.38 Sequence Diagram Melihat Data Riwayat Berobat	63
3.1.19 Use Case #19 Membuat Akun Instansi Kesehatan	64
3.1.19.1 Use Case Scenario #19 Membuat Akun Instansi Kesehatan	64
3.1.19.2 UI Design dan Deskripsi Objek UI #19 Membuat Akun Instansi Kesehatan	64
Gambar 3.39 UI MD-ADDIST	64
Tabel 3.55 Tabel Deskripsi Objek UI	64
Tabel 3.56 Page Membuat Akun Instansi Kesehatan - MD - ADDIST	65
3.1.19.3 Identifikasi Object Baru & Tipe Kelas #19 Membuat Akun Instansi Kesehatan	65
Tabel 3.57 Tabel Object Perancangan Membuat Akun Instansi Kesehatan	65
3.1.19.4 Sequence Diagram #19 Membuat Akun Instansi Kesehatan	65
Gambar 3.40 Sequence Diagram Membuat Akun Instansi Kesehatan	65
3.1.20 Use Case #20 Mengelola User	66
3.1.20.1 Use Case Scenario #20 Mengelola User	66
3.1.20.2 UI Design dan Deskripsi Objek UI #20 Mengelola User	66
Gambar 3.41 UI MD-EUSR1	66
Gambar 3.42 UI MD-EUSR2	67
Tabel 3.58 Tabel Deskripsi Objek UI Mengelola User	67
Tabel 3.59 Tabel Page List Pasien	67
Tabel 3.60 Tabel Page Tambah Pasien	67
3.1.20.3 Identifikasi Object Baru & Tipe Kelas #20 Mengelola User	68
Tabel 3.61 Tabel Object Perancangan Mengelola User	68
3.1.20.4 Sequence Diagram #20 Mengelola User	68
Gambar 3.43 Sequence Diagram Mengelola User	68
3.2 Diagram Kelas Keseluruhan	70
Gambar 3.44 Diagram Class Keseluruhan	70

Tabel 3.62 Tabel Kelas	72
3.3 Perancangan Data / Basis Data	73
Gambar 3.45 ER-Diagram MyDoc	73
3.4 Perancangan Algoritma dan/atau Query	74
4. Matriks Kerunutan (Requirement Traceability Matrix)	78
Tabel 3.63 Tabel Requirement Traceability Matrix	79

## Daftar Tabel

Tabel 1.1 Definisi dan Istilah	16
Tabel 2.1 Deskripsi Arsitektur Perangkat Lunak	18
Tabel 3.1 Tabel Use Case	20
Tabel 3.2 Tabel Deskripsi Objek UI Register	22
Tabel 3.3 Page Register - MD - REG	22
Tabel 3.4 Tabel Object Perancangan Register	22
Tabel 3.5 Tabel Deskripsi Objek UI Login	24
Tabel 3.6 MD-LGN - Page Login	25
Tabel 3.7 Tabel Object Perancangan Login	25
Tabel 3.8 Tabel Deskripsi Objek UI Dashboard	26
Tabel 3.9 MD-DSB - Page Dashboard	27
Tabel 3.10 Tabel Object Perancangan Dashboard	27
Tabel 3.11 Tabel Deskripsi Objek UI List Dokter	29
Tabel 3.12 Page List Dokter	30
Tabel 3.13 Tabel Object Perancangan List Dokter	30
Tabel 3.14 Tabel Deskripsi Objek UI Detail Dokter	32
Tabel 3.15 MD-DTL - Page Detail Dokter	33
Tabel 3.16 Tabel Object Perancangan Detail Dokter	33
Tabel 3.17 Tabel Deskripsi Objek UI Mendaftar Berobat	35
Tabel 3.18 Page Mendaftar Berobat	36
Tabel 3.19 Tabel Object Perancangan Mendaftar Berobat	36
Tabel 3.20 Tabel Deskripsi Objek UI Riwayat Berobat	37
Tabel 3.21 Page Riwayat Berobat	38
Tabel 3.22 Tabel Object Perancangan Riwayat Berobat	38
Tabel 3.23 Tabel Deskripsi Objek UI Edit Profile	39
Tabel 3.24 Page Edit Profile - MD - EDT2	40
Tabel 3.25 Tabel Object Perancangan Edit Profile	40
Tabel 3.26 Tabel Deskripsi Objek UI Pembayaran	41
Tabel 3.27 Pembayaran - MD - CHK	42
Tabel 3.28 Tabel Object Perancangan Pembayaran	42
Tabel 3.29 Tabel Deskripsi Objek UI Ulasan	44
Tabel 3.30 Page Ulasan	44
Tabel 3.31 Tabel Object Perancangan Ulasan	44
Tabel 3.32 Tabel Deskripsi Objek UI Onboarding	46
Tabel 3.33 Page Onboarding	47
Tabel 3.34 Tabel Object Perancangan Onboarding	47
Tabel 3.35 Tabel Deskripsi Objek UI Invoice	48
Tabel 3.36 Page Invoice - MD - INV	49
Tabel 3.37 Tabel Object Perancangan Invoice	49
Tabel 3.38 Tabel Deskripsi Objek UI Dashboard Instansi Kesehatan	50
Tabel 3.39 Page Dashboard Instansi Kesehatan - MD - DSB	51
Tabel 3.40 Tabel Object Perancangan Dashboard Instansi Kesehatan	51
Tabel 3.41 Tabel Deskripsi Objek UI Menambah List Dokter	52

Tabel 3.42 Page Menambah List Dokter - MD - ADD_DCT	53
Tabel 3.43 Tabel Object Perancangan	53
Tabel 3.44 Tabel Deskripsi Objek UI Verifikasi Antrian	54
Tabel 3.45 Page Verifikasi Antrian - MD - VRF	55
Tabel 3.43 Tabel Object Perancangan	55
Tabel 3.46 Tabel Deskripsi Objek UI Dashboard Admin	56
Tabel 3.47 Tabel Deskripsi Objek UI Dashboard Admin	57
Tabel 3.48 Tabel Object Perancangan Dashboard Admin	57
Tabel 3.49 Tabel Deskripsi Objek UI Mengelola Data Riwayat Berobat	59
Tabel 3.50 Mengelola Data Riwayat Berobat Admin	60
Tabel 3.51 Tabel Object Perancangan Mengelola Data Riwayat Berobat	60
Tabel 3.52 Tabel Deskripsi Objek UI Melihat Data Riwayat Berobat	62
Tabel 3.53 Tabel Page Melihat Data Riwayat Berobat	62
Tabel 3.54 Tabel Object Perancangan Melihat Data Riwayat Berobat	63
Tabel 3.55 Tabel Deskripsi Objek UI	64
Tabel 3.56 Page Membuat Akun Instansi Kesehatan - MD - ADDIST	65
Tabel 3.57 Tabel Object Perancangan Membuat Akun Instansi Kesehatan	65
Tabel 3.58 Tabel Deskripsi Objek UI Mengelola User	67
Tabel 3.59 Tabel Page List Pasien	67
Tabel 3.60 Tabel Page Tambah Pasien	67
Tabel 3.61 Tabel Object Perancangan Mengelola User	68
Tabel 3.62 Tabel Kelas	72
Tabel 3.63 Tabel Requirement Traceability Matrix	79

## Daftar Gambar

Gambar 3.1 Usecase Diagram	16
Gambar 3.2 UI Register	18
Gambar 3.3 Sequence Diagram Register	20
Gambar 3.4 UI Login	21
Gambar 3.5 Sequence Diagram Login	22
Gambar 3.6 UI Dashboard	23
Gambar 3.7 Sequence Diagram Dashboard	25
Gambar 3.8 UI ListDokter	26
Gambar 3.9 Sequence Diagram ListDokter	28
Gambar 3.10 UI Detail Dokter	29
Gambar 3.11 Sequence Diagram Detail Dokter	31
Gambar 3.12 UI Mendaftar Berobat	32
Gambar 3.13 Sequence Diagram Mendaftar Berobat	33
Gambar 3.14 UI Riwayat Berobat	34
Gambar 3.15 Sequence Diagram Riwayat Berobat	35
Gambar 3.16 UI MD-EDT1 Gambar 3.17 UI MD-EDT2	36
Gambar 3.18 Sequence Diagram Edit Profile	37
Gambar 3.19 UI MD-CHK	38
Gambar 3.20 Sequence Diagram Pembayaran	39
Gambar 3.21 UI Ulasan	40
Gambar 3.22 Sequence Diagram Ulasan	42
Gambar 3.23 UI Onboarding	43
Gambar 3.24 Sequence Diagram Onboarding	44
Gambar 3.25 UI MD-INV	45
Gambar 3.26 Sequence Diagram Invoice	46
Gambar 3.27 UI DashBoard Instansi Kesehatan	47
Gambar 3.28 Sequence Dashboard Instansi Kesehatan	48
Gambar 3.29 UI MD-ADD_DCT	49
Gambar 3.30 Sequence Diagram Menambah List Dokter	50
Gambar 3.31 UI MD-VRF	51
Gambar 3.32 Sequence Diagram Verifikasi Antrian	52
Gambar 3.33 UI Dashboard Admin	53
Gambar 3.34 Sequence Diagram Dashboard Admin	54
Gambar 3.35 UI Riwayat Berobat	55
Gambar 3.36 Sequence Mengelola Riwayat Berobat Instansi Kesehatan	57
Gambar 3.37 UI MD-SHRB	58
Gambar 3.38 Sequence Diagram Melihat Data Riwayat Berobat	59
Gambar 3.39 UI MD-ADDIST	60
Gambar 3.40 Sequence Diagram Membuat Akun Instansi Kesehatan	61
Gambar 3.41 UI MD-EUSR1	62
Gambar 3.42 UI MD-EUSR2	63
Gambar 3.43 Sequence Diagram Edit Profile	64

Gambar 3.44 Diagram Class Keseluruhan	67
Gambar 3.45 ER-Diagram MyDoc	70

67
70

# 1. Pendahuluan

## 1.1 Tujuan Penulisan Dokumen

Tujuan dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) 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. Dengan dokumen ini akan dijelaskan secara teknik yang dimulai dari desain component diagram, class diagram, dan sequence diagram yang akan diimplementasikan untuk membangun sistem ini. Dokumen ini pun digunakan sebagai acuan dalam pembangunan perangkat lunak dan evaluasi pada akhir pembangunan perangkat lunak.

## 1.2 Lingkup Masalah

MyDoc adalah aplikasi berbasis mobile yang berupa layanan kesehatan, yaitu sistem pendaftaran antrian berobat/konsultasi. Aplikasi MyDoc bertujuan untuk mempermudah pendaftaran berobat/konsultasi 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.

## 1.3 Definisi dan Istilah

Pada dokumen ini terdapat terdapat beberapa istilah yang sukar dimengerti bagi orang awam, maka dari itu dibuatlah tabel definisi dan istilah tersebut yang akan dijelaskan pada tabel berikut.

Definisi dan Istilah	Deskripsi
Register	proses penyimpanan data akun baru terhadap sistem
Login	Proses yang digunakan untuk masuk melalui data akun yang telah terdaftar ke sistem
User	seseorang yang menggunakan aplikasi
Mobile	sebuah benda yang berteknologi tinggi dan dapat bergerak tanpa menggunakan kabel
Client	Pengguna Produk Aplikasi
Maintenance	kegiatan untuk memonitor dan memelihara fasilitas dengan merancang, mengatur, menangani, dan memeriksa aplikasi
Title	Nama yang dipakai untuk sebuah judul
Entity	sesuatu yang memiliki keberadaan yang unik dan berbeda, walaupun tidak harus dalam bentuk fisik
Efisien	melakukan pekerjaan dengan tepat dan mampu

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 15 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

	menjalankan tugas dengan cermat, dan berdaya guna.
spesifikasi	proses, cara, perbuatan melakukan pemilihan (perincian)
Navigasi	proses mengendalikan arahan fitur
Verifikasi	pemeriksaan tentang kebenaran
Autentikasi	proses identifikasi pengguna
Index	Proses menghasilkan nilai atau referensi ke nilai dari dalam tabel atau rentang
Submit	Proses pengajuan atau kirim

**Tabel 1.1** Definisi dan Istilah

#### **1.4 Referensi**

- Slides Perkuliahan RPL: IMPLEMENTASI DAN PENGUJIAN PL
- SRS Document “Centralized Monitoring of Professional Drivers”
- DPPL Document “Halliu”



## 2 Perancangan Global

### 2.1 Rencana Lingkungan Implementasi

Untuk client, perangkat lunak ini dapat beroperasi pada OS Android 5+ keatas. Perangkat keras pada sisi client wajib terhubung dengan jaringan internet.

Aplikasi ini akan berfungsi pada web server dengan spesifikasi minimum :

- Server : Ubuntu Server
- Processor : 1 CPU
- RAM : 1 GB
- DBMS : Mysql

Perangkat lunak ini juga membutuhkan adanya maintenance pada sisi server dan client yang mengurus :  
Frontend Development : Bootstrap v4, JQuery, Android Native menggunakan Java  
Backend Development : Nginx, Codeigniter V4

### 2.2 Deskripsi Arsitektur Perangkat Lunak

Diisi dengan gambar **komponen diagram** atau **daftar modul** , dapat dituliskan dalam bentuk tabel berikut:

No	Nama Komponen	Keterangan
1	Login	Laman untuk masuk ke aplikasi Mydoc
2	Register	Laman Registrasi akun Mydoc agar bisa melakukan login
3	Dashboard	Menu Beranda Aplikasi Mydoc menampilkan informasi fitur yang ada di MyDoc
4	List Dokter	Menu yang ditujukan untuk mendaftar berobat berisi informasi list dokter yang dapat dipilih sesuai keinginan pasien
5	Riwayat Berobat	Menu yang berisi informasi riwayat berobat pasien
6	Beri Ulasan	sebuah fitur yang digunakan untuk memberi ulasan kepada dokter setelah selesai berobat
7	Onboarding	sebuah laman yang menampilkan overlay introduction page beserta pilihan menu register dan login
8	Payment Gateway	fitur yang dimiliki sistem MyDoc untuk melakukan transaksi pembayaran

9	Mendaftar Berobat	fitur unggulan MyDoc untuk mendaftar berobat setelah memilih List Dokter
10	Edit Profile	Fitur yang digunakan pasien untuk mengubah informasi terkait akun profil MyDoc di laman profile

**Tabel 2.1** Deskripsi Arsitektur Perangkat Lunak

### 3 Perancangan Rinci

Berdasarkan hasil analisa yang sudah dibuat, dibentuklah Use Case diagram yang digunakan untuk memberi gambaran secara general terhadap sistem yang dibuat. Selain itu akan memberikan gambaran hal yang harus dibuat berdasarkan Class Diagram Analisa dan Diagram Komponen.



**Gambar 3.1** Usecase Diagram

#### 3.1 Realisasi Use Case

Berisi TABEL USE CASE sebagai berikut :

No	Nama Use Case	Deskripsi Use Case
#1	Register	Mendaftarkan akun baru terhadap sistem
#2	Login	Masuk ke dalam aplikasi dengan akun yang sudah didaftarkan

#3	Dashboard	Memberikan navigation kepada user dan sekilas info data pasien (Saldo, ulasan yang diberikan, riwayat pendaftaran, dll)
#4	List Dokter	Melihat list rumah sakit dan dokter yang tersedia untuk membantu proses pendaftaran berobat
#5	Detail Dokter	Menampilkan detail dari dokter dan juga jadwal praktik dokter
#6	Mendaftar Berobat	Mendaftar berobat di rumah sakit dan dokter yang dituju
#7	Riwayat Berobat	Melihat informasi mengenai data riwayat berobat pasien
#8	Edit Profile	Menampilkan informasi detail akun user/pasien dan dapat memodifikasi data tersebut
#9	Pembayaran	Menyelesaikan transaksi pendaftaran berobat dengan membayar sesuai yang nominal yang ditampilkan
#10	Memberikan Ulasan	Mengulas hasil pengobatan agar dokter/instansi kesehatan dapat menjadikannya sebagai masukan
#11	OnBoarding	Memberikan navigasi kepada pasien untuk memilih masuk atau mendaftar
#12	Invoice	Memberikan rincian data pesanan
#13	Dashboard Instansi Kesehatan	Memberikan navigation kepada admin dan sekilas info data instansi kesehatan(jumlah berobat, total dokter, jumlah pasien)
#14	Menambah List Dokter	Menambahkan Dokter pada suatu instansi kesehatan
#15	Verifikasi Antrian	Memverifikasi antrian berobat pasien
#16	Dashboard Admin	Memberikan navigation kepada admin dan sekilas info data mydoc Jumlah berobat, jumlah pasien, total instansi, dan total dokter)
#17	Mengelola data Riwayat Berobat	Melihat seluruh data riwayat berobat role admin
#18	Melihat data riwayat berobat	melihat data riwayat berobat dengan role instansi kesehatan
#19	Membuat Akun Instansi Kesehatan	membuat akun instansi kesehatan baru yang bekerjasama dengan Mydoc dilakukan oleh admin
#20	Mengelola User	mengelola user pasien dengan role admin yang berisi fitur mengedit, tambah, dan delete data user.

