

MANUAL DAN SOURCE CODE SISTEM INFORMASI RUMAH SAKIT TERPADU (SIRUS)



Anggota :

- | | |
|---------------------------|------------|
| 1. Meilina Taffana Dewi | 4611418020 |
| 2. Muhammad Kahfi Aulia | 4611418009 |
| 3. Firza Nanda Rio Aditya | 4611418017 |

PENGANTAR

Sistem Informasi Rumah Sakit Terpadu ini merupakan program yang berfungsi untuk mengolah dan menyimpan data mengenai pasien serta mengintegrasikannya kedalam sistem berbasis server.

Program ini memiliki berbagai fitur yang memudahkan manajemen data pasien , diantaranya dapat mengelola data pasien baik menambah, merubah, maupun menghapus data pasien hingga melakukan melakukan transaksi pembayaran.

Pogram ini dibangun menggunakan Eclipse dengan Bahasa pemrograman Java serta untuk UI menggunakan FXML Untuk manajemen database sendiri , program ini menggunakan query dari MySQL yang memungkinkan dukungan multiuser serta jaminan kewananan data yang lebih baik.



Mengapa harus Sirius??

Sebelum kita mengerti betapa pentingnya menggunakan aplikasi sirus, terlebih dahulu kita perlu mengetahui pengertian aplikasi tersebut. Sirius ini yaitu suatu sistem yang berurusan dengan pengumpulan data, pengelolaan data, penyajian informasi, analisis dan penyimpulan informasi serta penyampaian informasi yang dibutuhkan untuk kegiatan rumah sakit

Sebuah sistem informasi rumah sakit idealnya mencakup integrasi fungsi-fungsi klinikal (medis), keuangan, serta manajemen yang nantinya merupakan sub sistem dari sebuah sistem informasi rumah sakit. Sub sistem ini merupakan unsur dari sistem informasi rumah sakit yang tugasnya menyiapkan informasi berdasarkan fungsi-fungsi yang ada untuk menyederhanakan pelayanan pada suatu rumah sakit.

Aplikasi ini dapat digunakan di kalangan rumah sakit maupun klinik. Selain itu Aplikasi ini Kelebihan diantaranya :

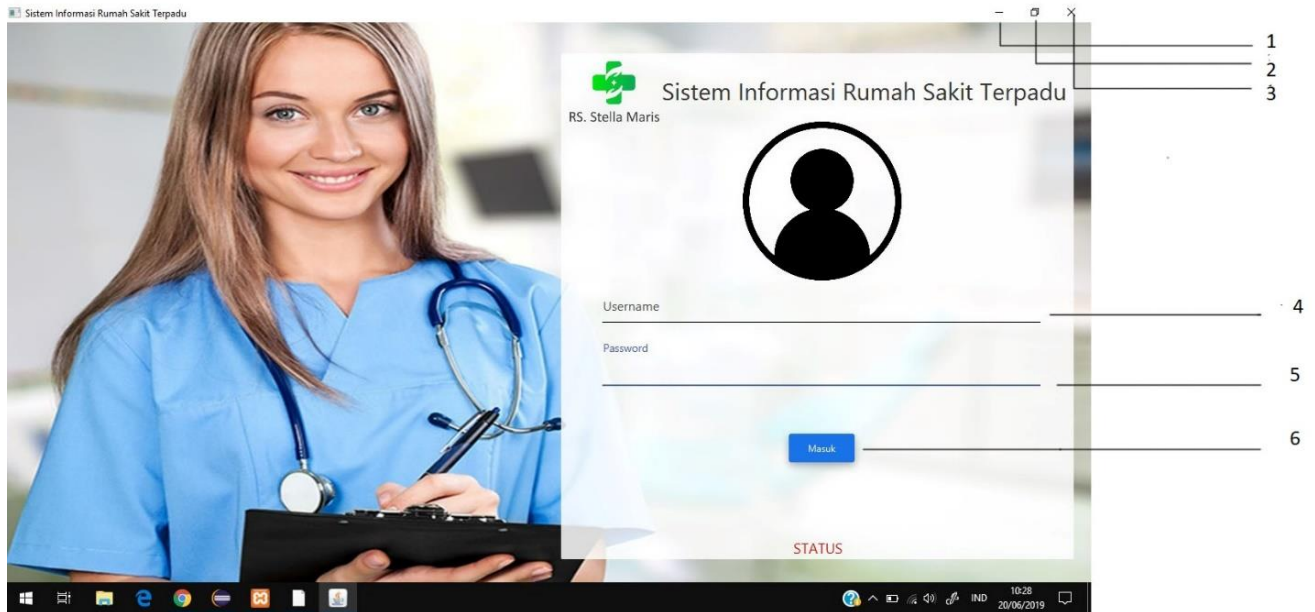
- mempercepat tugas pegawai dalam bekerja
- efisien waktu dan tempat
- informasi dapat diupdate secara cepat dan tersambung jaringan
- dapat menemukan informasi dengan mudah
- dapat dengan mudah meyimpan informasi

Daftar Isi :

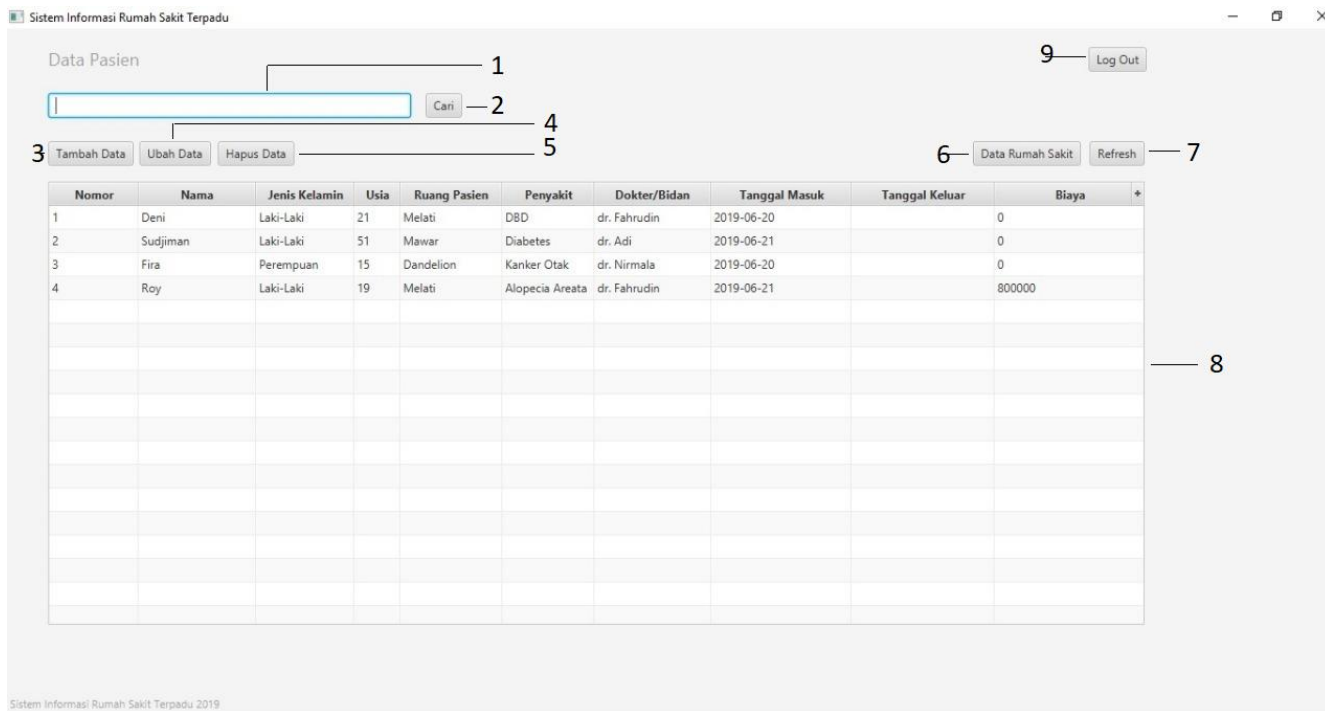
• Pengantar	2
• Daftar Isi	4
• Menu dan Fungsi	5
a) Halaman Login.....	5
b) Halaman Data Pasien.....	5
c) Halaman Tambah Data Pasien	6
d) Halaman Ubah Data Pasien dan Pembayaran	7
e) Halaman Data Rumah Sakit	8
f) Halaman Tambah Data Rumah Sakit	9
g) Halaman Ubah Data Rumah Sakit	10
• Petunjuk Penggunaan	12
Login	12
a) Menambah Data Pasien	13
b) Mengubah Data Pasien	14
c) Menghapus Data Pasien	16
d) Mencari Data Pasien	17
e) Pembayaran	17
f) Menambah Data Pegawai, Dokter atau Ruangan	19
g) Mengubah Data Pegawai, Dokter atau Ruangan	20
h) Mennghapus Data Pegawai, Dokter atau Ruangan	20
• Source Code.....	22
• Database MySQL.....	52