**Tabel 3.1** Tabel Use Case

### 3.1.1 Use Case #1 Register

#### 3.1.1.1 Use Case Scenario #1 Register

Skenario Use Case #1 :

- i. Pre-Condition
  - **RegisterAct** menampilkan UI
- ii. Use Case Description
  - a. Primary Flow
    1. User memasukkan form et\_fullname, et\_email, et\_birthday, et\_address, et\_password1, dan et\_password2
    2. User menekan tombol daftar [btn\_register]
    3. Sistem memanggil fungsi FieldValidator.isNull() dan register()
    4. Sistem memberi notifikasi bahwa user berhasil didaftarkan
  - b. Alternative Flow
    - E-1 : User mengosongkan form
      - Sistem memanggil fungsi showError() untuk memberikan notifikasi bahwa formulir tidak valid
      - Kembali ke primary flow ke-1
    - E-2 : Email sudah terdaftar
      - Sistem memanggil fungsi showError() untuk memberikan notifikasi bahwa email yang digunakan sudah terdaftar pada sistem
      - Kembali ke primary flow ke-1
- iii. Post-Condition
  - Akun berhasil dibuat dan tersimpan kedalam database
  - Sistem me-redirect ke halaman OnBoarding

#### 3.1.1.2 UI Design dan Deskripsi Objek UI #1 Register

< **Daftar**

Daftar Menjadi User **MyDoc!**

Nama Lengkap **et\_fullname**

E-mail **et\_email**

Tanggal Lahir **et\_birthday**

Alamat **et\_address**

Kata sandi **et\_password1**

Konfirmasi Kata sandi **et\_password2**

**Daftar**

**btn\_register**

**Gambar 3.2** UI Register

Tabel Deskripsi Objek UI

<b>ID. LAYAR</b>	<b>NAMA LAYAR</b>	<b>DESKRIPSI</b>
MD-REG	Register Page	

**Tabel 3.2** Tabel Deskripsi Objek UI Register

Page Register - MD-REG

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
txt_appbar_title	TextView	Daftar	Title untuk appbar
et_fullname	EditText	Nama Lengkap	Field Form untuk nama lengkap sebagai data registrasi
et_email	EditText	E-mail	Field Form untuk email sebagai data registrasi
et_birthday	EditText	Tanggal Lahir	Field Form untuk tanggal lahir sebagai data registrasi
et_address	EditText	Alamat	Field Form untuk alamat sebagai data registrasi
et_password1	EditText	Kata Sandi	Field Form untuk password sebagai data registrasi
et_password2	EditText	Konfirmasi Kata Sandi	Field Form untuk melihat kesamaan password
btn_register	Button	Daftar	Jika di klik akan memanggil function register()

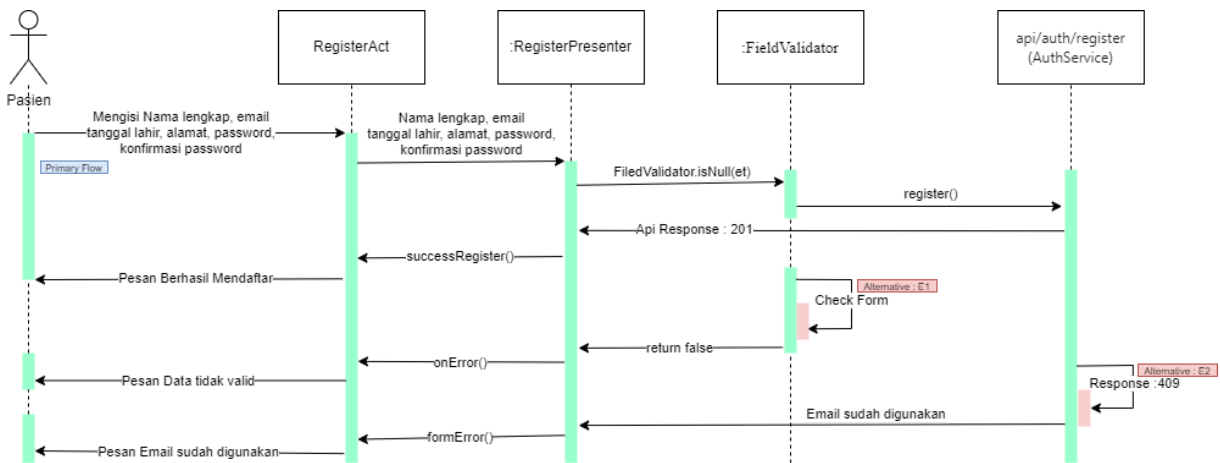
**Tabel 3.3** Page Register - MD - REG**3.1.1.3 Identifikasi Object dan Tipe nya #1 Register**

TABEL OBJECT PERANCANGAN

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1.	RegisterAct	Boundary
2.	RegisterPresenter	Controller
3.	RegisterModel	Entity
3.	AuthService	Entity
4.	FieldValidator	Controller

**Tabel 3.4** Tabel Object Perancangan Register

3.1.1.4 Sequence Diagram #1 Register



Gambar 3.3 Sequence Diagram Register

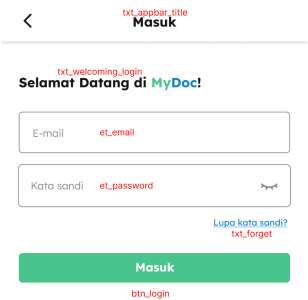
3.1.2 Use Case #2 Login

3.1.2.1 Use Case Scenario #2 Login

Skenario Use Case #2 :

- i. Pre-Condition
  - **LoginAct** menampilkan UI
- ii. Use Case Description
  - a. Primary Flow
    1. User memasukkan form et\_email dan et\_password
    2. User menekan tombol masuk[btn\_login]
    3. Sistem memanggil fungsi FieldValidator.isNull() dan login()
    4. User berhasil login dan diarahkan ke dashboard
  - b. Alternative Flow
    - E-1 : User mengosongkan form
      - Sistem memanggil fungsi showError() untuk memberikan notifikasi bahwa formulir tidak valid
      - Kembali ke primary flow ke-1
    - E-2 : Email atau password tidak sesuai
      1. Sistem memanggil fungsi showError() untuk memberi notifikasi bahwa email atau password tidak sesuai
      2. Kembali ke primary flow ke-1
- iii. Post-Condition
  - Sistem menampilkan dashboard

3.1.2.2 UI Design dan Deskripsi Objek UI #2 Login



Gambar 3.4 UI Login

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-LGN	Page Login	Untuk menampilkan halaman Login

Tabel 3.5 Tabel Deskripsi Objek UI Login

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 24 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		



Id_Objek	JENIS	LABEL*	Keterangan**
txt_appbar_title	TextView	Daftar	Title untuk appbar
et_email	EditText	E-Mail	Field Form untuk email sebagai data login
et_password	EditText	Password	Field Form untuk password sebagai data login
txt_forget	TextView	Lupa kata sandi?	Jika diklik akan menjalankan fungsi redirect ke halaman forget password
btn_login	Button	Masuk	Jika di klik akan memanggil function login()

Tabel 3.6 MD-LGN - Page Login

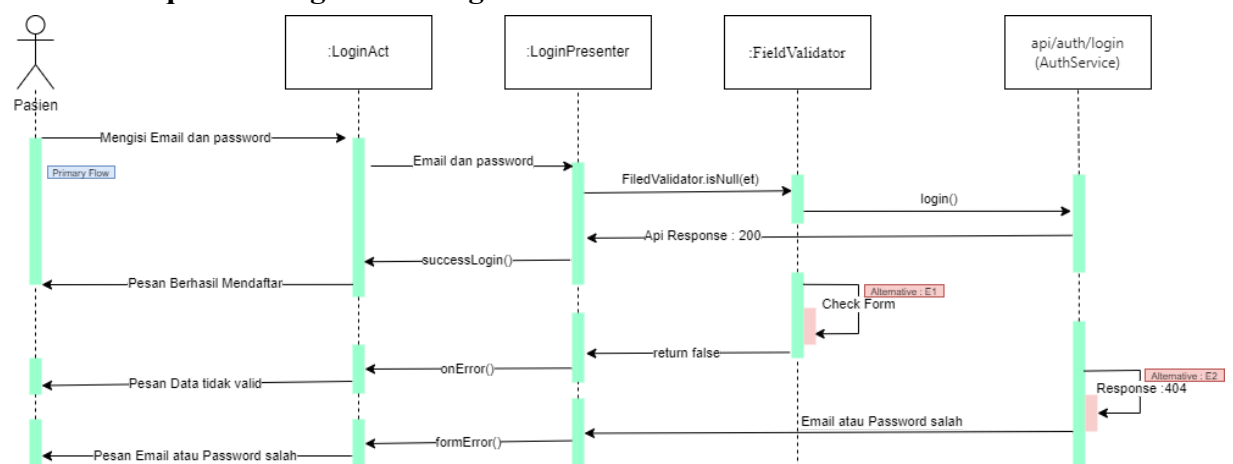
### 3.1.2.3 Identifikasi Object Baru & Tipe Kelas #2 Login

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1.	LoginAct	Boundary
2.	LoginPresenter	Controller
3.	LoginModel	Entity
4.	FieldValidator	Controller
5.	AuthService	Entity

Tabel 3.7 Tabel Object Perancangan Login

### 3.1.2.4 Sequence Diagram #2 Login



Gambar 3.5 Sequence Diagram Login

### 3.1.3 Use Case #3 Dashboard

#### 3.1.3.1 Use Case Scenario #3 Dashboard

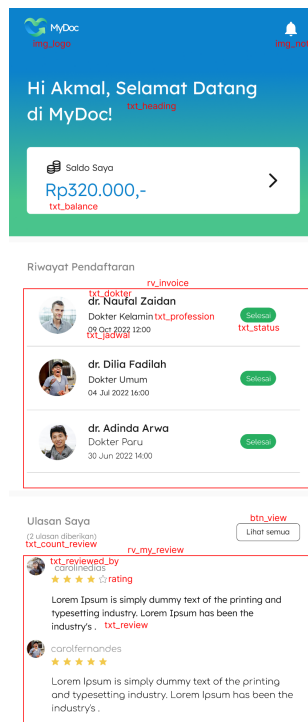
Skenario Use Case #3 :

- i. Pre-Condition
  - User sudah melakukan #2 Login
- ii. Use Case Description
  - a. Primary Flow
    1. Sistem memanggil fungsi getDashboard()
    2. Dashboard pasien tertampil
  - b. Alternative Flow
 

E-1:

    1. Sistem memanggil fungsi showError() untuk memberi notifikasi error.
    2. Kembali ke primary flow ke-1
- iii. Post-Condition
  - Dashboard tertampil

#### 3.1.3.2 UI Design dan Deskripsi Objek UI #3 Dashboard



**Gambar 3.6** UI Dashboard

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-DSB	Page Dashboard	Untuk menampilkan halaman utama aplikasi

**Tabel 3.8** Tabel Deskripsi Objek UI Dashboard

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 26 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
img_logo	Image		Logo Aplikasi
img_notif	Image		Untuk menampilkan pemberitahuan yang masuk
txt_heading	TextView	Selamat Datang di MyDoc	Title untuk appbar
txt_balance	TextView	Jumlah Saldo	Menampilkan saldo tersisa
btn_detail	Button	Detail Saldo	Untuk menampilkan detail saldo user
rv_invoice	Recyclervi ew		Menampilkan list invoice
img_dokter	image		Menampilkan profile dokter
txt_dokter	TextView	Nama dokter	Menampilkan nama dokter
txt_profession	TextView	Profesi Dokter	Menampilkan spesialis dokter
txt_jadwal	TextView	Tanggal dan jam	Menampilkan tanggal dan jadwal berobat
txt_status	Label	Selesai	Menandakan riwayat sudah selesai temu dokter
txt_count_review	TextView	Jumlah Ulasan	Menampilkan text
btn_view	Button	Lihat Semua	Untuk menampilkan semua ulasan
rv_my_review	Recyclervi ew		Menampilkan list review
rating	RatingBar		Rating bar berbentuk bintang
navbar	Navbar		Untuk menavigasi user

Tabel 3.9 MD-DSB - Page Dashboard

### 3.1.3.3 Identifikasi Object Baru & Tipe Kelas #3 Dashboard

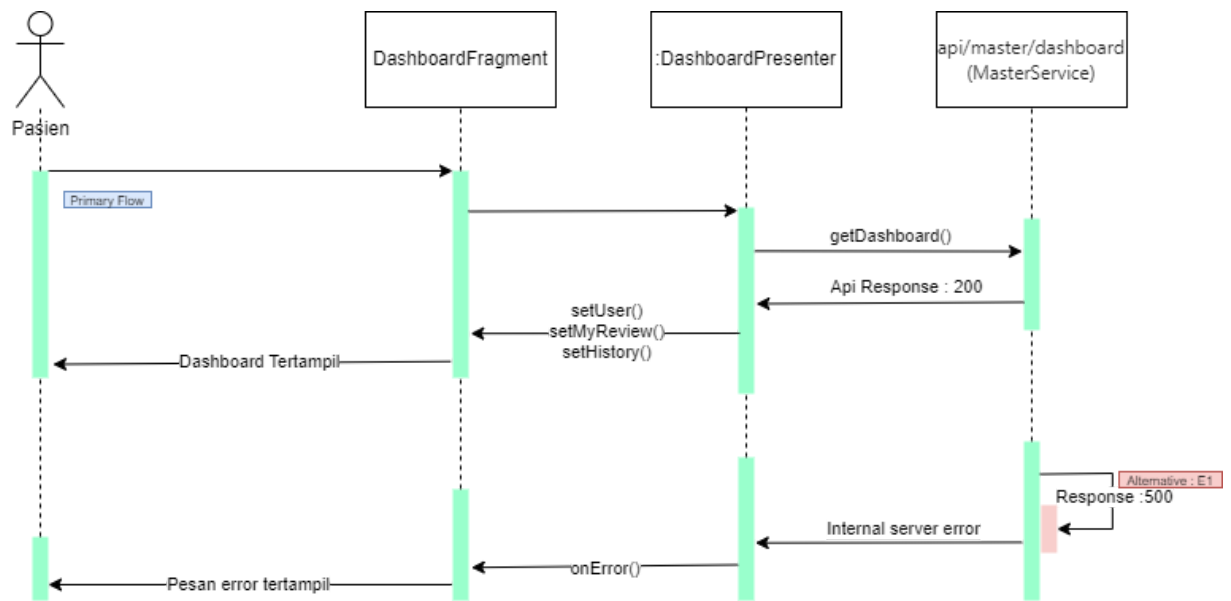
TABEL OBJECT PERANCANGAN

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1.	DashboardAct	Boundary
2.	DashboardPresenter	Controller
3.	DashboardModel	Entity
4.	MasterService	Entity

Tabel 3.10 Tabel Object Perancangan Dashboard

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 27 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

### 3.1.3.4 Sequence Diagram #3 Dashboard



**Gambar 3.7** Sequence Diagram Dashboard

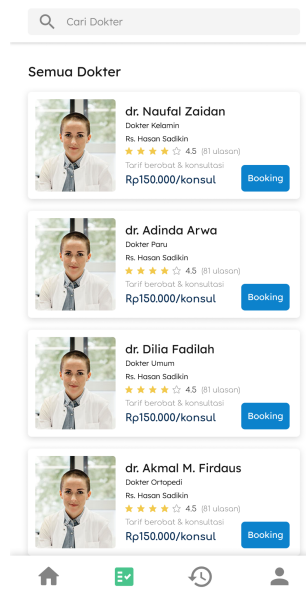
### 3.1.4 Use Case #4 List Dokter

#### 3.1.4.1 Use Case Scenario #4 List Dokter

Skenario Use Case #4 :

- i. Pre-Condition
  - User sudah melakukan #2 Login
  - User menekan menu **List Dokter**
- ii. Use Case Description
  - a. Primary Flow
    1. Sistem memanggil fungsi getListDokter()
    2. List dokter tertampil
  - b. Alternative Flow
    - A-1 :
      1. User mencari dokter tertentu
      2. Dokter yang dicari tertampil
- iii. Post-Condition
  - Informasi dan jadwal praktik yang cari tertampil di layar user

#### 3.1.4.2 UI Design dan Deskripsi Objek UI #4 List Dokter



Gambar 3.8 UI List Dokter

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-LDR	Page List Dokter	Untuk menampilkan list dokter yang tersedia

Tabel 3.11 Tabel Deskripsi Objek UI List Dokter

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 29 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

#### Page List Dokter

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
et_search	EditText	Cari Dokter	Untuk mencari nama dokter yang tersedia
txt_appbartitle	TextView	Semua Dokter	Title untuk appbar
img_dokter	EditImage		Menampilkan gambar dokter
txt_dokter	TextView	Nama Dokter	Menampilkan nama dokter
txt_profession	TextView	Profesi Dokter	Menampilkan spesialis dokter
txt_lokasi	TextView	Nama Rumah Sakit	Menampilkan nama rumah sakit
rating	RatingBar		Untuk menampilkan kepuasan para pasien
txt_rating	TextView	Nilai	Menampilkan akumulasi rate para dokter
txt_ulasan	TextView	Jumlah ulasan	Menampilkan jumlah orang yang telah memberi ulasan
txt_price	TextView	Tarif Konsul	Menampilkan tarif konsul
btn_booking	Button	Booking	Tombol untuk memilih konsul kepada dokter yang diinginkan
navbar	navbar		Untuk menavigasi user

**Tabel 3.12** Page List Dokter

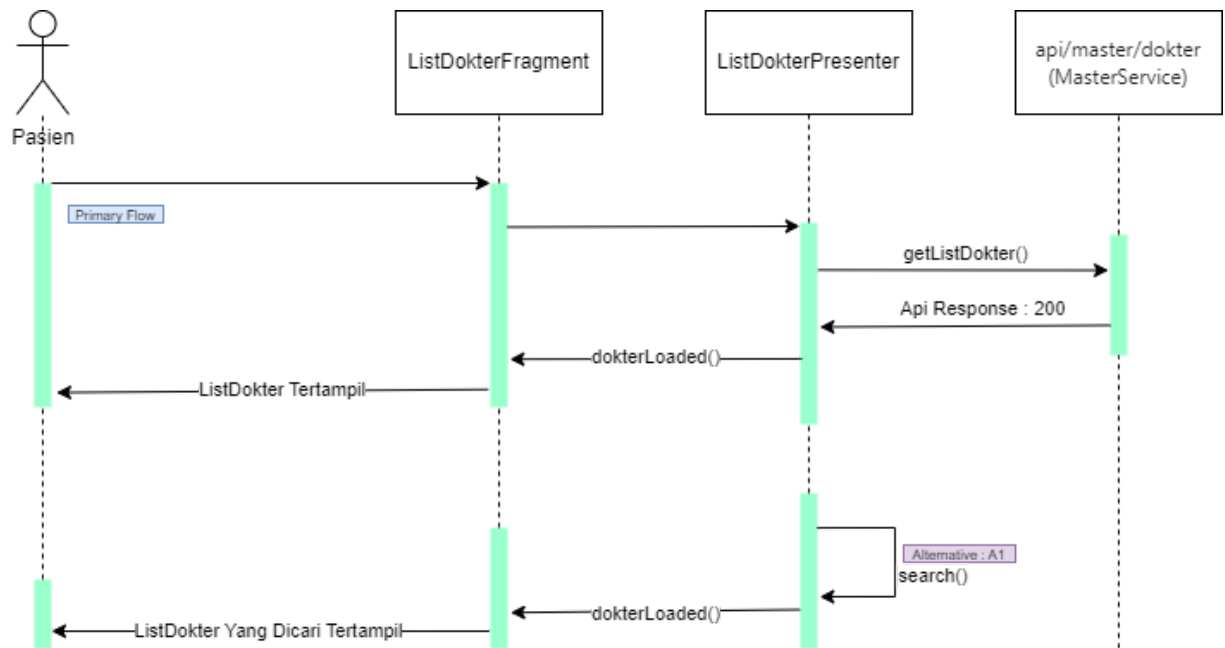
#### 3.1.4.3 Identifikasi Object Baru & Tipe Kelas #4 List Dokter

##### TABEL OBJECT PERANCANGAN

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1.	ListDokterAct	Boundary
2.	ListDokterPresenter	Controller
3.	ListDokterModel	Entity
5.	MasterService	Entity

**Tabel 3.13** Tabel Object Perancangan List Dokter

### 3.1.4.4 Sequence Diagram #4 List Dokter



**Gambar 3.9** Sequence Diagram ListDokter

### 3.1.5 Use Case #5 Detail Dokter

#### 3.1.5.1 Use Case Scenario #5 Detail Dokter

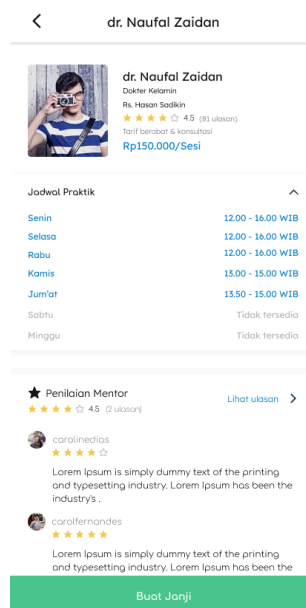
Skenario Use Case #5 Detail Dokter :

- i. Pre-Condition
  - User sudah melakukan #2 Login
  - User menekan dokter yang tersedia
- ii. Use Case Description
  - a. Primary Flow
    1. Sistem memanggil fungsi `getDokter(int id)`
    2. Sistem memanggil fungsi `loadDetailDokter()`
    3. Detail dokter ditampilkan
  - b. Alternative Flow
 

E-1:

    1. Sistem memanggil fungsi `showError()` untuk memberi notifikasi error.
    2. Notifikasi “Dokter tidak tersedia” ditampilkan
- iii. Post-Condition
  - Detail dokter ditampilkan dilayar user

#### 3.1.5.2 UI Design dan Deskripsi Objek UI #5 Detail dokter



**Gambar 3.10** UI Detail Dokter

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-DTL	Detail Dokter	Menampilkan Detail Dokter

**Tabel 3.14** Tabel Deskripsi Objek UI Detail Dokter

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 32 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		



## MD-DTL - Page Detail Dokter

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
txt_appbar_title	TextView	Nama Detail Dokter	Title untuk appbar
txt_doktername	TextView	Nama Dokter	Untuk mengetahui nama dokter yang akan dikunjungi
img_Dokter	Image		Menampilkan Image Dokter
txt_profesi	TextView	Dokter Spesialis	Untuk menampilkan profesi dokter
txt_hospital	TextView	Lokasi Rumah Sakit	Untuk mengetahui lokasi rumah sakit
ir_rating	RatingBar		Hasil inputan dari user berupa rating bintang 1 sampai 5
txt_tarif	TextView		Untuk mengetahui Tarif dokter
txt_jadwal	TextView		
txt_hari	TextView	Jadwal praktek	Untuk mengetahui har jadwal dokter
txt_jam	TextView	jadwal prakter	Untuk mengetahui Jam Jadwal dokter
txt_penilaian	TextView		Untuk menegetahui penilaian user
btn_view	Button	Lihat Ulasan	Untuk menampilkan semua ulasan
btn_buatjanji	Button	Buat Janji	jika diklik akan memanggil function janji()

Tabel 3.15 MD-DTL - Page Detail Dokter

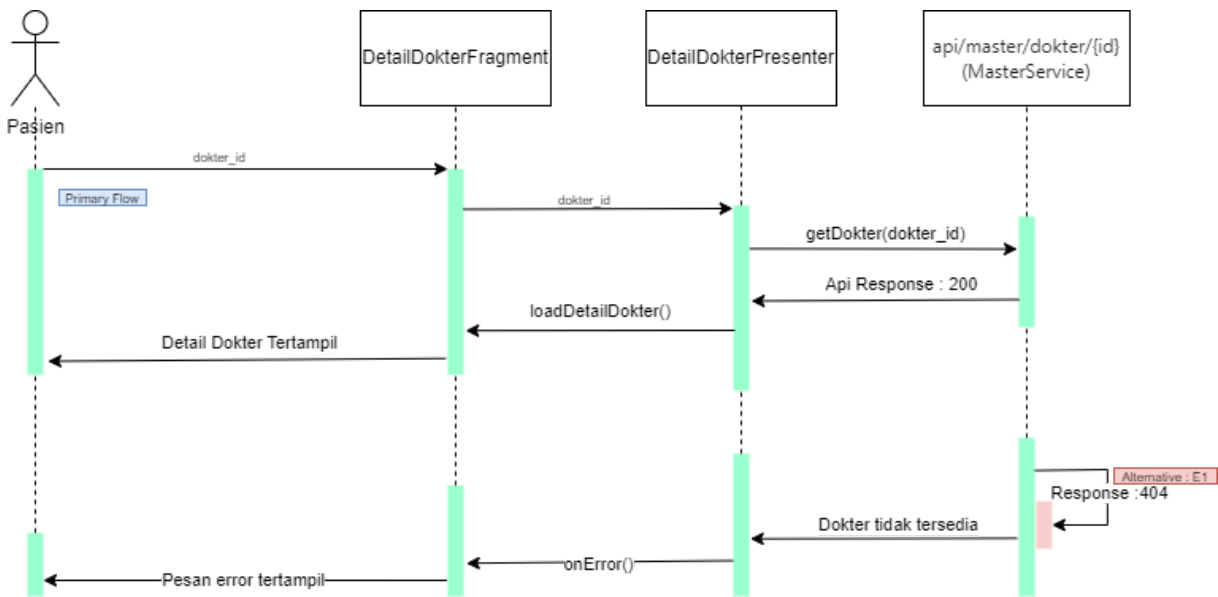
## 3.1.5.3 Identifikasi Object Baru &amp; Tipe Kelas #5 Detail Dokter

## TABEL OBJECT PERANCANGAN

<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
DetailDokterFragment	Boundary
DetailDokterPresenter	Controllor
DetailDokterModel	Entity
MasterService	Entity

Tabel 3.16 Tabel Object Perancangan Detail Dokter

3.1.5.4 Sequence Diagram #5 Detail Dokter



Gambar 3.11 Sequence Diagram Detail Dokter

### 3.1.6 Use Case #6 Mendaftar Berobat

#### 3.1.6.1 Use Case Scenario #6 Mendaftar Berobat

Skenario Use Case #6 :

- i. Pre-Condition
  - User sudah melakukan #2 Login
- ii. Use Case Description
  - a. Primary Flow
    1. User memilih metode pembayaran pada [container\_payment\_gateway]
    2. Lalu user menekan tombol [btn\_order]
    3. Sistem akan memanggil fungsi checkout()
    4. Sistem mengembalikan data lewat function successCheckout()
    5. Pesan sukses tampil
- iii. Post-Condition
  - Berhasil melakukan pembayaran

#### 3.1.6.2 UI Design dan Deskripsi Objek UI #6 Mendaftar Berobat

The screenshot shows a mobile application interface for ordering a medical consultation. At the top, there's a back arrow and the title 'Pemesanan'. Below this, the 'Informasi Pesanan' (Order Information) section displays a doctor's profile: Dr. Akmal Muhamad Firdaus, a Dermatologist (Dokter Kulit) at RS. Hasan Sadikin, with a consultation schedule for October 9, 2022. The 'Rincian Harga' (Price Details) section shows a price of Rp150.000,- for the consultation. Below that, the 'Metode Pembayaran' (Payment Method) section has a dropdown menu labeled 'Pilih metode pembayaran'. At the bottom, the 'Total' amount is Rp150.000,-, and there is a blue button labeled 'Buat Pesanan'.

**Gambar 3.12** UI Mendaftar Berobat

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-CHK	Layout Pembayaran	Untuk melakukan proses checkout dengan dokter

**Tabel 3.17** Tabel Deskripsi Objek UI Mendaftar Berobat

Page Mendaftar Berobat

Id_Objek	JENIS	LABEL*	Keterangan**
Img_dokter	ImageView		Menampilkan gambar dokter
txt_dokter	TextView	Nama dokter	Menampilkan nama dokter

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 35 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
txt_profession	TextView	Profesi Dokter	Menampilkan profesi dokter
txt_instansi	TextView	Nama instansi	Menampilkan nama instansi dokter tersebut praktik
txt_schedule	TextView	Tanggal berobat	Menampilkan tanggal dan waktu berobat
txt_price	TextView	Rpxxx.xxx	Menampilkan harga pendaftaran
container_payment_gateway	Constraint Layout		Ketika di klik akan memanggil function navigate() untuk menampilkan list pembayaran dari payment gateway
btn_order	Button	Buat Pesanan	Ketika di klik akan memanggil function checkout()

**Tabel 3.18** Page Mendaftar Berobat

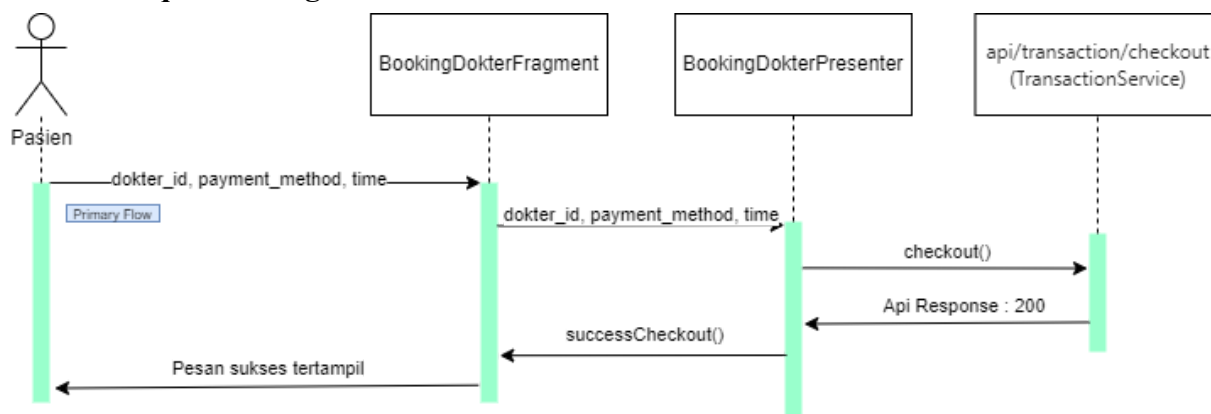
### 3.1.6.3 Identifikasi Object Baru & Tipe Kelas #6 Mendaftar Berobat

**TABEL OBJECT PERANCANGAN**

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1	BookingDokterFragment	Boundary
2	BookingDokterPresenter	Controllor
3	TransactionService	Entity

**Tabel 3.19** Tabel Object Perancangan Mendaftar Berobat

### 3.1.6.4 Sequence Diagram #6 Mendaftar Berobat



**Gambar 3.13** Sequence Diagram Mendaftar Berobat

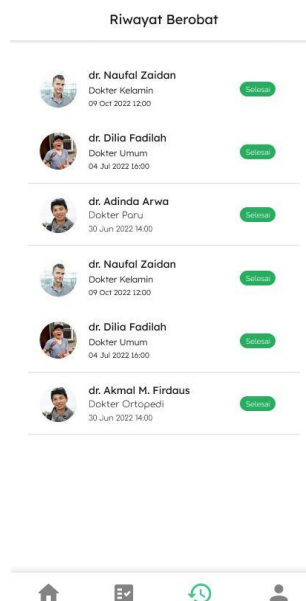
### 3.1.7 Use Case #7 Riwayat Berobat

#### 3.1.7.1 Use Case Scenario #7 Riwayat Berobat

Skenario Use Case #7 :

- i. Pre-Condition
  - User sudah melakukan #2 Login
- ii. Use Case Description
  - a. Primary Flow
    6. Sistem memanggil fungsi getInvoice()
    7. Sistem memanggil fungsi loadHistoryBerobat()
    8. Riwayat berobat tertampil
- iii. Post-Condition
  - List riwayat berobat tertampil

#### 3.1.7.2 UI Design dan Deskripsi Objek UI #7 Riwayat berobat



Gambar 3.14 UI Riwayat Berobat

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-RWB	Riwayat Berobat	Menampilkan list Riwayat Berobat

Tabel 3.20 Tabel Deskripsi Objek UI Riwayat Berobat

Page Riwayat Berobat

Id_Objek	JENIS	LABEL*	Keterangan**
txt_appbar_title	TextView	Riwayat Berobat	Title untuk appbar

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 37 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
txt_doktername	TextView	Nama Dokter	Untuk mengetahui nama dokter yang pernah user kunjungi
txt_profesi	TextView	Dokter Spesialis	Untuk menampilkan profesi dokter
txt_date	TextView	Tanggal berobat	Untuk mengetahui tanggal berobat
txt_status	TextView	success/process/failed	menampilkan status riwayat berobat(success/process/failed )
navbar	navigation bar		untuk menavigasi user

**Tabel 3.21** Page Riwayat Berobat

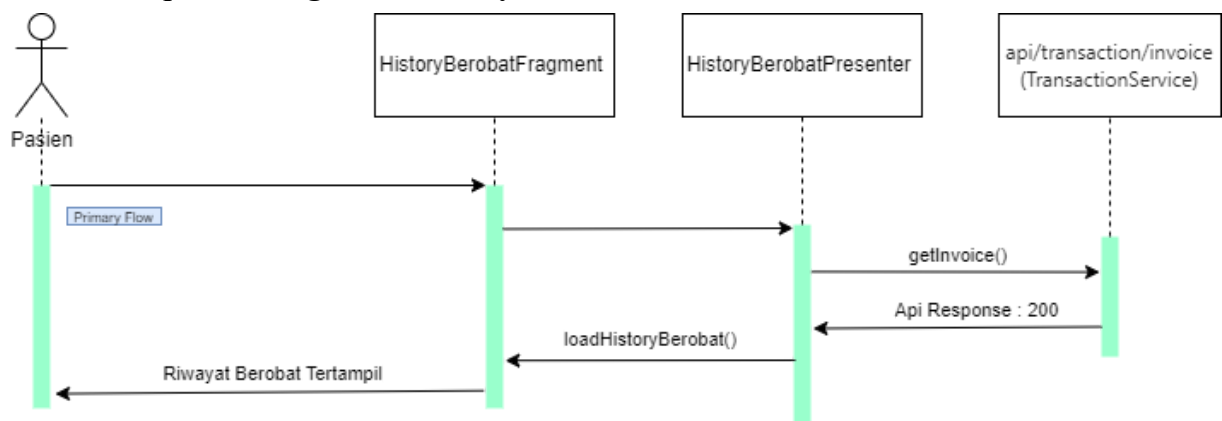
### 3.1.7.3 Identifikasi Object Baru & Tipe Kelas #7 Riwayat Berobat

**TABEL OBJECT PERANCANGAN**

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1.	RiwayatBerobatACT	Boundary
2.	RiwayatBerobatPresenter	Controller
3.	RiwayatBerobatModel	Entity
4.	MasterService	Entity