Menu dan Fungsi

A. Halaman Login



B. Halaman Data Pasien



1. **Kolom Pencarian**, mencari data pasien
2. **Cari** , tombol untuk mencari data pasien
3. **Tambah Data**, menambah data pasien
4. **Ubah Data**, mengubah data pasien
5. **Hapus Data**, menghapus data pasien
6. **Data Rumah Sakit**, masuk ke halaman data rumah sakit
7. **Refresh**, memperbarui tampilan data pasien
8. **Tabel Data**, table untuk menampilkan data pasien
9. **Logout**, kembali ke halaman login

C. Halaman Tambah Data Pasien

Sistem Informasi Rumah Sakit Terpadu

File Edit Help

Tambah Data

Nomor : 0

Nama :

Jenis Kelamin :

Usia : 0 Tahun

Ruang :

Penyakit :

Dokter/Bidan :

Tanggal Masuk :

Simpan

1

2

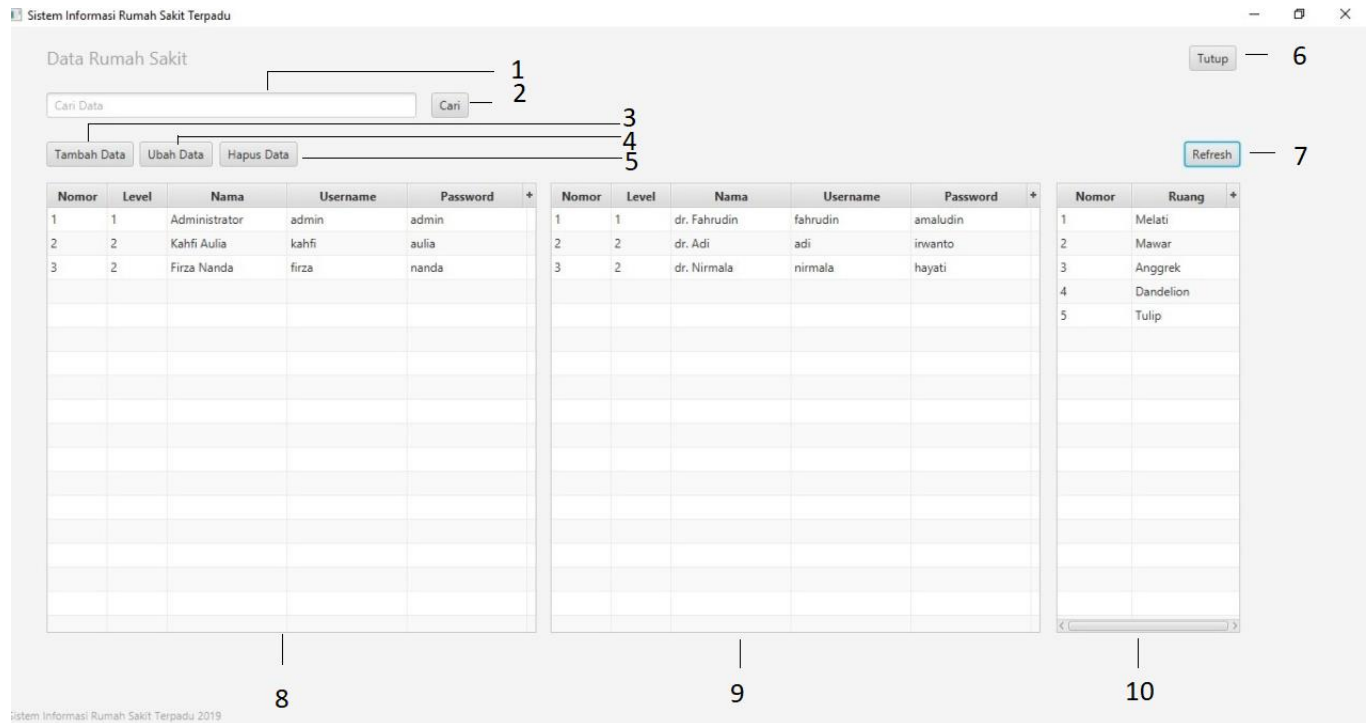
1. **Form Pasien**, formulir tambah data pasien
2. **Simpan**, menyimpan data pasien

D. Halaman Ubah Data Pasien dan Pembayaran

The screenshot displays a web application interface for a hospital information system. It is divided into two main sections: 'Ubah Data' (Change Data) on the left and 'Pembayaran' (Payment) on the right. The 'Ubah Data' section includes input fields for patient details: Nomor (1), Nama (Deni), Jenis Kelamin (Laki-Laki), Usia (21 Tahun), Ruang (Melati), Penyakit (DBD), Dokter/Bidan (dr. Fahrudin), Tanggal Masuk (2019-06-20), and Tanggal Keluar. A 'Simpan' button is at the bottom. The 'Pembayaran' section lists various charges: Biaya Perawatan, Registrasi Kamar, Sewa Kamar, Dokter, Ambulans, Penunjang Medis, and Lain-lain, each with a text input field for the amount. A 'Total Biaya' field shows 'Rp. 0'. At the bottom of the payment section are buttons for 'Bayar', 'Cetak', and 'Tutup'. Numbered annotations 1 through 6 point to specific elements: 1 points to the 'Usia' field, 2 points to the 'Simpan' button, 3 points to the 'Dokter' field, 4 points to the 'Total Biaya' field, 5 points to the 'Bayar' button, and 6 points to the 'Tutup' button.

1. **Form Pasien**, formulir ubah data pasien
2. **Simpan**, menyimpan perubahan data pasien
3. **Form Data Pembayaran**, formulir rincian pembayaran
4. **Bayar**, tombol untuk membayar tagihan
5. **Cetak**, mencetak rincian pembayaran
6. **Tutup**, menutup halaman ubah data

E. Halaman Data Rumah Sakit



1. **Kolom Pencarian**, mencari data pasien
2. **Cari** , tombol untuk mencari data pasien
3. **Tambah data**, menambah data pasien
4. **Ubah data**, mengubah data pasien
5. **Hapus Data**, menghapus data pasien
6. **Refresh**, memperbarui tampilan data pasien
7. **Tutup**, menutup halaman sekarang
8. **Tabel Pegawai**, tabel untuk menampilkan data pegawai
9. **Tabel Dokter**, tabel untuk menampilkan data dokter
10. **Tabel Ruangan**, tabel untuk menampilkan data ruangan

F. Halaman Tambah Data Rumah Sakit

The screenshot displays a web application window titled "Sistem Informasi Rumah Sakit Terpadu". It contains three distinct data entry sections:

- Tambah Data Pegawai (Form 1):** Includes input fields for "Nomor" (0), "Level" (1), "Nama", "Username", and "Password". A "Simpan" button is located at the bottom left, associated with the number 4.
- Tambah Data Dokter (Form 2):** Includes input fields for "Nomor" (0), "Level" (1), "Nama", "Username", and "Password". A "Simpan" button is located below the fields, associated with the number 4, and a "Tutup" button is at the bottom right.
- Tambah Data Ruang (Form 3):** Includes input fields for "Nomor" (0) and "Ruang". A "Simpan" button is located below the fields, associated with the number 4, and a "Tutup" button is at the bottom right.