**Tabel 3.22** Tabel Object Perancangan Riwayat Berobat

### 3.1.7.4 Sequence Diagram #7 Riwayat Berobat



**Gambar 3.15** Sequence Diagram Riwayat Berobat

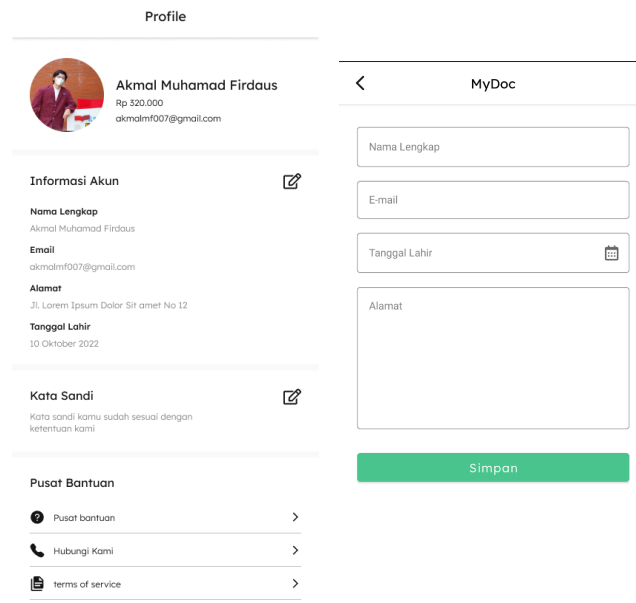
### 3.1.8 Use Case #8 Edit Profile

#### 3.1.8.1 Use Case Scenario #8 Edit Profile

Skenario Use Case #8 :

- i. Pre-Condition
  - User sudah melakukan #2 Login
- ii. Use Case Description
  - a. Primary Flow
    1. User mengganti data yang ingin diganti
    2. User menekan [btn\_simpan]
    3. Sistem memanggil fungsi editProfile()
    4. Sistem mengembalikan data lewat function successUpdate()
    5. Menampilkan pesan sukses
- iii. Post-Condition
  - Kembali ke page MD-EDT1
  - Data berhasil diubah

#### 3.1.8.2 UI Design dan Deskripsi Objek UI #8 Edit Profile



**Gambar 3.16** UI MD-EDT1 **Gambar 3.17** UI MD-EDT2

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-EDT1	Profile	Menampilkan data profile user
MD-EDT2	Edit Profile	Layout untuk mengedit profile

**Tabel 3.23** Tabel Deskripsi Objek UI Edit Profile

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 39 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

Id_Objek	JENIS	LABEL*	Keterangan**
txt appbar title	TextView	Edit Profile	Title untuk appbar
et fullname	EditText	Nama Lengkap	Field untuk mengubah nama
et email	EditText	Email	Field untuk mengubah email
et birthday	EditText	Tanggal Lahir	Field untuk mengubah tanggal lahir
et address	EditText	Address	Field untuk mengubah address
btn simpan	Button	Simpan	Ketika diklik akan memanggil fungsi editProfile()

Tabel 3.24 Page Edit Profile - MD - EDT2

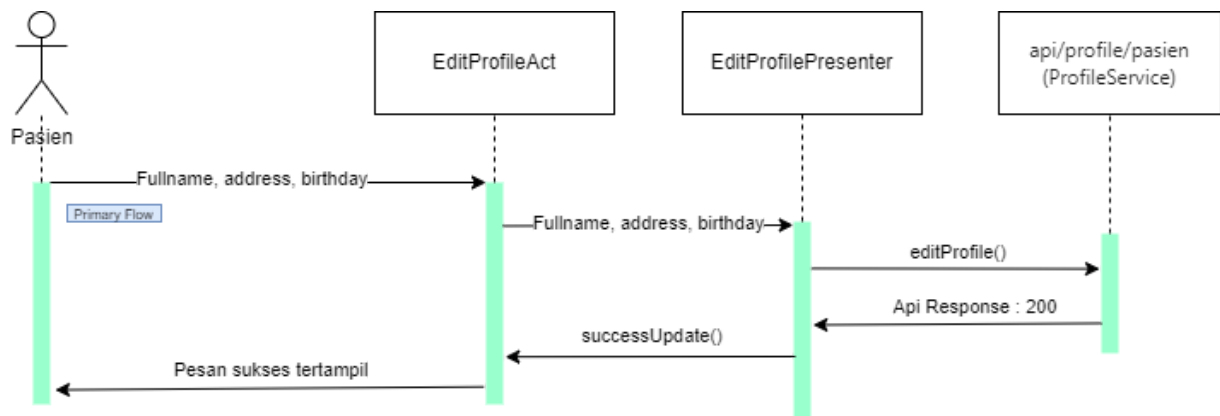
### 3.1.8.3 Identifikasi Object Baru & Tipe Kelas #8 Edit Profile

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1.	EditProfileAct	Boundary
2.	EditProfilePresenter	Controller
3.	EditProfileModel	Entity
4.	ProfileService	Entity

Tabel 3.25 Tabel Object Perancangan Edit Profile

### 3.1.8.4 Sequence Diagram #8 Edit Profile



Gambar 3.18 Sequence Diagram Edit Profile



### 3.1.9 Use Case #9 Pembayaran

#### 3.1.9.1 Use Case Scenario #9 Pembayaran

Skenario Use Case #9 :

- i. Pre-Condition
  - User sudah melakukan #2 Login
- ii. Use Case Description
  - a. Primary Flow
    1. User memilih metode pembayaran pada [container\_payment\_gateway]
    2. Lalu user menekan tombol [btn\_order]
    3. Sistem akan memanggil fungsi checkout()
    4. Sistem mengembalikan data lewat function successCheckout()
    5. Pesan sukses tampil
- iii. Post-Condition
  - Berhasil melakukan pembayaran

#### 3.1.9.2 UI Design dan Deskripsi Objek UI #9 Pembayaran

The screenshot shows a mobile application interface for a payment screen titled 'Pemesanan'. It includes a back arrow at the top left. The main content is divided into three sections: 'Informasi Pesanan' (Order Information) displaying a doctor's profile (Dr. Akmal Muhamad Firdaus, Dermatologist, at RS. Hasan Sadikin on Oct 9, 2022), 'Rincian Harga' (Price Details) showing a consultation fee of Rp150,000 and a total of Rp150,000, and 'Metode Pembayaran' (Payment Method) with a dropdown menu. At the bottom, there is a 'Total' of Rp150,000 and a blue 'Buat Pesanan' (Create Order) button.

**Gambar 3.19** UI MD-CHK

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-CHK	Layout Pembayaran	Untuk melakukan proses checkout dengan dokter

**Tabel 3.26** Tabel Deskripsi Objek UI Pembayaran

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 41 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

### Pembayaran – MD-CHK

Id_Objek	JENIS	LABEL*	Keterangan**
Img dokter	ImageView		Menampilkan gambar dokter
txt dokter	TextView	Nama dokter	Menampilkan nama dokter
txt profession	TextView	Profesi Dokter	Menampilkan profesi dokter
txt instansi	TextView	Nama instansi	Menampilkan nama instansi dokter tersebut praktik
txt schedule	TextView	Tanggal berobat	Menampilkan tanggal dan waktu berobat
txt price	TextView	Rpxxx.xxx	Menampilkan harga pendaftaran
container_paym ent_gateway	Constraint Layout		Ketika di klik akan memanggil function navigate() untuk menampilkan list pembayaran dari payment gateway
btn_order	Button	Buat Pesanan	Ketika di klik akan memanggil function checkout()

**Tabel 3.27** Pembayaran - MD - CHK

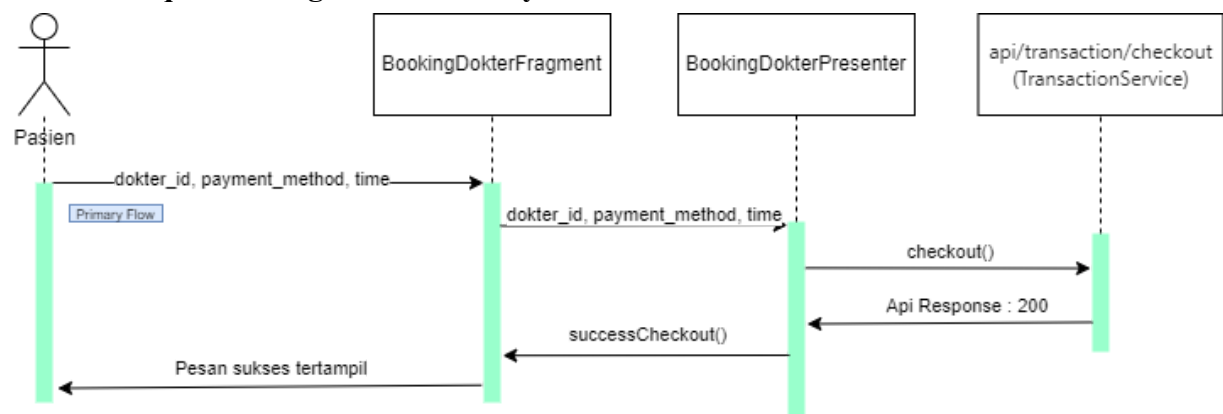
### 3.1.9.3 Identifikasi Object Baru & Tipe Kelas #9 Pembayaran

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	BookingDokterFragment	Boundary
2	BookingDokterPresenter	Controlller
4	TransactionService	Entity

**Tabel 3.28** Tabel Object Perancangan Pembayaran

### 3.1.9.4 Sequence Diagram #9 Pembayaran



**Gambar 3.20** Sequence Diagram Pembayaran

### 3.1.10 Use Case #10 Ulasan

#### 3.1.10.1 Use Case Scenario #10 Ulasan

Skenario Use Case #2 :

- i. Pre-Condition
  - User sudah menyelesaikan #6 pendaftaran berobat
- ii. Use Case Description
  - a. Primary Flow
    1. User diarahkan ke UlasanAct
    2. User mengisi nilai dokter [rating]
    3. User menekan tombol beri ulasan [btn\_beriulasan]
    4. Sistem memanggil fungsi giveReview()
    5. Sistem memanggil successUlasan()
    6. Ulasan berhasil diberikan
  - b. Alternative Flow
    - E-1 : Pesan Data tidak valid
      1. Sistem memanggil fungsi showError() untuk memberi notifikasi error.
      2. Kembali ke primary flow ke-1
    - E-2 : Error API
      1. Api mengembalikan value Invoice not found / Sudah mengisi ulasan
      2. Sistem memanggil fungsi showError() untuk memberi notifikasi error.
      3. Kembali ke primary flow ke-1
- iii. Post-Condition
  - User kembali ke menu utama

#### 3.1.10.2 UI Design dan Deskripsi Objek UI #6 Ulasan

< Beri Ulasan

 dr. Naufal Zaidan  
Dokter Kelamin

Beri mentor penilaian?

★★★★★

Beri deskripsi penilaian untuk mentor

Deskripsi (Optional)

Dokternya ramah, baik, dan sangat sopan. Mantap!

Beri Ulasan

Gambar 3.21 UI Ulasan

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 43 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-ULS	Page Ulasan	menampilkan inputan berupa deskripsi ulasan dan rating bintang

Tabel 3.29 Tabel Deskripsi Objek UI Ulasan

Page Ulasan

Id_Objek	JENIS	LABEL*	Keterangan**
txt_appbar_tittle	TextView	Beri Ulasan	Tittle untuk appbar
txt_nama_dokter	TextView	dr,Naufal Zaidan	menampilkan nama dokter berupa teks
txt_profesidokter	TextView	Dokter kelamin	menampilkan profesi dokter berupa teks
img_profile	IMG		menampilkan foto dokter berupa image
txt_question	TextView	Beri Dokter penilaian?	menampilkan pertanyaan berupa teks
ir_rating	RatingBar		inputan dari user berupa rating bintang 1 sampai 5
txt_deskripsi	TextView	Beri deskripsi penilaian untuk dokter	menampilkan teks berupa kalimat
et_deskripsi	EditText	Deskripsi (optional)	inputan dari user berupa deskripsi ulasan dalam bentuk teks
btn_beriulasan	Button	Beri ulasan	jika di klik akan memanggil fungsi submit()

Tabel 3.30 Page Ulasan

### 3.1.10.3 Identifikasi Object Baru & Tipe Kelas #6 Ulasan

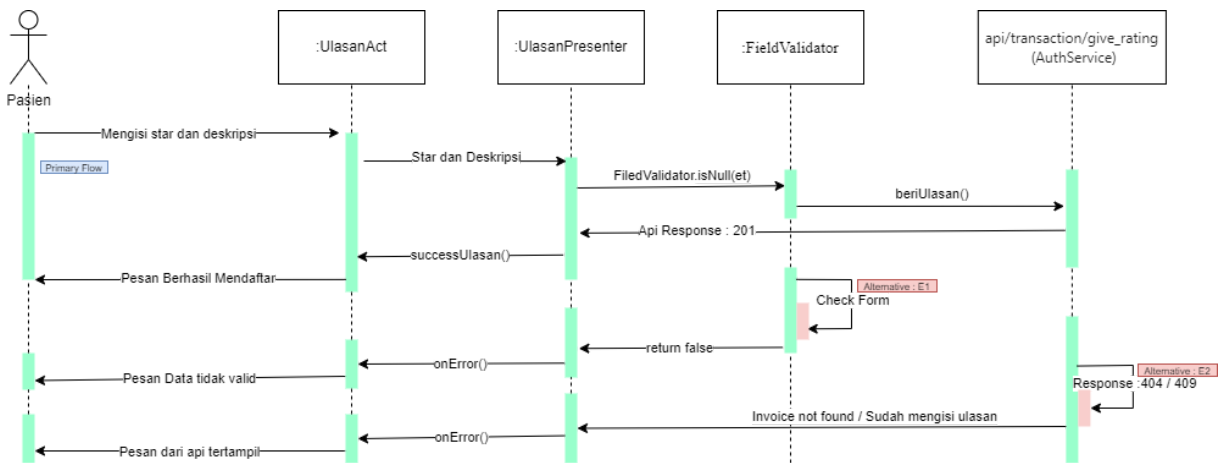
TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1.	UlasanAct	Boundary
2.	UlasanPresenter	Controller
3.	UlasanModel	Entity
4.	Database	Entity
5.	FieldValidator	Controller

Tabel 3.31 Tabel Object Perancangan Ulasan

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 44 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

3.1.10.4 Sequence Diagram #6 Ulasan



Gambar 3.22 Sequence Diagram Ulasan

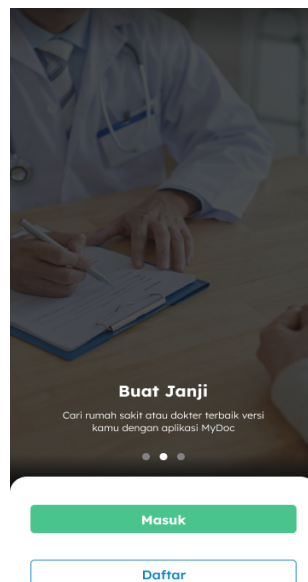
### 3.1.11 Use Case #11 Onboarding

#### 3.1.11.1 Use Case Scenario #11 Onboarding

Skenario Use Case #11 :

- i. Pre-Condition
  - User sudah menginstall aplikasi “MyDoc” di Playstore
- ii. Use Case Description
  - a. Primary Flow
    7. User mendownload aplikasi terlebih dahulu
    8. User membuka aplikasi “MyDoc”
  - b. Alternative Flow
    - E-1 : User belum melakukan instalasi aplikasi MyDoc atau belum mengupdate aplikasi. Menampilkan pesan “Update versi terbaru”.
- iii. Post-Condition
  - User dialihkan ke halaman register atau halaman login.

#### 3.1.11.2 UI Design dan Deskripsi Objek UI #11 Onboarding



**Gambar 3.23** UI Onboarding

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-ULS	Page Onboarding	menampilkan overlay introduction page beserta pilihan menu register dan login berupa button.

**Tabel 3.32** Tabel Deskripsi Objek UI Onboarding

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 46 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

#### Page Onboarding

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
txt_title	TextView	Buat Janji	Tittle untuk appbar
txt_deskripsi	TextView	cari rumah sakit atau dokter terbaik versi kamu dengan aplikasi MyDoc	menampilkan nama dokter berupa teks
img_onboarding	IMG		menampilkan foto background pada menu onboarding
btn_register	Button	Daftar	jika diklik akan memanggil fungsi pindahactivity
btn_login	Button	Masuk	jika diklik akan memanggil fungsi pindah activity

**Tabel 3.33** Page Onboarding

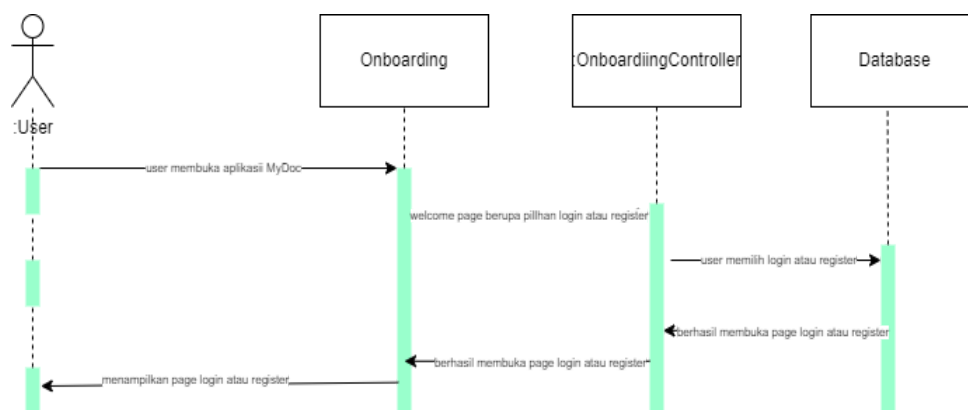
#### 3.1.11.3 Identifikasi Object Baru & Tipe Kelas #11 Onboarding

##### TABEL OBJECT PERANCANGAN

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1.	OnboardingAct	Boundary
2.	OnboardingPresenter	Controller
3.	OnboardingModel	Entity
4.	Database	Entity

**Tabel 3.34** Tabel Object Perancangan Onboarding

#### 3.1.11.4 Sequence Diagram #11 Onboarding



**Gambar 3.24** Sequence Diagram Onboarding

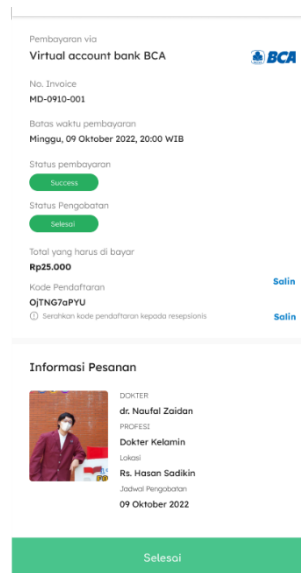
### 3.1.12 Use Case #12 Invoice

#### 3.1.12.1 Use Case Scenario #12 Invoice

Skenario Use Case #12 :

- i. Pre-Condition
  - User sudah melakukan #6 Mendaftar Berobat
- ii. Use Case Description
  - a. Primary Flow
    1. Sistem memanggil fungsi getInvoice()
    2. Sistem mengembalikan data melalui fungsi loadInvoice()
    3. Pesan sukses tertampil
- iii. Post-Condition
  - Data invoice tertampil

#### 3.1.12.2 UI Design dan Deskripsi Objek UI #12 Invoice



Gambar 3.25 UI MD-INV

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-INV	Invoice Layout	Layout untuk menampilkan invoice user

Tabel 3.35 Tabel Deskripsi Objek UI Invoice

Page Invoice – MD-INV

Id_Objek	JENIS	LABEL*	Keterangan**
txt_pembayaran via	TextView	Pembayaran via	Mengetahui pembayaran menggunakan via tertentu
txt_noinvoice	TextView	No. Invoice	Mengetahui nomor invoice

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 48 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		



<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
txt_btswaktu	TextView	Batas waktu pembayaran	Menampilkan batas waktu pembayaran
txt_statuspembayaran	TextView	Status pembayaran	Menampilkan status pembayaran pasien
btn_succes	Button	success	Menampilkan status pembayaran pasien success
btn_selesai	Button	selesai	Menampilkan status pengobatan pasien telah selesai
txt_total	TextVuw	Total yang harus dibayar	Menampilkan total harga yang harus dibayar oleh pasien
txt_kode	TextView	Kode Pendaftaran	Menampilkan kode pendaftaran pasien
btn_salin	Button	Salin	Button untuk menyalin atau mengcopy kode pembayaran
txt_informasi	TextView	Informasi Pesanan	Berisi Informasi yang dipesan oleh pasien
txt_dokter	TextView	Dokter	Menampilkan nama dokter yang dipilih oleh pasien
txt_profesi	TextView	Profesi	Menampilkan profesi dokter yang dipilih
txt_lokasi	TextView	Lokasi	Menampilkan lokasi rumah sakit yang dipilih
txt_schedule	TextView	Jadwal Pengobatan	Menampilkan jadwal pengobatan yang dipilih
btn_selesai	Button	selesai	jika diklik akan memanggil fungsi pindah activity
img_dokter	ImageView		Menampilkan gambar dokter

**Tabel 3.36** Page Invoice - MD - INV

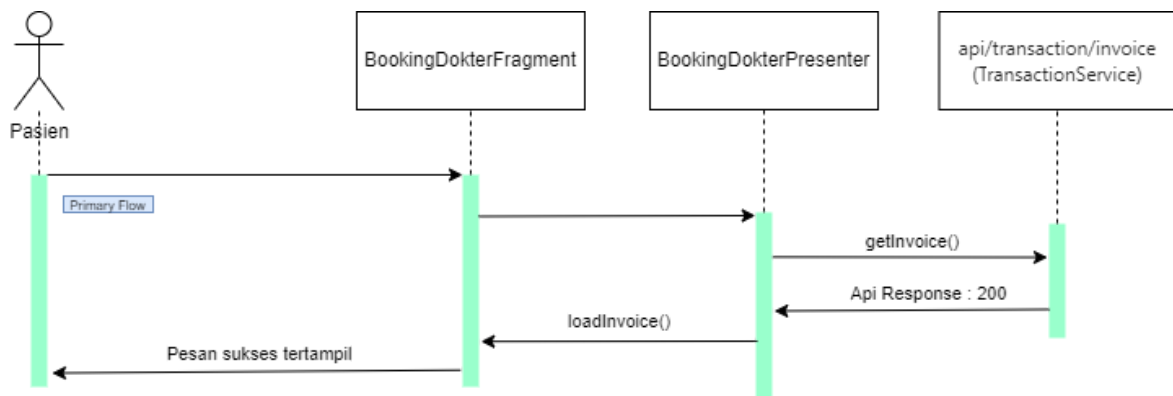
### 3.1.12.3 Identifikasi Object Baru & Tipe Kelas #12 Invoice

**TABEL OBJECT PERANCANGAN**

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1	InvoiceAct	Boundary
2	InvoicePresenter	Controllor
3	InvoiceModel	Entity
4	TransactionService	Entity

**Tabel 3.37** Tabel Object Perancangan Invoice

### 3.1.12.4 Sequence Diagram #12 Invoice



**Gambar 3.26** Sequence Diagram Invoice

### 3.1.13 Use Case #13 Dashboard Instansi Kesehatan

#### 3.1.13.1 Use Case Scenario #13 Dashboard Instansi Kesehatan

Skenario Use Case #13 :

- i. Pre-Condition
  - Data admin telah terdaftar di administrator "MyDoc"
- ii. Use Case Description
  - a. Primary Flow
    1. User memasukkan form et\_username dan et\_password
    2. User menekan tombol SUBMIT
    3. User berhasil login dan diarahkan ke dashboard
- iii. Post-Condition
  - Dashboard terampil

#### 3.1.13.2 UI Design dan Deskripsi Objek UI #13 Dashboard Instansi Kesehatan



**Gambar 3.27** UI DashBoard Instansi Kesehatan

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-DSB	Page Dashboard	Menampilkan Dashboard Instansi Kesehatan

**Tabel 3.38** Tabel Deskripsi Objek UI Dashboard Instansi Kesehatan

Page Dashboard Instansi Kesehatan – MD-DSB

Id_Objek	JENIS	LABEL*	Keterangan**
txt_title	TextView	MyDoc	Tittle untuk appbar
txt_deskripsi	TextView	Silahkan masukkan username dan password anda	Perintah untuk memasukkan username dan password
et_username	EditText	Username	Field Form untuk username sebagai data submit
et_password	EditText	Password	Field From untuk password sebagai data submit
btn_submit	Button	SUBMIT	jika diklik akan memanggil fungsi pindah activity

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 50 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
txt title	TextView	MyDoc	Tittle untuk appbar

**Tabel 3.39** Page Dashboard Instansi Kesehatan - MD - DSB

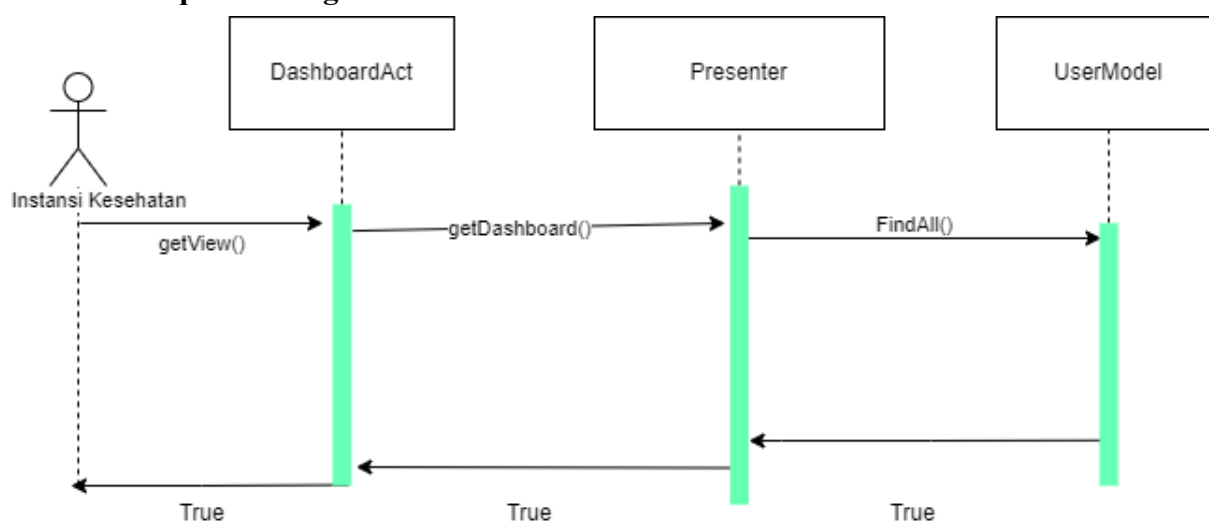
### 3.1.13.3 Identifikasi Object Baru & Tipe Kelas #13 Dashboard Instansi Kesehatan

TABEL OBJECT PERANCANGAN

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1.	DashboardgAct	Boundary
2.	DashboardPresenter	Controller
3.	DashboardModel	Entity
4.	Database	Entity

**Tabel 3.40** Tabel Object Perancangan Dashboard Instansi Kesehatan

### 3.1.13.4 Sequence Diagram #12 Dashboard Instansi Kesehatan



**Gambar 3.28** Sequence Dashboard Instansi Kesehatan

### 3.1.14 Use Case #14 Menambah List Dokter

#### 3.1.14.1 Use Case Scenario #14 Menambah List Dokter

Skenario Use Case #14 :

- i. Pre-Condition
  - Instansi Kesehatan sudah melakukan #2 Login
- ii. Use Case Description
  - a. Primary Flow
    1. Instansi Kesehatan mengisi [foto\_dokter],[nama],[profession] dan [price]
    2. Instansi Kesehatan menekan tombol [btn\_submit]
    3. Sistem memanggil fungsi add()
    4. Sistem memanggil fungsi insert pada DokterModel()
    5. Pesan sukses tampil
- iii. Post-Condition
  - Dokter berhasil ditambahkan

#### 3.1.14.2 UI Design dan Deskripsi Objek UI #14 Menambah List Dokter

The screenshot displays the 'MyDoc' web application interface. On the left is a dark sidebar with navigation links: 'Dashboard', 'List Dokter', 'Verifikasi Antrian', and 'Riwayat Berobat'. The main content area is titled 'Tambah Dokter' and contains a form with the following elements: a 'Foto dokter (Optional)' field with a 'Browse' button, a 'Nama Dokter' text input, a 'Profession' text input, and a 'Price' text input. A green 'Tambah' button is positioned below the 'Price' field. The top of the page features a green header with the 'MyDoc' logo and a user profile dropdown labeled 'Rs. Instansi Kesehatan'. The footer includes a copyright notice: 'Copyright © By MyDoc 2022 Template by: Lupa'.

**Gambar 3.29** UI MD-ADD\_DCT

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-ADD_DCT	Tambah Dokter	Layout untuk menambah dokter dari instansi Kesehatan tertentu

**Tabel 3.41** Tabel Deskripsi Objek UI Menambah List Dokter

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 52 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

#### Page Menambah List Dokter - MD - ADD DCT

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
foto_dokter	ImageInput	Foto Dokter	Untuk menambahkan foto pada profile dokter (Bersifat optional)
nama	TextInput	Nama Dokter	Field untuk nama dokter
profession	TextInput	Profession	Field untuk profesi dokter
price	NumberInput	Price	Field untuk harga / tarif dokter dalam sekali berobat
btn_submit	Button	Tambah	Ketika di klik akan memanggil function add() pada class Dokter

**Tabel 3.42** Page Menambah List Dokter - MD - ADD\_DCT

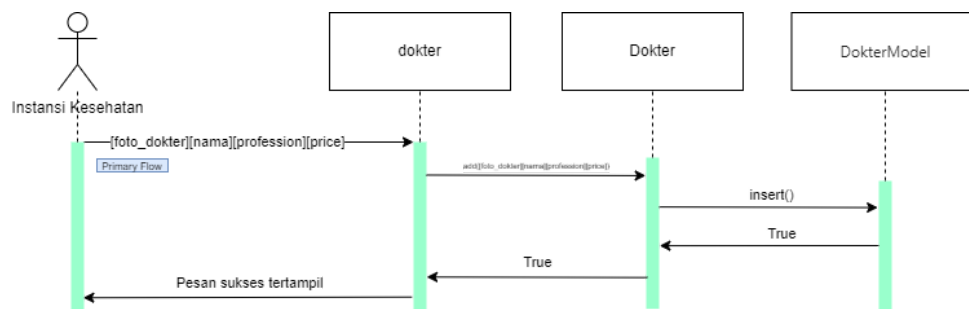
### 3.1.14.3 Identifikasi Object Baru & Tipe Kelas #14 Menambah List Dokter

TABEL OBJECT PERANCANGAN

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1	dokter	Boundary
2	Dokter	Controller
3	DokterModel	Entity

**Tabel 3.43** Tabel Object Perancangan

### 3.1.14.4 Sequence Diagram #14 Menambah List Dokter



**Gambar 3.30** Sequence Diagram Menambah List Dokter

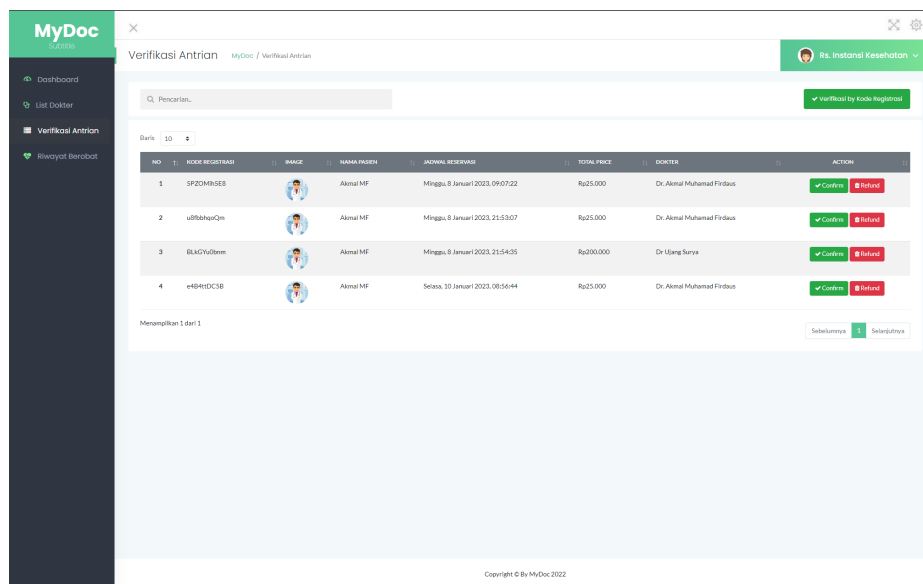
### 3.1.15 Use Case #15 Verifikasi Antrian

#### 3.1.15.1 Use Case Scenario #15 Verifikasi Antrian

Skenario Use Case #15 :

- i. Pre-Condition
  - Terdapat pasien dalam sistem antrian
- ii. Use Case Description
  - a. Primary Flow
    1. Instansi Kesehatan menekan tombol Confirm[btn\_confirm] pada pasien yang menyerahkan kode registrasi
    2. Sistem memanggil function confirm pada controller(Verifikasi Antrian)
    3. Sistem meneruskan ke function confirmInvoice(InvoiceModel)
    4. Sistem memverifikasi antrian
    5. Sistem merubah status antrian menjadi selesai
- iii. Post-Condition
  - Status antrian berubah menjadi selesai

#### 3.1.15.2 UI Design dan Deskripsi Objek UI #15 Verifikasi Antrian



Gambar 3.31 UI MD-VRF

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-VRF	Verifikasi Antrian	Menampilkan sistem antrian dan instansi kesehatan dapat memverifikasi sistem antrian

Tabel 3.44 Tabel Deskripsi Objek UI Verifikasi Antrian

Prodi S1 Informatika Tel-U	DPPL-01	Halaman 54 dari 79
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
btn_confirm	Button	Confirm	menampilkan Modal[modal-confirm] untuk konfirmasi antrian
btn_refund	Button	Refund	Membatalkan sistem antrian dan sistem akan merefund saldo pasien
btn_verif	Button	Verifikasi by Kode Registrasi	Memverifikasi lewat registrasi code secara manual

Tabel 3.45 Page Verifikasi Antrian - MD - VRF

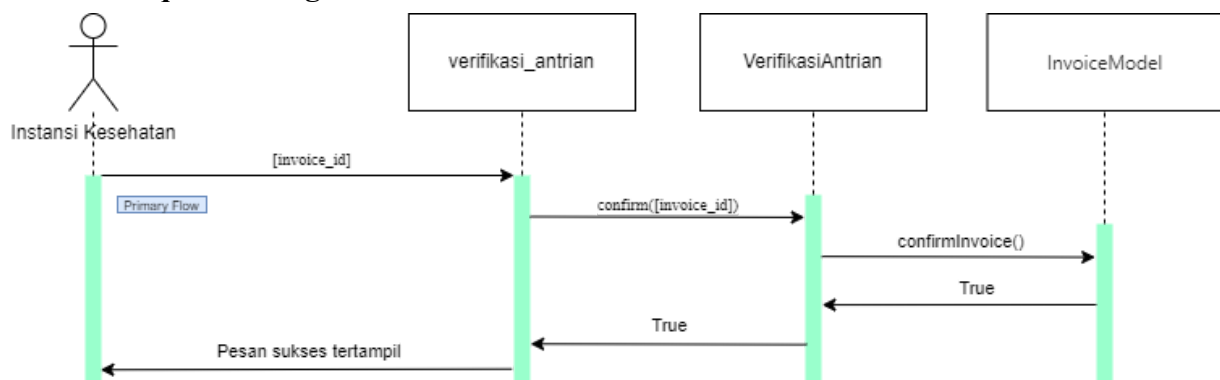
### 3.1.15.3 Identifikasi Object Baru & Tipe Kelas #15 Verifikasi Antrian