1. **Form Pegawai**, Formulir tambah data pegawai
2. **Form Dokter**, Formulir tambah data pegawai
3. **Form Ruangan** , Formulir tambah data pegawai
4. **Simpan**, Menyimpan data yang ditambahkan

G. Halaman Tambah Data Rumah Sakit

The screenshot displays a software window titled "Sistem Informasi Rumah Sakit Terpadu" with a menu bar (File, Edit, Help). The main area is divided into three panels:

- Ubah Data Pegawai (Left Panel):**
 - Nomor : 2
 - Level : 2
 - Nama : Kahfi Aulia
 - Username : kahfi
 - Password :
 - Annotation 1 points to the Nama field.
 - Annotation 4 points to the Simpan button.
- Ubah Data Dokter (Top Right Panel):**
 - Nomor : 0
 - Level : 1
 - Nama :
 - Username :
 - Password :
 - Annotation 2 points to the Nama field.
 - Annotation 4 points to the Simpan button.
- Ubah Data Ruang (Bottom Right Panel):**
 - Nomor : 0
 - Ruang :
 - Annotation 3 points to the Ruang field.
 - Annotation 4 points to the Simpan button.
 - A Tutup button is located at the bottom right of this panel.

The footer of the window reads "Sistem Informasi Rumah Sakit Terpadu 2019".

1. **Form Pegawai**, Formulir ubah data pegawai
2. **Form Dokter**, Formulir ubah data pegawai
3. **Form Ruangan** , Formulir ubah data pegawai
4. **Simpan**, Menyimpan data yang diubah

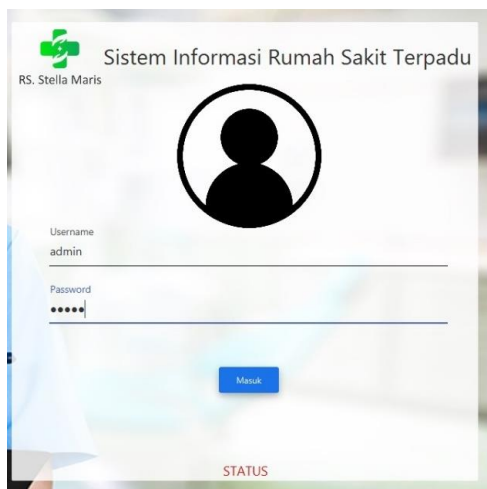
Petunjuk Penggunaan

A. Login

1. Buka halaman login



2. Isikan nama pengguna yang terdaftar (admin/pegawai) pada kolom **“username”** dan kata sandi pengguna pada kolom **“password”**



3. klik tombol **“login”**



4. Jika login sukses, maka akan masuk ke halaman data pasien

Data Pasien Log Out

Cari

Tambah Data Ubah Data Hapus Data Data Rumah Sakit Refresh

Nomor	Nama	Jenis Kelamin	Usia	Ruang Pasien	Penyakit	Dokter/Bidan	Tanggal Masuk	Tanggal Keluar	Biaya	+
1	Deni	Laki-Laki	21	Melati	D8D	dr. Fahrudin	2019-06-20		0	
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21		0	
3	Fira	Perempuan	15	Dandelion	Kanker Otak	dr. Nirmala	2019-06-20		0	
4	Roy	Laki-Laki	19	Melati	Alopecia Areata	dr. Fahrudin	2019-06-21		800000	

B. Menambah Data Pasien

1. Pada halaman data pasien, klik tombol **“tambah data”**



2. Masukkan data pasien pada **“form pasien”** yang tersedia

File Edit Help

Tambah Data

Nomor : 4

Nama : Roy

Jenis Kelamin : Laki-Laki

Usia : 19 Tahun

Ruang : Melati

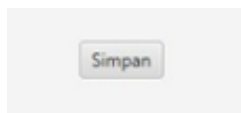
Penyakit : Alopecia Areata

Dokter/Bidan : dr. Fahrudin

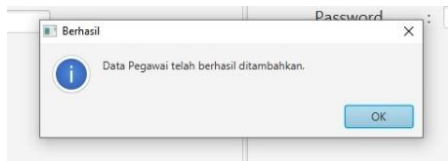
Tanggal Masuk : 2019-06-21

Simpan

3. Klik “simpan”



4. Jika berhasil, maka akan muncul notifikasi



5. Data yang baru ditambahkan akan muncul pada tabel data pasien

Data Pasien

Log Out

Cari

Tambah Data Ubah Data Hapus Data

Data Rumah Sakit Refresh

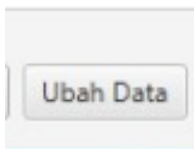
Nomor	Nama	Jenis Kelamin	Usia	Ruang Pasien	Penyakit	Dokter/Bidan	Tanggal Masuk	Tanggal Keluar	Biaya
1	Deni	Laki-Laki	21	Melati	DBD	dr. Fahrudin	2019-06-20		0
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21		0
3	Fira	Perempuan	15	Dandelion	Kanker Otak	dr. Nirmala	2019-06-20		0
4	Roy	Laki-Laki	19	Melati	Alopecia Areata	dr. Fahrudin	2019-06-21		800000

C. Mengubah Data Pasien

1. Klik data yang akan diubah pada halaman data pasien

Nomor	Nama	Jenis Kelamin	Usia	Ruang Pasien	Penyakit	Dokter/Bidan	Tanggal Masuk	Tanggal Keluar	Biaya	+
1	Deni	Laki-Laki	21	Melati	D8D	dr. Fahrudin	2019-06-20		0	
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21		0	
3	Fira	Perempuan	15	Dandelion	Kanker Otak	dr. Nirmala	2019-06-20		0	
4	Roy	Laki-Laki	19	Melati	Alopecia Areata	dr. Fahrudin	2019-06-21		800000	

2. klik tombol “ubah data”



3. Ubah data yang ingin diubah pada “form pasien” yang tersedia

Ubah Data

Nomor : 4

Nama : Roy

Jenis Kelamin : Laki-Laki

Usia : 19 Tahun

Ruang : Melati

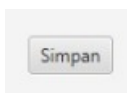
Penyakit : Alopecia Areata

Dokter/Bidan : dr. Fahrudin

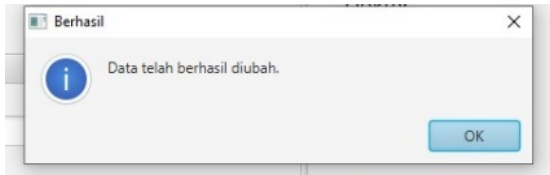
Tanggal Masuk : 2019-06-21

Tanggal Keluar : 2019-06-22

4. Klik “simpan”



5. Jika berhasil, maka akan muncul notifikasi



6. Data pasien pada tabel akan berubah

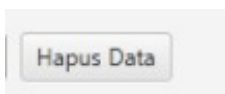
Nomor	Nama	Jenis Kelamin	Usia	Ruang Pasien	Penyakit	Dokter/Bidan	Tanggal Masuk	Tanggal Keluar	Biaya	+
1	Deni	Laki-Laki	21	Melati	D&D	dr. Fahrudin	2019-06-20		0	
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21		0	
3	Fira	Perempuan	15	Dandelion	Kanker Otak	dr. Nirmala	2019-06-20		0	
4	Roy	Laki-Laki	19	Melati	Alopecia Areata	dr. Fahrudin	2019-06-21	2019-06-22	800000	