TABEL OBJECT PERANCANGAN

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1	verifikasi antrian	Boundary
2	VerifikasiAntrian	Controller
3	InvoiceModel	Entity

Tabel 3.43 Tabel Object Perancangan

### 3.1.15.4 Sequence Diagram #15 Verifikasi Antrian



Gambar 3.32 Sequence Diagram Verifikasi Antrian

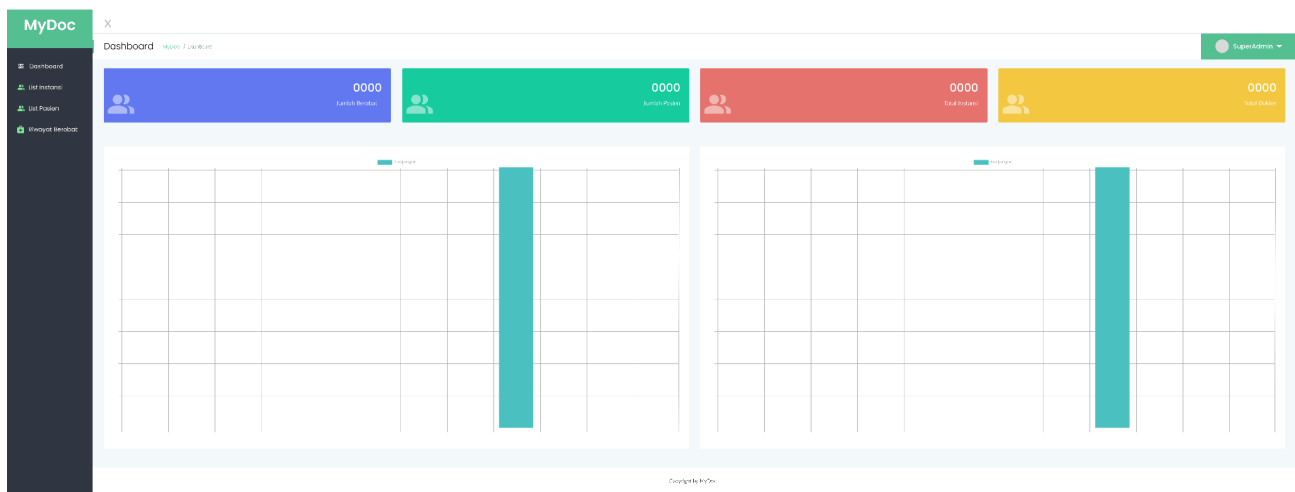
### 3.1.16 Use Case #16 Dashboard Admin

#### 3.1.16.1 Use Case Scenario #16 Dashboard Admin

Skenario Use Case #16 :

- i. Pre-Condition
  - Data admin telah terdaftar di administrator "MyDoc"
- ii. Use Case Description
  - a. Primary Flow
    3. Sistem memanggil fungsi index() pada controller Dashboard
    4. Dashboard admin tertampil
  - b. Alternative Flow
    -
- iii. Post-Condition
  - Dashboard tertampil

#### 3.1.16.2 UI Design dan Deskripsi Objek UI #15 Dashboard Admin



**Gambar 3.33** UI Dashboard Admin

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-DSB	Page Dashboard	Untuk menampilkan halaman utama Admin yang berisi informasi terkait aplikasi MyDoc

**Tabel 3.46** Tabel Deskripsi Objek UI Dashboard Admin

MD-DSB - Page Dashboard Admin

Id_Objek	JENIS	LABEL*	Keterangan**
txt_titleappsidebar	TextView	MyDoc	Title sidebar Aplikasi
btn_close	Button	x	Untuk menutup website admin
txt_tittlenavbar	TextView	Dashboard	Title untuk appbar
txt_titleappnavbar	TextView	Mydoc	Title untuk appbar
txt_subtitlenavbar	TextView	Dashboard	Untuk menampilkan informasi page apa yang sedang dibuka
drpdwn_account	DropDow n	SuperAdmin	Menampilkan informasi detail profil user pada akun admin

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 56 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		



<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
btn_pagesidebar	Button	Dashboard	Menampilkan sidebar dashboard
btn_pagesidebar2	Button	List Instansi	Menampilkan profile dokter
btn_pagesidebar3	Button	List Pasien	Menampilkan nama dokter
btn_pagesidebar4	Button	Riwayat Berobat	Menampilkan spesialis dokter
img_icon_berobat	IMG		Menampilkan Ikon jumlah pasien yang telah berobat
img_icon_pasien	IMG		Menampilkan Ikon jumlah pasien
img_icon_instansi	IMG		Menampilkan Ikon total Instansi
img_icon_dokter	IMG		Menampilkan Ikon Total Dokter
txt_jumlah_berobat	TextView		Menampilkan jumlah pasien yang berobat dalam
txt_jumlah_pasien	TextView		Menampilkan jumlah pasien yang terdaftar di MyDoc
txt_total_instansi	TextView		Menampilkan total Instansi Kesehatan yang teregistrasi pada sistem MyDoc
txt_Total_Dokter	TextView		Menampilkan Total jumlah Dokter pada instansi kesehatan yang telah terdaftar di MyDoc
rv_Linegraph	RecycleView	kunjungan	Menampilkan grafik

**Tabel 3.47** Tabel Deskripsi Objek UI Dashboard Admin

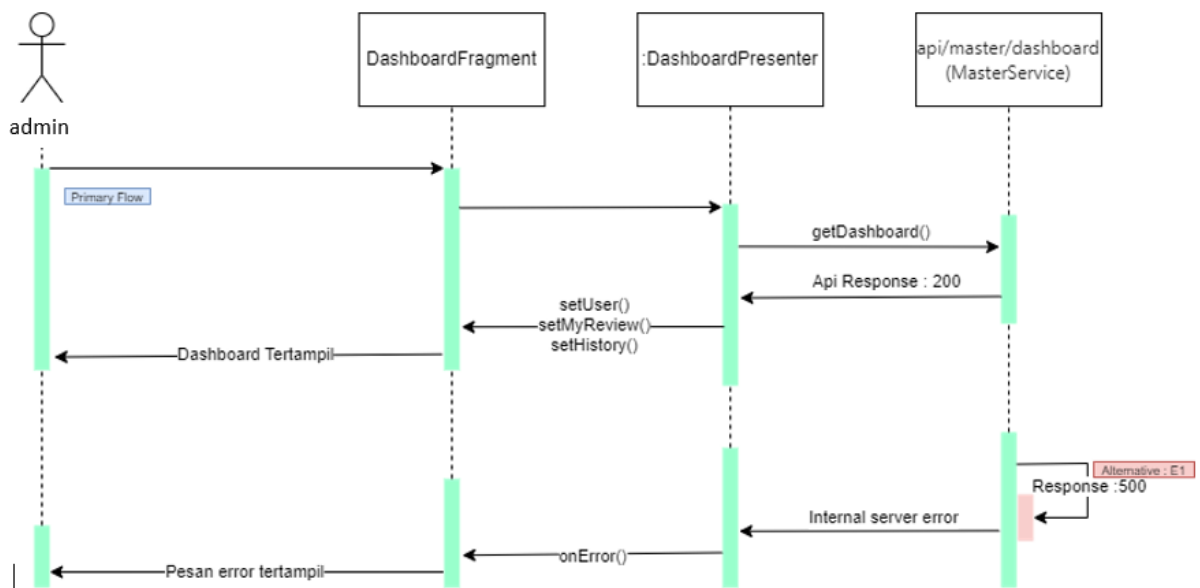
### 3.1.16.3 Identifikasi Object Baru & Tipe Kelas #16 Dashboard Admin

TABEL OBJECT PERANCANGAN

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1.	DashboardAct	Boundary
2.	DashboardPresenter	Controller
3.	DashboardModel	Entity
4.	MasterService	Entity

**Tabel 3.48** Tabel Object Perancangan Dashboard Admin

### 3.1.16.4 Sequence Diagram #15 Dashboard Admin



**Gambar 3.34** Sequence Diagram Dashboard Admin

### 3.1.17 Use Case #17 Mengelola data Riwayat Berobat

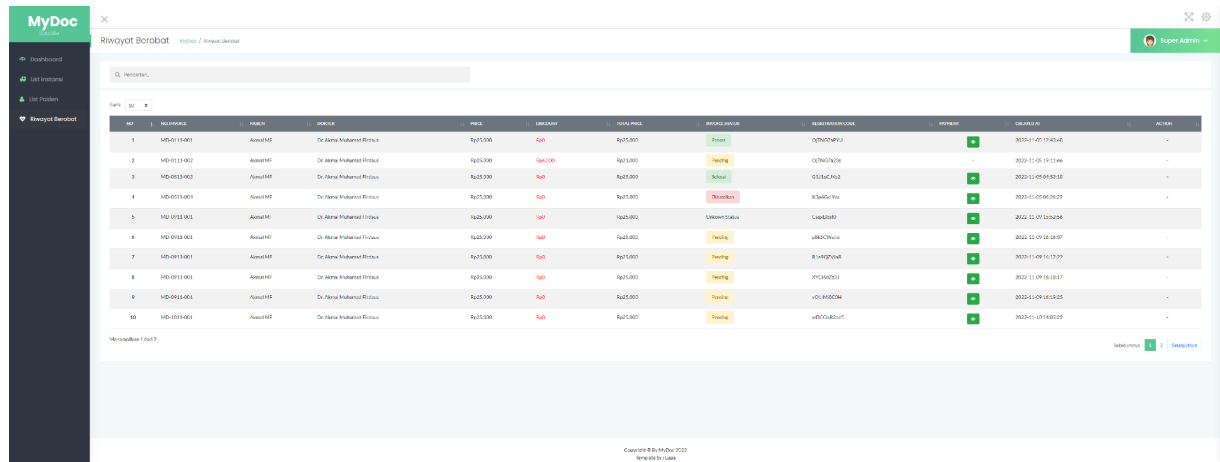
#### 3.1.17.1 Use Case Scenario #17 Mengelola data Riwayat Berobat

Skenario Use Case #17 :

- i. Pre-Condition
  - Instansi Kesehatan sudah login sebagai Instansi Kesehatan
- ii. Use Case Description
  - a. Primary Flow
    1. Instansi Kesehatan memilih menu riwayat berobat
    2. Instansi Kesehatan melihat seluruh informasi riwayat berobat para user
  - b. Alternative Flow
 

E-1 : Instansi Kesehatan belum memiliki akun yang terdaftar sebagai admin
- iii. Post-Condition
  - Riwayat berobat para user telah ditampilkan

### 3.1.17.2 UI Design dan Deskripsi Objek UI #17 Mengelola data Riwayat Berobat



**Gambar 3.35** UI Mengelola Data Riwayat Berobat

Tabel Deskripsi Objek UI Mengelola Data Riwayat Berobat

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-RWBA	Riwayat Berobat Admin	Menampilkan Seluruh List Riwayat Berobat Admin

**Tabel 3.49** Tabel Deskripsi Objek UI Mengelola Data Riwayat Berobat

Page Mengelola Data Riwayat Berobat Admin

Id_Objek	JENIS	LABEL*	Keterangan**
txt_titleappsidebar	TextView	MyDoc	Title untuk sidebar
btn_close	Button	Close	Untuk menutup list riwayat berobat
btn_fullscreen	Button	fullscreen	Untuk menampilkan list riwayat berobat dengan layar full screen
btn_setting	Button	Setting	Untuk menampilkan pengaturan pada website admin
txt_tittlenavbar	TextView	Riwayat Berobat	Menampilkan judul halaman website
txt_titleappnavbar	TextView	MyDoc	Menampilkan nama aplikasi
txt_subtittlenavbar	TextView	Riwayat Berobat	Menampilkan sub judul pada navbar
drpdwn_account	DropDown	Super Admin	Menampilkan detail pada akun detail admin
search_riwayatberobat	InputText	Pencarian	Untuk mencari data dari riwayat berobat
txt_baris	TextView	Baris	Untuk menampilkan judul
drpdwn_baris	DropDown	Jumlah	Untuk memilih jumlah baris yang diinginkan
col_riwayatberobat	Column	Column	Menampilkan list riwayat berobat
txt_nocol	TextView	NO	Menampilkan Judul kolom
txt_detailNo	TextView	Nomor	Menampilkan nomor kolom
btn_noInvoice	Button	Listby	Untuk Mengurutkan berdasarkan list invoice
txt_noInvoice	TextView	NO.INVOICE	Menampilkan judul kolom
txt_detailnoInvoice	TextView	Nomor Invoice	Menampilkan nomor invoice
btn_pasien	Button	Listby	Untuk Mengurutkan berdasarkan list pasien
txt_pasien	TextView	PASIEN	Menampilkan judul kolom
txt_detailPasien	TextView	Nama Pasien	Menampilkan nama pasien
btn_dokter	Button	Listby	Untuk Mengurutkan berdasarkan list dokter
txt_dokter	TextView	DOKTER	Menampilkan judul kolom
txt_detaiDokter	TextView	Nama Dokter	Menampilkan nama dokter
btn_price	Button	Listby	Untuk Mengurutkan berdasarkan harga
txt_price	TextView	PRICE	Menampilkan judul kolom
txt_detailPrice	TextView	Harga	Menampilkan harga
btn_disc	Button	Listby	Untuk Mengurutkan berdasarkan diskon

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
txt_disc	TextView	DISCOUNT	Menampilkan judul kolom
txt_detailDisc	TextView	Total Discount	Menampilkan total discount
btn_totalPrice	Button	Listby	Untuk Mengurutkan berdasarkan total harga
txt_totalPrice	TextView	TOTAL PRICE	Menampilkan judul kolom
txt_detailTotalPrice	TextView	Total Harga	Menampilkan total harga
btn_statusInvo	Button	Listby	Untuk Mengurutkan berdasarkan status invoice
txt_statusInvo	TextView	INVOICE STATUS	Menampilkan judul kolom
txt_detailstatusInvo	TextView	Status Invoice	Menampilkan status invoice
btn_regiscode	Button	Listby	Untuk Mengurutkan berdasarkan kode registrasi
txt_regiscode	TextView	REGISTRATION CODE	Menampilkan judul kolom
txt_regiscode	TextView	Kode Registrasi	Menampilkan kode registrasi
btn_payment	Button	Listby	Untuk Mengurutkan berdasarkan payment
txt_payment	TextView	PAYMENT	Menampilkan judul kolom
txt_detailPayment	TextView	Pembayaran detail	Menampilkan detail pembayaran
btn_create	Button	Listby	Untuk Mengurutkan berdasarkan tanggal dibuat
txt_create	TextView	CREATED AT	Menampilkan judul kolom
txt_detailCreate	TextView	Tanggal dibuat	Menampilkan tanggal dibuat
btn_action	Button	Listby	Untuk Mengurutkan berdasarkan action
txt_action	TextView	ACTION	Menampilkan judul kolom
txt_detailAction	TextView	Status Action	Menampilkan detail action
txt_viewdown	TextView	Menampilkan 1 dari 2	Menampilkan detail halaman
btn_prev	Button	Sebelumnya	Untuk menampilkan halaman sebelumnya
btn_noHalaman	Button	Nomor	Untuk beralih ke halaman sesuai dengan nomor halaman
btn_next	Button	Selanjutnya	Untuk menampilkan halaman Selanjutnya
btn_sidebarDashboard	Button	Dashboard	Untuk menampilkan halaman dashboard
btn_sidebarListInstansi	Button	List Instansi	Untuk menampilkan halaman list instansi
btn_sidebarListPasien	Button	List Pasien	Untuk menampilkan halaman list pasien
btn_sidebarRiwayatBerobat	Button	Dashboard	Untuk menampilkan halaman riwayat berobat

**Tabel 3.50** Mengelola Data Riwayat Berobat Admin

Identifikasi Object Baru & Tipe Kelas #16 Riwayat Berobat Admin

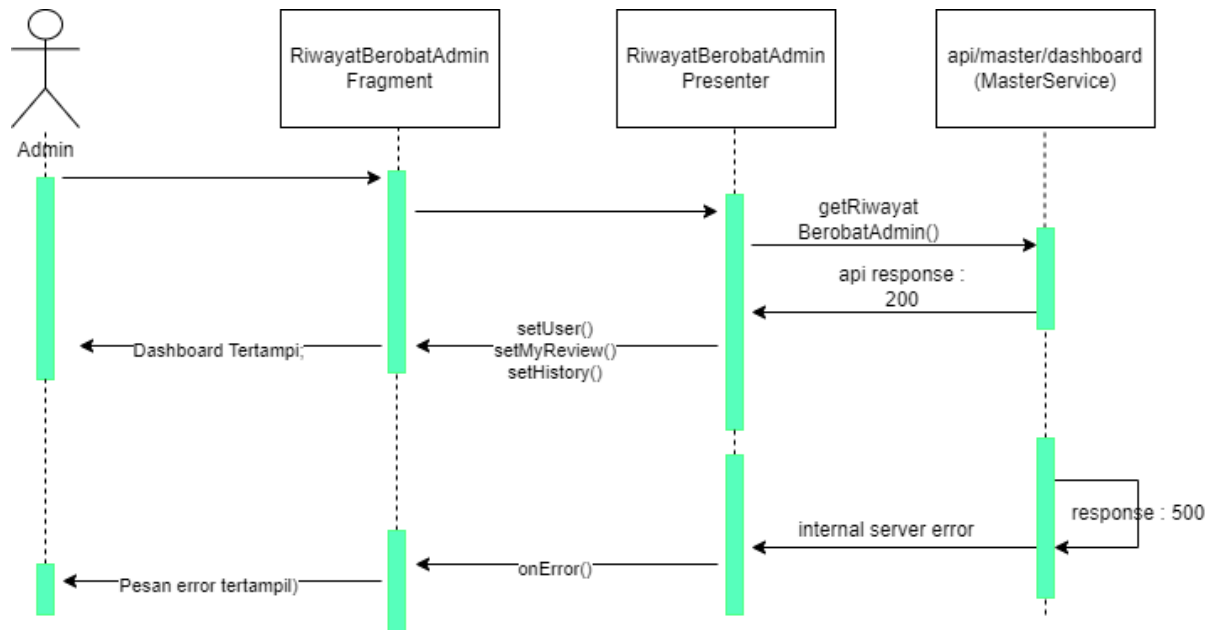
**TABEL OBJECT PERANCANGAN**

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1.	RiwayatBerobatAdminACT	Boundary
2.	RiwayatBerobatAdminPresenter	Controller
3.	RiwayatBerobatAdminModel	Entity
4.	TransactionService	Entity