D. Menghapus Data Pasien

1. Pada halaman data pasien, Klik baris pasien yang akan dihapus pada tabel data pasien

Nomor	Nama	Jenis Kelamin	Usia	Ruang Pasien	Penyakit	Dokter/Bidan	Tanggal Masuk	Tanggal Keluar	Biaya	+
1	Deni	Laki-Laki	21	Melati	D&D	dr. Fahrudin	2019-06-20		0	
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21		0	
3	Fira	Perempuan	15	Dandelion	Kanker Otak	dr. Nirmala	2019-06-20		0	
4	Roy	Laki-Laki	19	Melati	Alopecia Areata	dr. Fahrudin	2019-06-21	2019-06-22	800000	

2. klik tombol “hapus data”



3. Akan muncul notifikasi untuk konfirmasi, pilih “OK”



4. Jika berhasil, maka data akan terhapus dari tabel data pasien

Nomor	Nama	Jenis Kelamin	Usia	Ruang Pasien	Penyakit	Dokter/Bidan	Tanggal Masuk	Tanggal Keluar	Biaya	+
1	Deni	Laki-Laki	21	Melati	D&D	dr. Fahrudin	2019-06-20		0	
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21		0	
3	Fira	Perempuan	15	Dandelion	Kanker Otak	dr. Nirmala	2019-06-20		0	

E. Mencari Data Pasien

1. Pada halaman data pasien, isikan nama pasien yang akan dicari pada “**kolom pencarian**”

Data Pasien

su

2. Klik tombol “cari”

3. Data yang dicari akan muncul pada tabel data pasien

su

Tambah Data Ubah Data Hapus Data Data Rumah Sakit Refresh

Nomor	Nama	Jenis Kelamin	Usia	Ruang Pasien	Penyakit	Dokter/Bidan	Tanggal Masuk	Tanggal Keluar	Biaya	+
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21		0	

F. Pembayaran

1. Klik baris data pasien yang akan melakukan pembayaran pada tabel data pasien

Tambah Data Ubah Data Hapus Data Data Rumah Sakit Refresh

Nomor	Nama	Jenis Kelamin	Usia	Ruang Pasien	Penyakit	Dokter/Bidan	Tanggal Masuk	Tanggal Keluar	Biaya	+
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21		0	
3	Fira	Perempuan	15	Dandelion	Kanker Otak	dr. Nirmala	2019-06-20		0	
4	Lina	Perempuan	10	Dandelion	Malnutrisi	dr. Fahrudin	2019-06-26		0	
6	Deni	Laki-Laki	29	Melati	DBD	dr. Fahrudin	2019-06-20		0	

2. klik tombol “ubah data”

3. Masukkan tanggal keluar pasien pada **form “ubah data”** di sebelah kiri layar

Tanggal Masuk :

Tanggal Keluar :

4. isi rincian biaya pembayaran pada **“form pembayaran”** di sebelah kanan layar

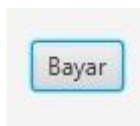
Pembayaran

Biaya Perawatan	:	Rp.	<input type="text" value="200000"/>
Registrasi Kamar	:	Rp.	<input type="text" value="50000"/>
Sewa Kamar	:	Rp.	<input type="text" value="100000"/>
Dokter	:	Rp.	<input type="text" value="100000"/>
Ambulans	:	Rp.	<input type="text" value="0"/>
Penunjang Medis	:	Rp.	<input type="text" value="50000"/>
Lain-lain	:	Rp.	<input type="text" value="0"/>
Total Biaya	:	Rp.	<input type="text" value="0"/>

5. Total biaya yang harus dibayarkan akan muncul pada kolom **“total biaya”**

Total Biaya : Rp.

6. Klik tombol **“bayar”** untuk menyelesaikan pembayaran



7. Jika berhasil, maka kolom tanggal keluar dan biaya pasien akan ditampilkan pada tabel pasien

Tambah Data Ubah Data Hapus Data			Data Rumah Sakit Refresh						
Nomor	Nama	Jenis Kelamin	Usia	Ruang Pasien	Penyakit	Dokter/Bidan	Tanggal Masuk	Tanggal Keluar	Biaya
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21		0
3	Fira	Perempuan	15	Dandelion	Kanker Otak	dr. Nirmala	2019-06-20		0
4	Lina	Perempuan	10	Dandelion	Malnutrisi	dr. Fahrudin	2019-06-26		0
6	Deni	Laki-Laki	29	Melati	DBD	dr. Fahrudin	2019-06-20	2019-06-21	500000

G. Menambah Data Pegawai, Dokter, atau Ruangan

1. Pada halaman data rumah sakit, klik tombol “tambah data”

Data Rumah Sakit

Cari Data Cari

Tambah Data Ubah Data Hapus Data

Nomor	Level	Nama	Username	Password
1	1	Administrator	admin	admin
2	2	Kahfi Aulia	kahfi	aulia
3	2	Firza Nanda	firza	nanda

2. Masukkan data yang akan ditambahkan pada form pegawai, dokter, atau ruangan

File Edit Help

Tambah Data Pegawai

Nomor : 4

Level : 2

Nama : Agaphier

Username : aga

Password : *****

Tambah Data Dokter

Nomor : 4

Level : 2

Nama : dr. Afif

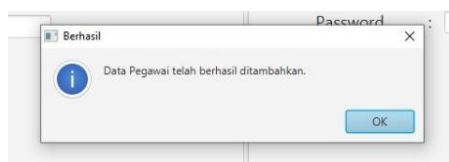
Username : afif

Password : ****

3. Klik “simpan”

Simpan

4. Jika berhasil, maka akan muncul notifikasi



5. Dan data baru yang ditambahkan akan muncul pada tabel

Tambah Data

Ubah Data

Hapus Data

Nomor	Level	Nama	Username	Password	+
1	1	Administrator	admin	admin	
2	2	Kahfi Aulia	kahfi	aulia	
3	2	Firza Nanda	firza	nanda	
4	2	Agaphier	aga	redha	

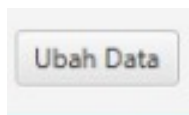
Refresh

Nomor	Level	Nama	Username	Password	+
1	1	dr. Fahrudin	fahrudin	amaludin	
2	2	dr. Adi	adi	irwanto	
3	2	dr. Nirmala	nirmala	hayati	
4	2	dr. Afif	afif	viva	

Nomor	Ruang	+
1	Melati	
2	Mawar	
3	Anggrek	
4	Dandelion	
5	Tulip	
6	Kamboja	

H. Mengubah Data Pegawai, Dokter, atau Ruangan

1. Pada halaman data rumah sakit, klik tombol **“ubah data”**

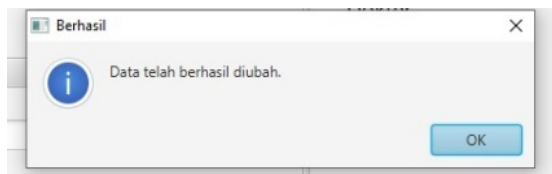


2. Ubah data yang ingin diubah pada form pegawai, dokter, atau ruangan

3. Klik **“simpan”**



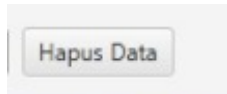
4. Jika berhasil, maka akan muncul notifikasi



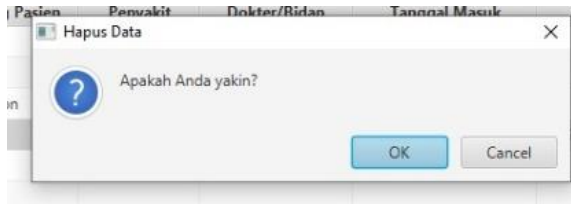
5. Jika berhasil, maka data pada tabel akan berubah