**Tabel 3.51** Tabel Object Perancangan Mengelola Data Riwayat Berobat

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 60 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

### 3.1.17.3 Sequence Diagram



**Gambar 3.36** Sequence Mengelola Riwayat Berobat Instansi Kesehatan

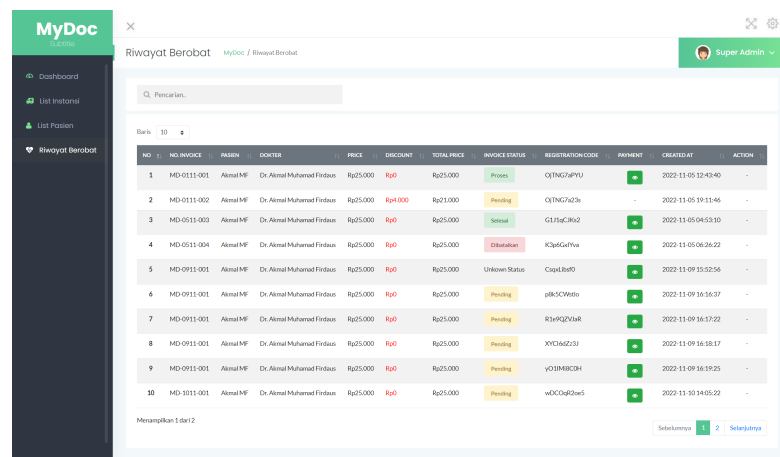
### 3.1.18 Use Case #18 Melihat Data Riwayat Berobat

#### 3.1.18.1 Use Case Scenario #18 Melihat Data Riwayat Berobat

Skenario Use Case #18:

- i. Pre-Condition
  - Admin sudah melakukan #2 Login menggunakan role admin
- ii. Use Case Description
  - a. Primary Flow
    1. Sistem memanggil fungsi index() pada controller[RiwayatBerobat]
    2. Sistem memanggil fungsi getInvoice() pada model[InvoiceModel]
    3. Sistem mengembalikan data invoice
- iii. Post-Condition
  - Invoice user tertampil

#### 3.1.18.2 UI Design dan Deskripsi Objek UI #18 Melihat Data Riwayat Berobat



**Gambar 3.37** UI MD-SHRB

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-SHRB	Riwayat Berobat	Menampilkan list Riwayat berobat pasien untuk role admin

**Tabel 3.52** Tabel Deskripsi Objek UI Melihat Data Riwayat Berobat

Page Melihat Data Riwayat Berobat MD-SHRB

Id_Objek	JENIS	LABEL*	Keterangan**
detail_payment	Button	Icon Mata	Menampilkan detail pembayaran yang digunakan user

**Tabel 3.53** Tabel Page Melihat Data Riwayat Berobat

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 62 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

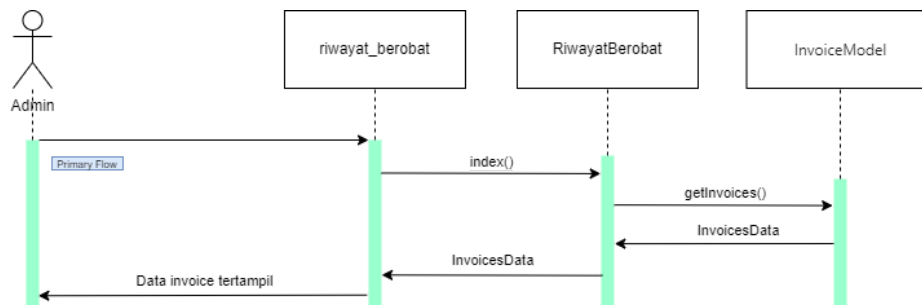
### 3.1.18.3 Identifikasi Object Baru & Tipe Kelas #18 Melihat Data Riwayat Berobat

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	riwayat_berobat	Boundary
2	RiwayatBerobat	Controller
3	InvoiceModel	Entity

Tabel 3.54 Tabel Object Perancangan Melihat Data Riwayat Berobat

### 3.1.18.4 Sequence Diagram #18 Melihat Data Riwayat Berobat



Gambar 3.38 Sequence Diagram Melihat Data Riwayat Berobat

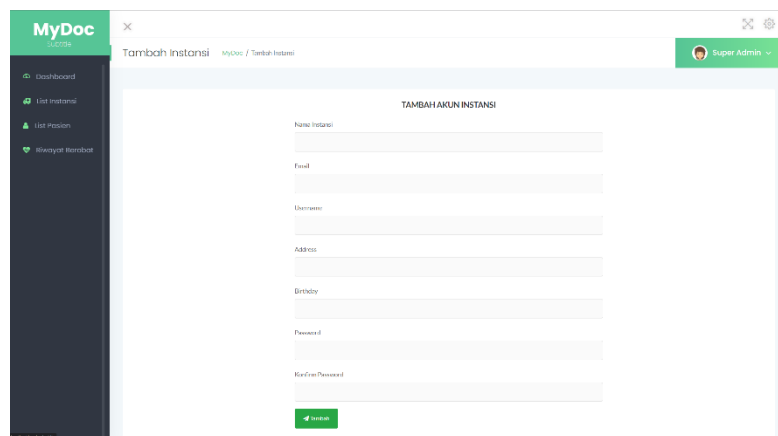
### 3.1.19 Use Case #19 Membuat Akun Instansi Kesehatan

#### 3.1.19.1 Use Case Scenario #19 Membuat Akun Instansi Kesehatan

Skenario Use Case #19 :

- i. Pre-Condition
  - Admin sudah melakukan #2 Login menggunakan role admin
- ii. Use Case Description
  - a. Primary Flow
    1. Admin mengisi [nama],[email],[username],[address],[birthday],[password], dan [password2]
    2. Admin menekan btn\_submit
    3. Sistem memanggil fungsi add() pada controller[Instansi]
    4. Sistem memanggil fungsi createInstansi() pada model[UserModel]
    5. Pesan sukses tampil
- iii. Post-Condition
  - Instansi Kesehatan berhasil ditambahkan

#### 3.1.19.2 UI Design dan Deskripsi Objek UI #19 Membuat Akun Instansi Kesehatan



**Gambar 3.39** UI MD-ADDIST

Tabel Deskripsi Objek UI Membuat Akun Instansi Kesehatan

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-ADDIST	Tambah Instansi	Layout untuk menambahkan instansi

**Tabel 3.55** Tabel Deskripsi Objek UI

Page Membuat Akun Instansi Kesehatan – MD-ADDIST

Id_Objek	JENIS	LABEL*	Keterangan**
nama	TextInput	Nama Instansi	Field untuk nama instansi
email	TextInput	Email	Field untuk email
username	TextInput	Username	Field untuk username
address	TextInput	Address	Field untuk address
Birthday	TextInput	Birthday	Field untuk birthday
Password	TextInput	Password	Field untuk password
password2	TextInput	Konfirmasi Password	Field untuk konfirmasi password



<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
btn_submit	Button	Tambah	Ketika di klik akan memanggil function add()

**Tabel 3.56** Page Membuat Akun Instansi Kesehatan - MD - ADDIST

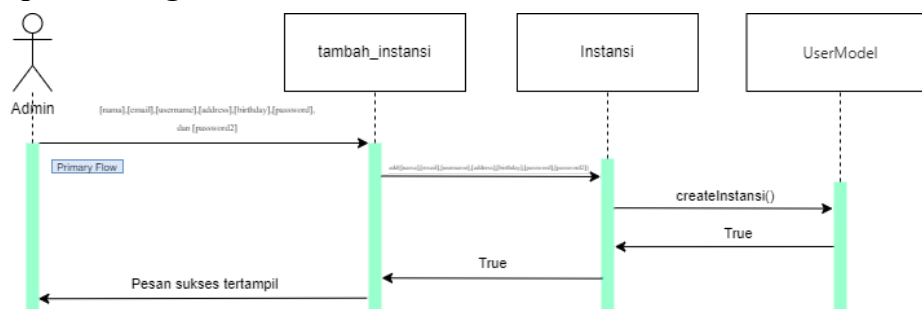
### 3.1.19.3 Identifikasi Object Baru & Tipe Kelas #19 Membuat Akun Instansi Kesehatan

TABEL OBJECT PERANCANGAN

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
1	tambah_instansi	Boundary
2	Instansi	Controller
3	UserModel	Entity

**Tabel 3.57** Tabel Object Perancangan Membuat Akun Instansi Kesehatan

### 3.1.19.4 Sequence Diagram #19 Membuat Akun Instansi Kesehatan



**Gambar 3.40** Sequence Diagram Membuat Akun Instansi Kesehatan

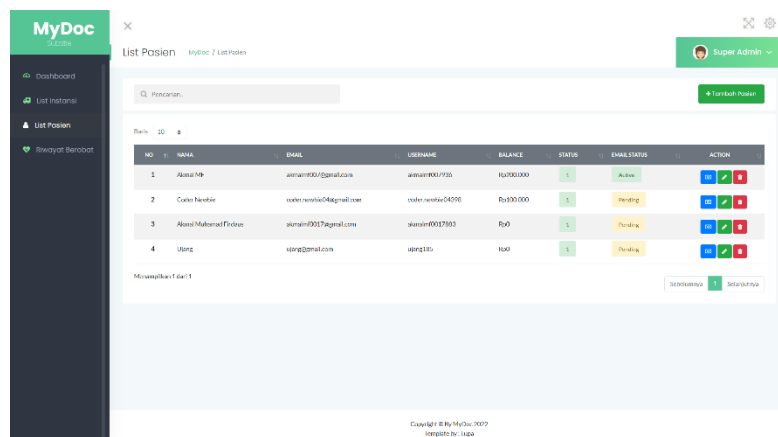
### 3.1.20 Use Case #20 Mengelola User

#### 3.1.20.1 Use Case Scenario #20 Mengelola User

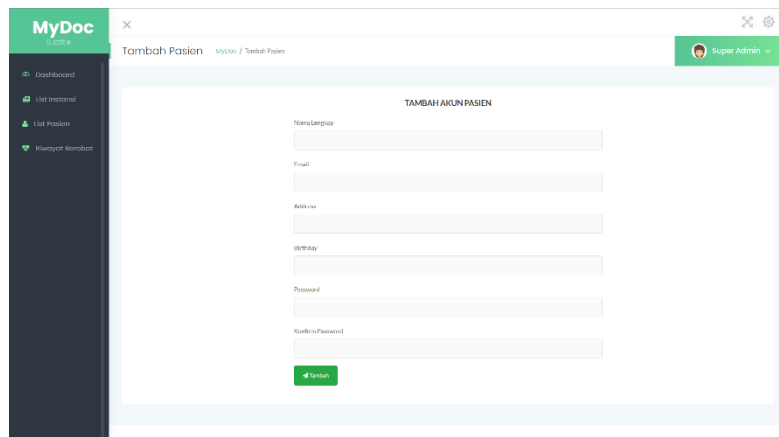
Skenario Use Case #20 :

- i. Pre-Condition
  - Admin sudah melakukan #2 Login menggunakan role admin
- ii. Use Case Description
  - a. Primary Flow
    1. Sistem memanggil fungsi index() pada controller[Pasien]
    2. Sistem memanggil fungsi getListPasien() pada model[UserModel]
  - b. Alternative Flow
    - A-1 : Admin menambah user
      1. Admin mengisi [nama],[email],[username],[address],[birthday],[password], dan [password2]
      2. Admin menekan tombol [btn\_submit]
      3. Sistem memanggil fungsi add() pada controller[Pasien]
      4. Sistem memanggil fungsi createAccount pada model[UserModel]
      5. Pesan sukses tertampil
    - A-2 : Admin mengedit user
      1. Admin mengisi [nama],[status],[active],[password] (Password bersifat optional)
      2. Admin menekan tombol [btn\_edit]
      3. Sistem memanggil fungsi edit() pada controller[Pasien]
      4. Sistem memanggil fungsi update() pada model[UserModel]
      5. Pesan sukses tertampil
    - A-3 : Admin meng-delete pasien
      1. Admin menekan tombol [btn\_remove] pada user yang akan dihapus
      2. Sistem memanggil fungsi delete() pada controller[Pasien]
      3. Sistem memanggil fungsi delete() pada model[UserModel]
      4. Pesan sukses tertampil
- iii. Post-Condition
  - List user tertampil

#### 3.1.20.2 UI Design dan Deskripsi Objek UI #20 Mengelola User



Gambar 3.41 UI MD-EUSR1



**Gambar 3.42** UI MD-EUSR2

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MD-EUSR1	List Pasien	
MD-EUSR2	Tambah Pasien	

**Tabel 3.58** Tabel Deskripsi Objek UI Mengelola User

Page List Pasien – MD-EUSR1

Id_Objek	JENIS	LABEL*	Keterangan**
btn edit	Button	Logo Pensil	Ketika di klik akan memanggil fungsi edit()
btn remove	Button	Logo tempat sampah	Ketika di klik akan memanggil fungsi remove()

**Tabel 3.59** Tabel Page List Pasien

Page Tambah Pasien – MD-EUSR2

Id_Objek	JENIS	LABEL*	Keterangan**
nama	TextInput	Nama Instansi	Field untuk nama instansi
email	TextInput	Email	Field untuk email
username	TextInput	Username	Field untuk username
address	TextInput	Address	Field untuk address
Birthday	TextInput	Birthday	Field untuk birthday
Password	TextInput	Password	Field untuk password
password2	TextInput	Konfirmasi Password	Field untuk konfirmasi password
btn_submit	Button	Tambah	Ketika di klik akan memanggil function add()

**Tabel 3.60** Tabel Page Tambah Pasien

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 67 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

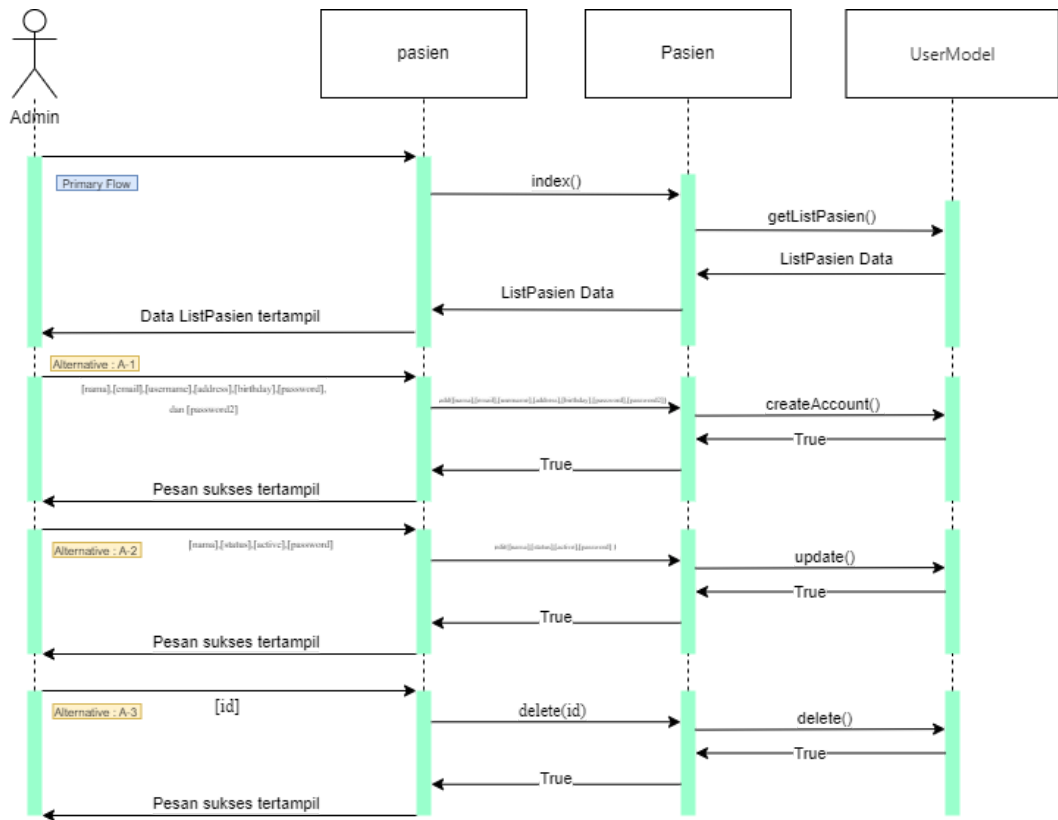
3.1.20.3 Identifikasi Object Baru & Tipe Kelas #20 Mengelola User