I. Menghapus Data Pegawai, Dokter, atau Ruangan

1. Pada halaman rumah sakit, Klik baris data yang ingin dihapus
2. klik tombol **“hapus data”**



3. Akan muncul notifikasi untuk konfirmasi, pilih **“OK”**



4. Jika berhasil, maka data akan terhapus dari tabel data

Source Code

Layout

```
<?xml version="1.0" encoding="UTF-8"?>

<?import com.jfoenix.controls.JFXButton?>
<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.Menu?>
<?import javafx.scene.control.MenuBar?>
<?import javafx.scene.control.MenuItem?>
<?import javafx.scene.control.SeparatorMenuItem?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.HBox?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.paint.Color?>
```

```

<?import javafx.scene.text.Font?>

<VBox prefHeight="700.0" prefWidth="1400.0" xmlns="http://javafx.com/javafx/8.0.171"
xmlns:fx="http://javafx.com/fxml/1" fx:controller="application.MainController2">

  <children>

    <MenuBar VBox.vgrow="NEVER">

      <menus>

        <Menu mnemonicParsing="false" text="File">

          <items>

            <MenuItem mnemonicParsing="false" text="New" />

            <MenuItem mnemonicParsing="false" text="Open..." />

            <Menu mnemonicParsing="false" text="Open Recent" />

            <SeparatorMenuItem mnemonicParsing="false" />

            <MenuItem mnemonicParsing="false" text="Close" />

            <MenuItem mnemonicParsing="false" text="Save" />

            <MenuItem mnemonicParsing="false" text="Save As..." />

            <MenuItem mnemonicParsing="false" text="Revert" />

            <SeparatorMenuItem mnemonicParsing="false" />

            <MenuItem mnemonicParsing="false" text="Preferences..." />

            <SeparatorMenuItem mnemonicParsing="false" />

            <MenuItem mnemonicParsing="false" text="Quit" />

          </items>

        </Menu>

        <Menu mnemonicParsing="false" text="Edit">

          <items>

```

```

    <MenuItem mnemonicParsing="false" text="Undo" />
    <MenuItem mnemonicParsing="false" text="Redo" />
    <SeparatorMenuItem mnemonicParsing="false" />
    <MenuItem mnemonicParsing="false" text="Cut" />
    <MenuItem mnemonicParsing="false" text="Copy" />
    <MenuItem mnemonicParsing="false" text="Paste" />
    <MenuItem mnemonicParsing="false" text="Delete" />
    <SeparatorMenuItem mnemonicParsing="false" />
    <MenuItem mnemonicParsing="false" text="Select All" />
    <MenuItem mnemonicParsing="false" text="Unselect All" />
  </items>
</Menu>

<Menu mnemonicParsing="false" text="Help">
  <items>
    <MenuItem mnemonicParsing="false" text="About MyHelloApp" />
  </items>
</Menu>
</menus>
</MenuBar>

<AnchorPane id="Content" minHeight="-1.0" minWidth="-1.0" prefHeight="700.0"
prefWidth="1400.0">
  <children>
    <ImageView fitHeight="654.0" fitWidth="1398.0">
      <image>
        <Image url="@gejala-kanker-usus-besar-biasanya-dimulai-dari-gangguan-bab-
jawapos-com.jpg" />

```



```

</image>

</ImageView>

<Label alignment="CENTER" layoutX="122.0" layoutY="3.0" prefHeight="27.0"
prefWidth="344.0" text="Data Pasien" textAlignment="CENTER" textFill="WHITE"
wrapText="false">

    <font>

        <Font size="31.0" fx:id="x6" />

    </font>

</Label>

<TextField fx:id="txtCari" layoutX="104.0" layoutY="68.0" prefHeight="34.0"
prefWidth="371.0" promptText="Cari Pasien" />

<Button fx:id="btnCari" layoutX="466.0" layoutY="60.0" mnemonicParsing="false"
onAction="#Cari" prefHeight="46.0" prefWidth="67.0" style="-fx-background-color:
transparent;" textFill="WHITE">

    <font>

        <Font size="14.0" />

    </font>

    <graphic>

        <ImageView fitHeight="35.0" fitWidth="48.0">

            <image>

                <Image url="@icon/search.png" />

            </image>

        </ImageView>

    </graphic></Button>

    <TableView fx:id="tblPasien" editable="true" layoutX="66.0" layoutY="156.0"
prefHeight="469.0" prefWidth="1286.0" style="-fx-background-color: white;"
tableMenuButtonVisible="true">

    <columns>

```

```

        <TableColumn fx:id="nomor" prefWidth="61.666664123535156"
resizable="false" style="-fx-background-color: #48c8ff; -fx-color: snow;" text="Nomor" />

        <TableColumn fx:id="nama" prefWidth="164.99996948242188" resizable="false"
style="-fx-background-color: #48c8ff; -fx-color: snow;" text="Nama" />

        <TableColumn fx:id="jenisKelamin" prefWidth="111.66665649414062"
resizable="false" style="-fx-background-color: #48c8ff; -fx-color: snow;" text="Jenis Kelamin"
/>

        <TableColumn fx:id="usia" prefWidth="50.333343505859375" resizable="false"
style="-fx-background-color: #48c8ff; -fx-color: snow;" text="Usia" />

        <TableColumn fx:id="ruang" prefWidth="124.33331298828125" resizable="false"
style="-fx-background-color: #48c8ff; -fx-color: snow;" text="Ruang Pasien" />

        <TableColumn fx:id="penyakit" prefWidth="86.33331298828125" resizable="false"
style="-fx-background-color: #48c8ff; -fx-color: snow;" text="Penyakit" />

        <TableColumn fx:id="dokter" prefWidth="156.0" resizable="false" style="-fx-
background-color: #48c8ff; -fx-color: snow;" text="Dokter/Bidan" />

        <TableColumn fx:id="masuk" prefWidth="138.0" resizable="false" style="-fx-
background-color: #48c8ff; -fx-color: snow;" text="Tanggal Masuk" />

        <TableColumn fx:id="masuk1" prefWidth="161.33331298828125" resizable="false"
style="-fx-background-color: #48c8ff; -fx-color: snow;" text="Tanggal Keluar" />

        <TableColumn fx:id="masuk11" prefWidth="246.3333740234375" resizable="false"
style="-fx-background-color: #48c8ff; -fx-color: snow;" text="Biaya" />

    </columns>

</TableView>

<Button fx:id="btnLogout" layoutX="1238.0" layoutY="22.0" mnemonicParsing="false"
onAction="#Logout" style="-fx-background-color: transparent;" text="logout"
textFill="WHITE">

    <font>

        <Font size="14.0" />

    </font>

<graphic>

```

```

<ImageView>

    <image>

        <Image url="@icon/baseline_exit_to_app_black_18dp.png" />

    </image>

</ImageView>

</graphic></Button>

<Button fx:id="btnRefresh" layoutX="1234.0" layoutY="102.0"
mnemonicParsing="false" onAction="#Refresh" style="-fx-background-color: transparent;"
textFill="WHITE">

    <font>

        <Font size="14.0" />

    </font>

    <graphic>

        <ImageView>

            <image>

                <Image url="@icon/baseline_cached_black_18dp.png" />

            </image>

        </ImageView>

    </graphic></Button>

<JFXButton layoutX="140.0" layoutY="112.0" prefHeight="25.0" prefWidth="67.0"
style="-fx-background-color: transparent;" textFill="WHITE">

    <graphic>

        <ImageView fitHeight="34.0" fitWidth="47.0">

            <image>

                <Image url="@icon/baseline_add_box_black_18dp.png" />

            </image>

```

```

        </ImageView>

        </graphic></JFXButton>

        <JFXButton layoutX="262.0" layoutY="114.0" prefHeight="37.0" prefWidth="61.0"
style="-fx-background-color: transparent;" text="Button" textFill="WHITE">

        <graphic>

        <ImageView fitHeight="34.0" fitWidth="36.0">

        <image>

        <Image url="@icon/baseline_delete_sweep_black_18dp.png" />

        </image>

        </ImageView>

        </graphic>

        </JFXButton>

        <JFXButton layoutX="212.0" layoutY="110.0" prefHeight="48.0" prefWidth="71.0"
style="-fx-background-color: transparent;" text=" " textFill="WHITE">

        <graphic>

        <ImageView fitHeight="25.0" fitWidth="36.0">

        <image>

        <Image url="@icon/baseline_create_black_18dp.png" />

        </image>

        </ImageView>

        </graphic>

        </JFXButton>

    </children>

</AnchorPane>

<HBox id="HBox" alignment="CENTER_LEFT" spacing="5.0" VBox.vgrow="ALWAYS">

    <children>

```

```

<Label maxHeight="1.7976931348623157E308" maxWidth="-1.0" text="Sistem Informasi
Rumah Sakit Terpadu 2019" HBox.hgrow="ALWAYS">

    <font>

        <Font size="11.0" fx:id="x3" />

    </font>

    <textFill>

        <Color blue="0.625" green="0.625" red="0.625" fx:id="x4" />

    </textFill>

</Label>

<Pane prefHeight="-1.0" prefWidth="-1.0" HBox.hgrow="ALWAYS" />

<Label font="$x3" maxWidth="-1.0" text="Online" textFill="$x4"
HBox.hgrow="NEVER" />

</children>

<padding>

    <Insets bottom="3.0" left="3.0" right="3.0" top="3.0" />

</padding>

</HBox>

</children>

</VBox>

```

Main Controller

```
package application;
```

```
import java.io.IOException;
import java.net.URL;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Date;
import java.util.Optional;
import java.util.ResourceBundle;

import com.mysql.jdbc.Connection;
import com.mysql.jdbc.PreparedStatement;

import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.scene.Node;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.Alert;
import javafx.scene.control.Alert.AlertType;
import javafx.scene.control.Button;
import javafx.scene.control.ButtonType;
```

```

import javafx.scene.control.TableColumn;
import javafx.scene.control.TableView;
import javafx.scene.control.TextField;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.control.cell.TextFieldTableCell;
import javafx.stage.Stage;
import javafx.util.converter.DateStringConverter;
import javafx.util.converter.IntegerStringConverter;

public class MainController2<Nama> implements Initializable{

    @FXML

    private TableView<Pasien> tblPasien;

    @FXML

    private TableColumn<Pasien, String> nomor;

    @FXML

    private TableColumn<Pasien, String> nama;

    @FXML

    private TableColumn<Pasien, String> jenisKelamin;

    @FXML

    private TableColumn<Pasien, Integer> usia;

    @FXML

    private TableColumn<Pasien, String> ruang;

    @FXML

    private TableColumn<Pasien, String> penyakit;

    @FXML

```

```
private TableColumn<Pasien, String> dokter;  
  
@FXML  
  
private TableColumn<Pasien, Date> masuk;  
  
  
@FXML  
  
public ObservableList<Pasien> list = FXCollections.observableArrayList();  
  
  
  
@FXML  
  
private Button btnTambah;  
  
  
  
@FXML  
  
private Button btnUbah;  
  
  
  
@FXML  
  
private Button btnHapus;  
  
  
  
@FXML  
  
private Button btnCari;  
  
  
  
@FXML  
  
private TextField txtCari;  
  
  
  
@FXML  
  
private Button btnRefresh;
```



```
@FXML
```

```
private Button btnLogout;
```

```
@FXML
```

```
public void Refresh(ActionEvent event) throws IOException{
```

```
    tblPasien.getItems().clear();
```

```
    try {
```

```
        java.sql.Connection c = (Connection) teskoneksi.configDB();
```

```
        String sql = "SELECT * FROM pasiendata";
```

```
        java.sql.Statement s = c.createStatement();
```

```
        ResultSet rs = s.executeQuery(sql);
```

```
        while (rs.next()) {
```

```
            list.add(new Pasien(rs.getString("nomor"), rs.getString("nama"),
rs.getString("jeniskelamin"), rs.getInt("usia"),
rs.getString("ruang"),rs.getString("penyakit"),rs.getString("dokter"),
rs.getDate("tanggalmasuk")));
```

```
        }
```

```
    }
```

```
        catch(Exception e){
```

```
            e.printStackTrace();
```

```
        }
```

```
    }
```

```
@FXML
```

```
public void Logout(ActionEvent event) throws IOException{
```

```

    try{

        Parent halamanUtamaParent =
FXMLLoader.load(getClass().getResource("/application/Tesd.fxml"));

        Scene halamanUtamaScene = new Scene (halamanUtamaParent);

        Stage app_stage = (Stage) ((Node)
event.getSource()).getScene().getWindow();

        app_stage.setTitle("Sistem Informasi Rumah Sakit Terpadu");

        app_stage.setScene(halamanUtamaScene);

        app_stage.show();

    }

    catch(Exception e){

        e.printStackTrace();

    }

}

@FXML

public void Ubah(ActionEvent event) throws IOException{

    try{

        Parent halamanUtamaParent =
FXMLLoader.load(getClass().getResource("/application/ubahdata.fxml"));

        Scene halamanUtamaScene = new Scene (halamanUtamaParent);

        Stage app_stage = (Stage) ((Node)
event.getSource()).getScene().getWindow();

        app_stage.setTitle("Sistem Informasi Rumah Sakit Terpadu");

        app_stage.setScene(halamanUtamaScene);

```

```

        app_stage.show();

    }

    catch(Exception e){
        e.printStackTrace();
    }

}

@FXML

public void Tambah(ActionEvent event) throws IOException{

    try{

        Parent halamanUtamaParent =
FXMLLoader.load(getClass().getResource("/application/tambahdata.fxml"));

        Scene halamanUtamaScene = new Scene (halamanUtamaParent);

        Stage app_stage = (Stage) ((Node)
event.getSource()).getScene().getWindow();

        app_stage.setTitle("Sistem Informasi Rumah Sakit Terpadu");

        app_stage.setScene(halamanUtamaScene);

        app_stage.show();

    }

    catch(Exception e){
        e.printStackTrace();
    }

}

    Connection connection = null;

```

```

PreparedStatement stmt = null;

ResultSet resultSet = null;

@FXML

    public void Cari(ActionEvent event) throws IOException{

        try {

            java.sql.Connection c = (Connection) teskoneksi.configDB();

            String sql = "SELECT * FROM pasiendata WHERE nama =
"+"txtCari.getText() +""";

            java.sql.Statement s = c.createStatement();

            ResultSet rs = s.executeQuery(sql);

            tblPasien.getItems().clear();

            while (rs.next()) {

                list.add(new Pasien(rs.getString("nomor"), rs.getString("nama"),
rs.getString("jeniskelamin"), rs.getInt("usia"),
rs.getString("ruang"),rs.getString("penyakit"),rs.getString("dokter"),
rs.getDate("tanggalmasuk")));

            }

        } catch (SQLException e) {

            // TODO: handle exception

            e.printStackTrace();

        }

    }

@FXML

```

```

public void Hapus(ActionEvent event) throws IOException, SQLException{

    Alert alert = new Alert(AlertType.CONFIRMATION);

    alert.setTitle("Hapus Data");

    alert.setHeaderText(null);

    alert.setContentText("Apakah Anda yakin?");


    Optional<ButtonType> result = alert.showAndWait();

    if (result.get() == ButtonType.OK){

        try {

            btnHapus.setDisable(false);

            Pasien selectedItem =
tblPasien.getSelectionModel().getSelectedItem();

            java.sql.Connection c = (Connection) teskoneksi.configDB();

            String hapusSql = "DELETE FROM pasiendata WHERE nomor = ?";

            stmt = (PreparedStatement) c.prepareStatement(hapusSql);

            stmt.setString(1, selectedItem.getNomor());

            stmt.executeUpdate();


            tblPasien.getItems().remove(selectedItem);

        } catch (SQLException e) {

            // TODO: handle exception

            e.printStackTrace();

        }

    } else {

        // ... user chose CANCEL or closed the dialog