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	pasien	Boundary
2	Pasien	Controllor
3	UserModel	Entity

Tabel 3.61 Tabel Object Perancangan Mengelola User

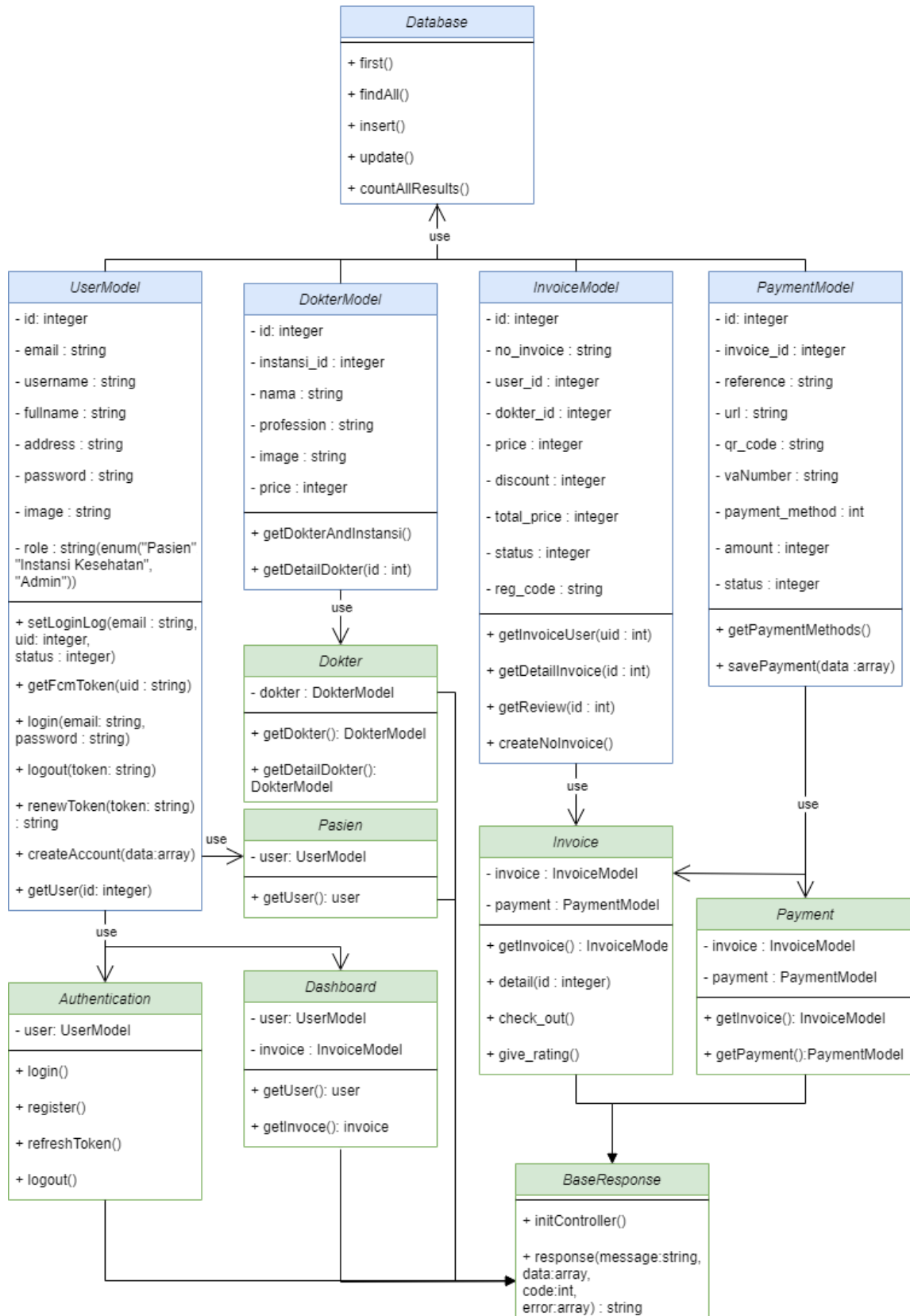
3.1.20.4 Sequence Diagram #20 Mengelola User



Gambar 3.43 Sequence Diagram Mengelola User

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-01</b>	<b>Halaman 69 dari 79</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

### 3.2 Diagram Kelas Keseluruhan



Gambar 3.44 Diagram Class Keseluruhan

**TABEL KELAS :**

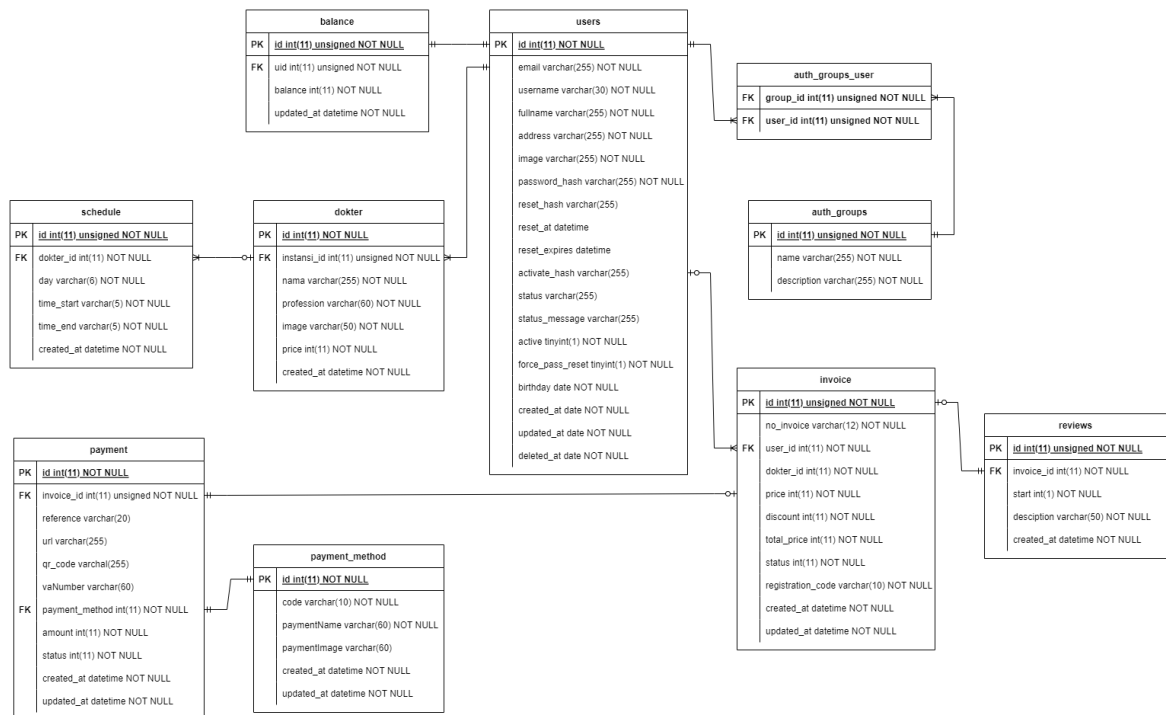
<b>ID Kelas</b>	<b>Nama Kelas Perancangan</b>	<b>Atribute (visibility)</b>	<b>Method / Operation</b>
MDC-01	Database		first()
			findAll()
			insert()
			update()
			countAllResult()
MDC-02	UserModel	id (private) email (private) username (private) address (private) password_hash (private) image (private) role (private)	setLoginLog(email : string, uid: integer, status : integer)
			getFcmToken(uid : string)
			login(email: string, password : string)
			logout(token: string)
			renewToken(token: string): string
			createAccount(data:array)
			getUser(id: integer)
MDC-03	DokterModel	id (private) instansi_id (private) nama (private) profession (private) image (private) price (private)	getDokterAndInstansi()
			getDetailDokter(id : integer)
MDC-04	InvoiceModel	id (private) no_invoice (private) user_id (private) dokter_id (private) price (private) discount (private) total_price (private) status (private) reg_code (private)	getInvoiceUser(uid:integer)
			getDetailInvoice(id:integer)
			getReview(id:integer)
			createNoInvoice()
MDC-05	PaymentModel	id (private) no_invoice (private) reference (private) url (private) qr_code (private) vaNumber (private)	getPaymentMethod()
			savePayment(data:array)

		payment_method (private) amount (private) status (private)	
MDC-06	Dokter	dokter (private)	getDokter()
			getDetailDokter()
MDC-07	Pasien	user (private)	getUser()
MDC-09	Authentication	user (private)	login()
			register()
			refreshToken()
			logout()
MDC-10	Dashboard	user (private) invoice (private)	getUser()
			getInvoice()
MDC-11	Invoice	invoice (private) payment (private)	getInvoice()
			detail()
			check_out()
			give_rating()
MDC-12	Payment	invoice (private) payment (private)	getInvoice
			getPayment()
MDC-13	BaseResponse		initController()
			response(msg:string, data:array, code:int,error:array)

**Tabel 3.62** Tabel Kelas



### 3.3 Perancangan Data / Basis Data



Gambar 3.45 ER-Diagram MyDoc

### 3.4 Perancangan Algoritma dan/atau Query

#### Algoritma #1

Nama Kelas : Register

Nama Operasi : index

Algoritma :

```
public function index()
{
    helper(['form']);
    $rules = [
        'fullname' => 'required|alpha_space',
        'email' =>
'required|valid_email|is_unique[users.email]',
        'birthday' => 'required|date_valid',
        'address' => 'required',
        'password1' => 'required|min_length[6]',
        'password2' => 'required|matches[password1]'
    ];
    if(!$this->validate($rules))
        return $this->response("Form invalid!", null, 422,
$this->validator->getErrors());

    $email = $this->request->getVar("email");
    $uname = explode("@", $email)[0].rand(000,999);
    $data = [
        'fullname' => $this->request->getVar("fullname"),
        'email' => $this->request->getVar("email"),
        'username' => $uname,
        'birthday' => $this->request->getVar("birthday"),
        'address' => $this->request->getVar("address"),
        'password_hash' =>
password_hash($this->request->getVar("password1"),
PASSWORD_BCRYPT),
    ];
    $model = new UserModel();
    if(!$model->createAccount($data))
        return $this->response("Gagal menyimpan ke database!",
null, 500);

    return $this->response("Berhasil register!", null, 201);
}
```

Nama Kelas : Login  
Nama Operasi : index  
Algoritma :

```
public function index()
{
    helper(['form']);
    $rules = [
        'email' => 'required|valid_email',
        'password' => 'required',
        'fcm_token' => 'required'
    ];
    if(!$this->validate($rules))
        return $this->response("Form invalid!", null, 422,
$this->validator->getErrors());
    $model = new UserModel();
    $authModel = new AuthModel();
    $user = $model->where("email",
$this->request->getVar('email'))->first();
    if($user){
        $verify =
password_verify($this->request->getVar("password"),
$user['password_hash']);
        if($verify){
            $jti = substr(md5(time()), 3, 16);
            $login = $authModel->login($user, $jti,
$this->request->getVar("fcm_token"));
            if (!$login)
                return $this->response("Gagal menyimpan token
kedatabase", null, 500);
            $payload = [
                'jti' => $jti,
                'iss' => getenv("app.baseURL"),
                'iat' => time(),
                'exp' => time()+getenv("JWT_TIME_TO_LIVE"),
                'uid' => $user['id'],
                'email' => $user['email']
            ];
            $jwt = JWT::encode($payload, getenv("JWT_SECRET"),
'HS256');
            $rUser = $model->getUser($user["id"]);
            //$rUser['id'] = (int) $rUser['id'];
            return $this->response("Berhasil login!",
['token'=>$jwt, 'user' => $rUser], 200);
        }
    }
    $authModel->setLoginLog($this->request->getVar('email'), null,
0);
    return $this->response("Email atau password salah!", null,
404);
}
```

Nama Kelas : Invoice  
Nama Operasi : index  
Algoritma :

```
public function index()
{
    if(!$this->user)
        return $this->response("Kesalahan pada database!", null,
500);
    $invoiceModel = new InvoiceModel();
    return $this->response("Berhasil Mendapatkan data!",
$invoiceModel->getInvoiceUser($this->user['id'], 0), 200);
}
```

Nama Kelas : Invoice  
Nama Operasi : check\_out  
Algoritma :

```
public function check_out()
{
    if(!$this->user)
        return $this->response("Kesalahan pada database!", null,
500);

    helper(['form']);
    $rules = [
        'dokter_id' => 'required|is_not_unique[dokter.id]',
        'payment_method' =>
'required|is_not_unique[payment_methods.code]',
        'time' => 'required'
    ];
    if(!$this->validate($rules))
        return $this->response("Harap isi formulir dengan benar!",
null, 422, $this->validator->getErrors());

    $dokterModel = new DokterModel();
    $dokter = $dokterModel->where(['id' =>
$this->request->getVar("dokter_id")])>first();
    $options = [
        'baseURI' => 'https://sandbox.duitku.com',
        'timeout' => 3,
    ];
    $client = \Config\Services::curlrequest($options);
    $invoiceModel = new InvoiceModel();
    $paymentModel = new PaymentModel();
    $paymentId =
$paymentModel->getPaymentId($this->request->getVar("payment_method"));
    $orderId = $invoiceModel->getLastOrderId() + 1;
    try {
        $response = $client->request('POST',
'/webapi/api/merchant/v2/inquiry', ['json' =>
[
```

```

        'merchantCode' => getenv("MERCHANT_CODE"),
        'paymentAmount' => $dokter['price'],
        'paymentMethod' =>
$this->request->getVar("payment_method"),
        'merchantOrderId' => $orderId,
        'productDetails' => 'Pembayaran untuk berobat
dengan '.$dokter['nama'],
        'customerVaName' => $this->user['fullname'],
        'email' => $this->user['email'],
        'itemDetails' => [{"name" => "1 Sesi konsultasi
dengan ".$dokter['nama'], 'price' => $dokter['price'], "quantity"=>1}],
        'callbackUrl' =>
getenv("app.baseURL").'/api/v1/callback',
        'returnUrl' =>
getenv("app.baseURL").'/api/v1/return',
        'expiryPeriod' => getenv("PERIOD_CHECKOUT"),
        'signature' =>
md5(getenv("MERCHANT_CODE").$orderId.$dokter['price'].getenv("DUITKU_AP
IKEY"))
    ]
    });
    $json = json_decode($response->getBody(), true);
    if($json['statusCode'] == "00" || $json['statusMessage'] ==
"SUCCESS"){
        helper('invoice');
        $data = [
            'id' => $orderId,
            'user_id' => $this->user['id'],
            'dokter_id' => $this->request->getVar("dokter_id"),
            'price' => $dokter['price'],
            'discount' => 0,
            'total_price' => $dokter['price'],
            'status' => 0,
            'registration_code' => createRegistrationCode()
        ];
        $invId = $invoiceModel->createInvoice($json, $data,
$paymentId);
        if(!$invId)
            return $this->response("Kesalahan pada database!",
null, 500);

            return $this->response("Berhasil melakukan checkout",
$invId, 201);
        } else {
            return $this->response("Kesalahan pada payment
gateway!", null, 500);
        }
    } catch (\Exception $e) {
        return $this->response("Kesalahan pada payment gateway:
".$e->getMessage()."! ", null, 500);
    }
    return $this->response("Kesalahan pada database!", null, 500);
}

```

## Query #2

```
Query      :
START TRANSACTION;
INSERT INTO users VALUES
(fullname,email,username,birthday,address,password_hash);
INSERT INTO balance VALUES(LAST_INSERT_ID(), 0);
INSERT INTO auth_groups_users (group_id, user_id) VALUES(3,
LAST_INSERT_ID());
COMMIT;
```

Keterangan : Query untuk melakukan register

```
Query      :
SELECT * FROM users WHERE email = "{email}"
```

Keterangan : Query untuk melakukan login

```
Query      :
START TRANSACTION;
INSERT INTO invoice VALUES ({id}, {user_id}, {dokter_id}, {price},
{discount}, {total_price}, {status}, {registration_code});
INSERT INTO payment VALUES(LAST_INSERT_ID(), {amount},
{payment_method});
COMMIT;
```

Keterangan : Query untuk melakukan checkout

## 4. Matriks Keruntutan (Requirement Traceability Matrix)

Mapping Requirement dengan Use Case yang direalisasikan

Kode FR	Nama Functional Requirement	Nama Use Case
MD-FR-01	Authentication (SRS-MD-01)	1. Login 2. Register
MD-FR-02	Main menu (SRS-MD-02)	1. Dashboard 2. List Dokter 3. Riwayat Berobat 4. Profile
MD-FR-03	Mendaftar berobat (SRS-MD-03)	1. Detail Dokter 2. Mendaftar Berobat
MD-FR-04	Pembayaran (SRS-MD-04)	1. Pembayaran 2. Invoice
MD-FR-05	Ulasan (SRS-MD-05)	1. Memberikan Ulasan
MD-FR-06	Pemantauan dan pengelolaan aplikasi (SRS-MD-06)	1. Mengelola User 2. Dashboard Admin 3. Melihat Data Riwayat Berobat 4. Membuat Akun Instansi Kesehatan

MD-FR-07	Memantau pembayaran pasien (SRS-MD-07)	<ol style="list-style-type: none"> <li>1. Dashboard Instansi Kesehatan</li> <li>2. Mengelola data riwayat berobat</li> </ol>
MD-FR-08	Memproses antrian pasien (SRS-MD-08)	<ol style="list-style-type: none"> <li>1. Mengatur Sistem Antrian</li> </ol>
MD-FR-09	Menambah data dokter (SRS-MD-09)	<ol style="list-style-type: none"> <li>1. Menambah List dokter</li> </ol>

**Tabel 3.63** Tabel Requirement Traceability Matrix