```

```

    }

}

@Override

public void initialize(URL location, ResourceBundle resources) {

    // TODO Auto-generated method stub

    try {

        java.sql.Connection c = (Connection) teskoneksi.configDB();

        String sql = "SELECT * FROM pasiendata";

        java.sql.Statement s = c.createStatement();

        ResultSet rs = s.executeQuery(sql);

        while (rs.next()) {

            list.add(new Pasien(rs.getString("nomor"), rs.getString("nama"),
rs.getString("jeniskelamin"), rs.getInt("usia"),
rs.getString("ruang"),rs.getString("penyakit"),rs.getString("dokter"),
rs.getDate("tanggalmasuk")));

        }

    }

    catch(Exception e){

        e.printStackTrace();

    }

    nomor.setCellValueFactory(new PropertyValueFactory<Pasien,
String>("nomor"));

    nama.setCellValueFactory(new PropertyValueFactory<Pasien, String>("nama"));

```

```

        jenisKelamin.setCellValueFactory(new PropertyValueFactory<Pasien,
String>("jenisKelamin"));

        usia.setCellValueFactory(new PropertyValueFactory<Pasien, Integer>("usia"));
        ruang.setCellValueFactory(new PropertyValueFactory<Pasien, String>("ruang"));
        penyakit.setCellValueFactory(new PropertyValueFactory<Pasien,
String>("penyakit"));

        dokter.setCellValueFactory(new PropertyValueFactory<Pasien,
String>("dokter"));

        masuk.setCellValueFactory(new PropertyValueFactory<Pasien,
Date>("masuk"));

        tblPasien.setItems(list);

        tblPasien.setEditable(true);

        nomor.setCellFactory(TextFieldTableCell.forTableColumn());
        nama.setCellFactory(TextFieldTableCell.forTableColumn());
        jenisKelamin.setCellFactory(TextFieldTableCell.forTableColumn());
        usia.setCellFactory(TextFieldTableCell.forTableColumn(new
IntegerStringConverter()));
        ruang.setCellFactory(TextFieldTableCell.forTableColumn());
        penyakit.setCellFactory(TextFieldTableCell.forTableColumn());
        dokter.setCellFactory(TextFieldTableCell.forTableColumn());
        masuk.setCellFactory(TextFieldTableCell.forTableColumn(new
DateStringConverter()));

    }
}

```

Connection

```
package application;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

class teskoneksi {

    private static Connection mysqlconfig;

    public static Connection configDB()throws SQLException{
        try {
            String url;
            url = "jdbc:mysql://localhost:3306/pasien1 ";
            String user="root"; //user database
            String pass=""; //password database

            DriverManager.registerDriver(new com.mysql.jdbc.Driver());
            mysqlconfig=DriverManager.getConnection(url, user, pass);
        } catch (Exception e) {
            System.err.println("koneksi gagal "+e.getMessage()); //perintah menampilkan error pada koneksi
        }
    }
}
```



```

        return mysqlconfig;
    }
}

```

Initialize

```

@Override

    public void initialize(URL location, ResourceBundle resources) {

        // TODO Auto-generated method stub

        try {

            java.sql.Connection c = (Connection) teskoneksi.configDB();

            String sql = "SELECT * FROM pasiendata";

            java.sql.Statement s = c.createStatement();

            ResultSet rs = s.executeQuery(sql);

            while (rs.next()) {

                list.add(new Pasien(rs.getString("nomor"), rs.getString("nama"),
rs.getString("jeniskelamin"), rs.getInt("usia"),
rs.getString("ruang"),rs.getString("penyakit"),rs.getString("dokter"),
rs.getDate("tanggalmasuk")));

            }

        }

        catch(Exception e){

            e.printStackTrace();

        }
    }

```

```

        nomor.setCellValueFactory(new PropertyValueFactory<Pasien,
String>("nomor"));

        nama.setCellValueFactory(new PropertyValueFactory<Pasien, String>("nama"));

        jenisKelamin.setCellValueFactory(new PropertyValueFactory<Pasien,
String>("jenisKelamin"));

        usia.setCellValueFactory(new PropertyValueFactory<Pasien, Integer>("usia"));

        ruang.setCellValueFactory(new PropertyValueFactory<Pasien, String>("ruang"));

        penyakit.setCellValueFactory(new PropertyValueFactory<Pasien,
String>("penyakit"));

        dokter.setCellValueFactory(new PropertyValueFactory<Pasien,
String>("dokter"));

        masuk.setCellValueFactory(new PropertyValueFactory<Pasien,
Date>("masuk"));

        tblPasien.setItems(list);

        tblPasien.setEditable(true);

        nomor.setCellFactory(TextFieldTableCell.forTableColumn());

        nama.setCellFactory(TextFieldTableCell.forTableColumn());

        jenisKelamin.setCellFactory(TextFieldTableCell.forTableColumn());

        usia.setCellFactory(TextFieldTableCell.forTableColumn(new
IntegerStringConverter()));

        ruang.setCellFactory(TextFieldTableCell.forTableColumn());

        penyakit.setCellFactory(TextFieldTableCell.forTableColumn());

        dokter.setCellFactory(TextFieldTableCell.forTableColumn());

        masuk.setCellFactory(TextFieldTableCell.forTableColumn(new
DateStringConverter()));

```

```

    }
}

```

Tombol Login

```

public void Login(ActionEvent event) throws IOException{

    try{

        java.sql.Connection c = (Connection) teskoneksi.configDB();

        String sql = "SELECT * FROM logindata WHERE username = '"+txtUsername.getText()
+" AND password = '"+txtPassword.getText() +"'";

        java.sql.Statement s = c.createStatement();

        resultSet = s.executeQuery(sql);

        if(!resultSet.next()){

            lblStatus.setText("Login Gagal");

        }else{

            lblStatus.setText("Login Sukses");

        }

        Parent halamanUtamaParent =
FXMLLoader.load(getClass().getResource("/application/halamanUtama.fxml"));

        Scene halamanUtamaScene = new Scene (halamanUtamaParent);

        Stage app_stage = (Stage) ((Node)
event.getSource()).getScene().getWindow();

        app_stage.setTitle("Sistem Informasi Rumah Sakit Terpadu");

        app_stage.setScene(halamanUtamaScene);
    }
}

```

```

        app_stage.show();

    }

}

catch(Exception e){

    e.printStackTrace();

}

}

```

Tombol Tambah Data

```

public void Tambah(ActionEvent event) throws IOException{

    try{

        Parent halamanUtamaParent =
FXMLLoader.load(getClass().getResource("/application/tambahdata.fxml"));

        Scene halamanUtamaScene = new Scene (halamanUtamaParent);

        Stage app_stage = (Stage) ((Node)
event.getSource()).getScene().getWindow();

        app_stage.setTitle("Sistem Informasi Rumah Sakit Terpadu");

        app_stage.setScene(halamanUtamaScene);

        app_stage.show();

    }

    catch(Exception e){

        e.printStackTrace();

    }

}

```

```
}
```

Tombol Hapus Data

```
public void Hapus(ActionEvent event) throws IOException, SQLException{

    Alert alert = new Alert(AlertType.CONFIRMATION);

    alert.setTitle("Hapus Data");

    alert.setHeaderText(null);

    alert.setContentText("Apakah Anda yakin?");


    Optional<ButtonType> result = alert.showAndWait();

    if (result.get() == ButtonType.OK){

        try {

            btnHapus.setDisable(false);

            Pasien selectedItem =
tblPasien.getSelectionModel().getSelectedItem();

            java.sql.Connection c = (Connection) teskoneksi.configDB();

            String hapusSql = "DELETE FROM pasiendata WHERE nomor = ?";

            stmt = (PreparedStatement) c.prepareStatement(hapusSql);

            stmt.setString(1, selectedItem.getNomor());

            stmt.executeUpdate();


            tblPasien.getItems().remove(selectedItem);

        } catch (SQLException e) {

            // TODO: handle exception
```

```

        e.printStackTrace();

    }

    } else {

        // ... user chose CANCEL or closed the dialog

    }

}

```

Tombol Ubah Data

```

public void Ubah(ActionEvent event) throws IOException{

    try{

        Parent halamanUtamaParent =
FXXMLLoader.load(getClass().getResource("/application/ubahdata.fxml"));

        Scene halamanUtamaScene = new Scene (halamanUtamaParent);

        Stage app_stage = (Stage) ((Node)
event.getSource()).getScene().getWindow();

        app_stage.setTitle("Sistem Informasi Rumah Sakit Terpadu");

        app_stage.setScene(halamanUtamaScene);

        app_stage.show();

    }

    catch(Exception e){

        e.printStackTrace();

    }

}

```

```
}
```

Tombol Cari

```
public void Cari(ActionEvent event) throws IOException{

    try {

        java.sql.Connection c = (Connection) teskoneksi.configDB();

        String sql = "SELECT * FROM pasiendata WHERE nama = "
        ""+txtCari.getText()+""";

        java.sql.Statement s = c.createStatement();

        ResultSet rs = s.executeQuery(sql);

        tblPasien.getItems().clear();

        while (rs.next()) {

            list.add(new Pasien(rs.getString("nomor"), rs.getString("nama"),
            rs.getString("jeniskelamin"), rs.getInt("usia"),
            rs.getString("ruang"),rs.getString("penyakit"),rs.getString("dokter"),
            rs.getDate("tanggalmasuk")));

        }

    } catch (SQLException e) {

        // TODO: handle exception

        e.printStackTrace();

    }

}
```

Tombol Refresh

```

public void Refresh(ActionEvent event) throws IOException{

    tblPasien.getItems().clear();

    try {

        java.sql.Connection c = (Connection) teskoneksi.configDB();

        String sql = "SELECT * FROM pasiendata";

        java.sql.Statement s = c.createStatement();

        ResultSet rs = s.executeQuery(sql);

        while (rs.next()) {

            list.add(new Pasien(rs.getString("nomor"), rs.getString("nama"),
rs.getString("jeniskelamin"), rs.getInt("usia"),
rs.getString("ruang"),rs.getString("penyakit"),rs.getString("dokter"),
rs.getDate("tanggalmasuk"))));

        }

    }

    catch(Exception e){

        e.printStackTrace();

    }

}

```

Tombol Save / Simpan


```

public void Simpan(ActionEvent event) throws IOException, SQLException{

    java.sql.Connection con = (Connection) teskoneksi.configDB();

    String sql = "UPDATE pasiendata SET nomor= ?, nama = ?, jeniskelamin = ?, usia = ?,
ruang = ?, penyakit= ?, dokter = ?, tanggalmasuk = ? WHERE nama =
"+txtNama.getText()+"";

    stmt = (PreparedStatement) con.prepareStatement(sql);

    stmt.setString(1, ((TextField)spnNomor.getEditor()).getText());

    stmt.setString(2, txtNama.getText());

    stmt.setString(3, cbKelamin.getValue());

    stmt.setString(4, ((TextField)spnUsia.getEditor()).getText());

    stmt.setString(5, cbRuang.getValue());

    stmt.setString(6, txtPenyakit.getText());

    stmt.setString(7, cbDokter.getValue());

    stmt.setString(8, dpTanggalMasuk.getValue().toString());

    stmt.executeUpdate();

}

```

Tombol Close / Tutup

```

public void Tutup(ActionEvent event) throws IOException{

    try{

        Parent halamanUtamaParent =
FXMLLoader.load(getClass().getResource("/application/halamanUtama.fxml"));

        Scene halamanUtamaScene = new Scene (halamanUtamaParent);

        Stage app_stage = (Stage) ((Node)
event.getSource()).getScene().getWindow();

```

```

        app_stage.setTitle("Sistem Informasi Rumah Sakit Terpadu");

        app_stage.setScene(halamanUtamaScene);

        app_stage.show();

    }

    catch(Exception e){

        e.printStackTrace();

    }

}

```

Tombol Logout

```

public void Logout(ActionEvent event) throws IOException{

    try{

        Parent halamanUtamaParent =
FXMLLoader.load(getClass().getResource("/application/Tesd.fxml"));

        Scene halamanUtamaScene = new Scene (halamanUtamaParent);

        Stage app_stage = (Stage) ((Node)
event.getSource()).getScene().getWindow();

        app_stage.setTitle("Sistem Informasi Rumah Sakit Terpadu");

        app_stage.setScene(halamanUtamaScene);

        app_stage.show();

    }

    catch(Exception e){

        e.printStackTrace();

    }

}

```

}

Database mySQL

localhost / 127.0.0.1 / pasien / dr

phpMyAdmin

Server: 127.0.0.1 » Database: pasien » Table: dokterdata

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 3 (4 total. Query took 0.0007 seconds)

`SELECT * FROM `dokterdata``

Options

nomor	level	nama	username	password
1	1	dr. Fahrudin	fahrudin	amaludin
2	2	dr. Adi	adi	irwanto
3	2	dr. Nirmala	nirmala	hayati
4	2	dr. Afif	afif	viva

Query results operations

Print view Export Display chart Create view

localhost / 127.0.0.1 / pasien / pa

phpMyAdmin

Server: 127.0.0.1 » Database: pasien » Table: pasientdata

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 3 (4 total. Query took 0.0008 seconds) [nomor: 1... - 4...]

`SELECT * FROM `pasientdata` ORDER BY `nomor` ASC`

Options

nomor	nama	jeniskelamin	usia	ruang	penyakit	dokter	tanggalmasuk	tanggalkeluar	biaya
1	Deni	Laki-Laki	21	Melati	DBD	dr. Fahrudin	2019-06-20	NULL	
2	Sudjiman	Laki-Laki	51	Mawar	Diabetes	dr. Adi	2019-06-21	NULL	
3	Fira	Perempuan	15	Dandelion	Kanker Otak	dr. Nirmala	2019-06-20	NULL	
4	Roy	Laki-Laki	19	Melati	Alopecia Areata	dr. Fahrudin	2019-06-21	2019-06-22	800000

Query results operations

Print view Export Display chart Create view

localhost / 127.0.0.1 / pasien / login

phpMyAdmin

Server: 127.0.0.1 » Database: pasien » Table: logindata

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 3 (4 total. Query took 0.0263 seconds.) [username: KAHFI... - ADMIN...]

`SELECT * FROM `logindata` ORDER BY `username` DESC`

[Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

nomor	level	nama	username	password
2	2	Kahfi Aulia	kahfi	aulia
3	2	Firza Nanda	firza	nanda
4	2	Agaphier	aga	redha
1	1	Administrator	admin	admin

Show all | Number of rows: 25 | Filter rows: Search this table

Query results operations

Print view | Export | Display chart | Create view

Console

localhost / 127.0.0.1 / pasien / ruang

phpMyAdmin

Server: 127.0.0.1 » Database: pasien » Table: ruangdata

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 5 (6 total. Query took 0.0007 seconds.)

`SELECT * FROM `ruangdata``

[Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

nomor	ruang
1	Melati
2	Mawar
3	Angrek
4	Dandelion
5	Tulip
6	Kamboja

Show all | Number of rows: 25 | Filter rows: Search this table

Query results operations

Print view | Export | Display chart | Create view

Console