LAPORAN

TUGAS AKHIR

Mata Kuliah Pemrograman Berorientasi Objek

Dosen Pengampu:

M. Bahrul Subkhi, M.Kom

Disusun oleh:

1. Cahyo Septian Nugroho 2213020153
2. Rafael Yonathan Timotius 2213020155
3. Yustitio Caesar 2213020143

PROGRAM STUDI TEKNIK INFORMATIKA

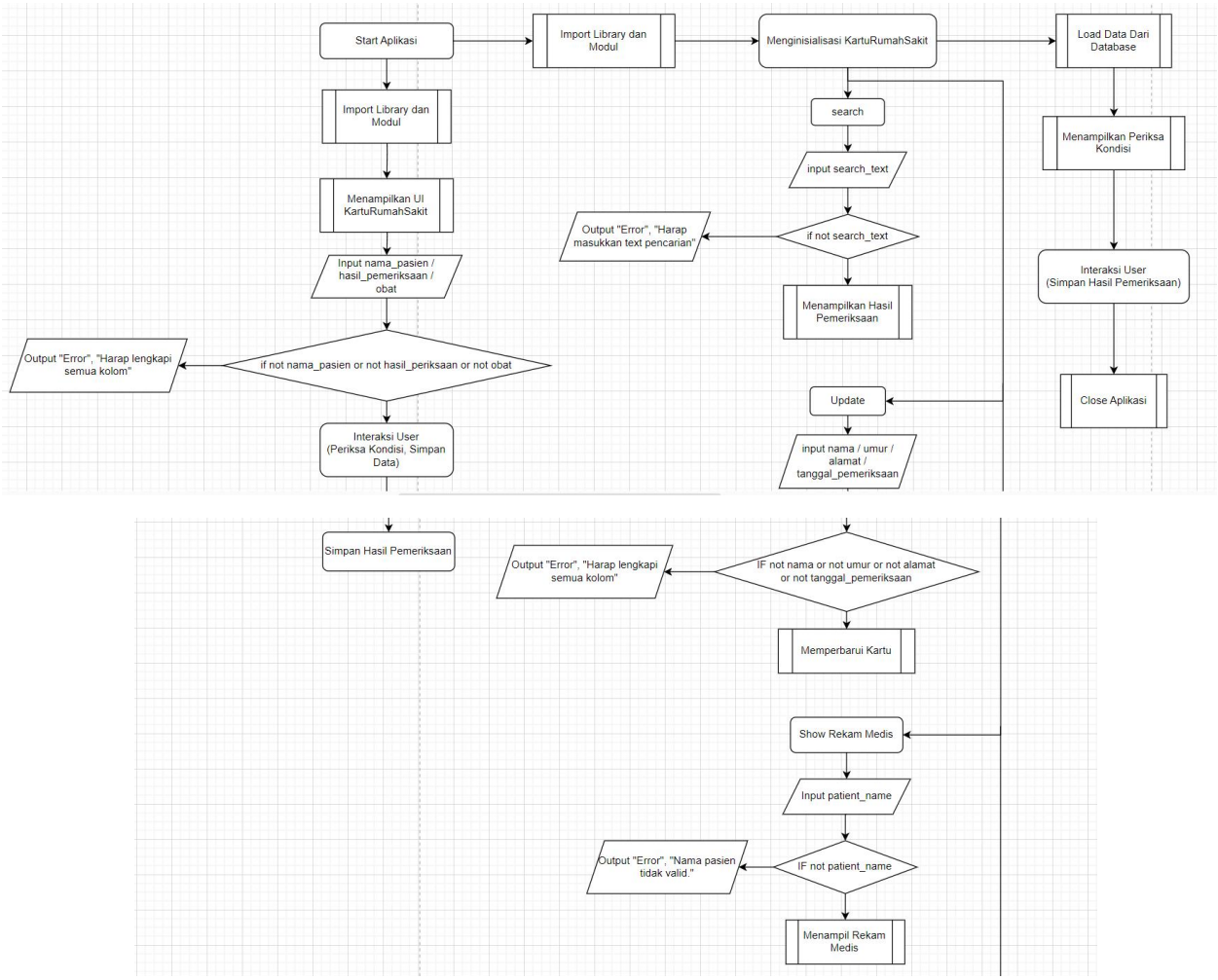
FAKULTAS TEKNIK DAN ILMU KOMPUTER

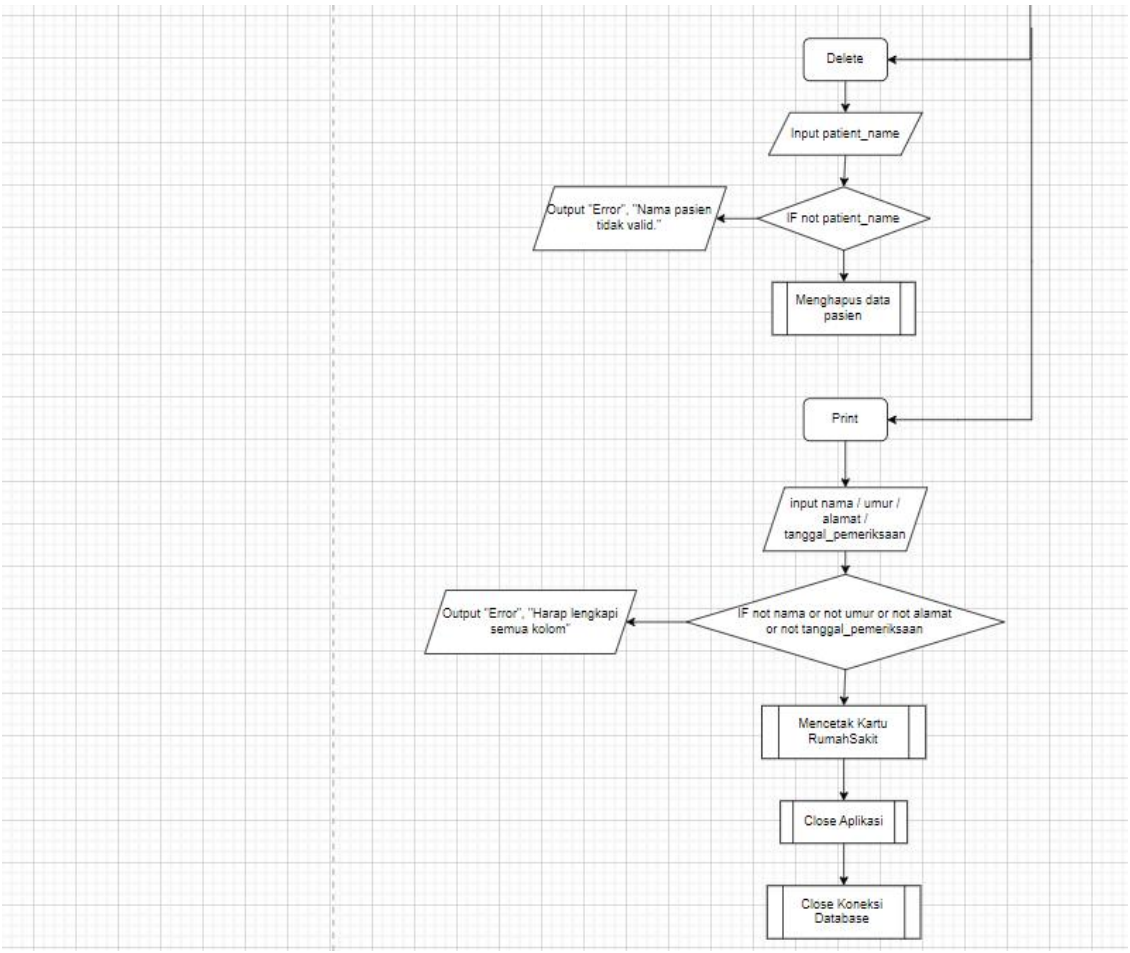
UNIVERSITAS NUSANTARA PGRI KEDIRI

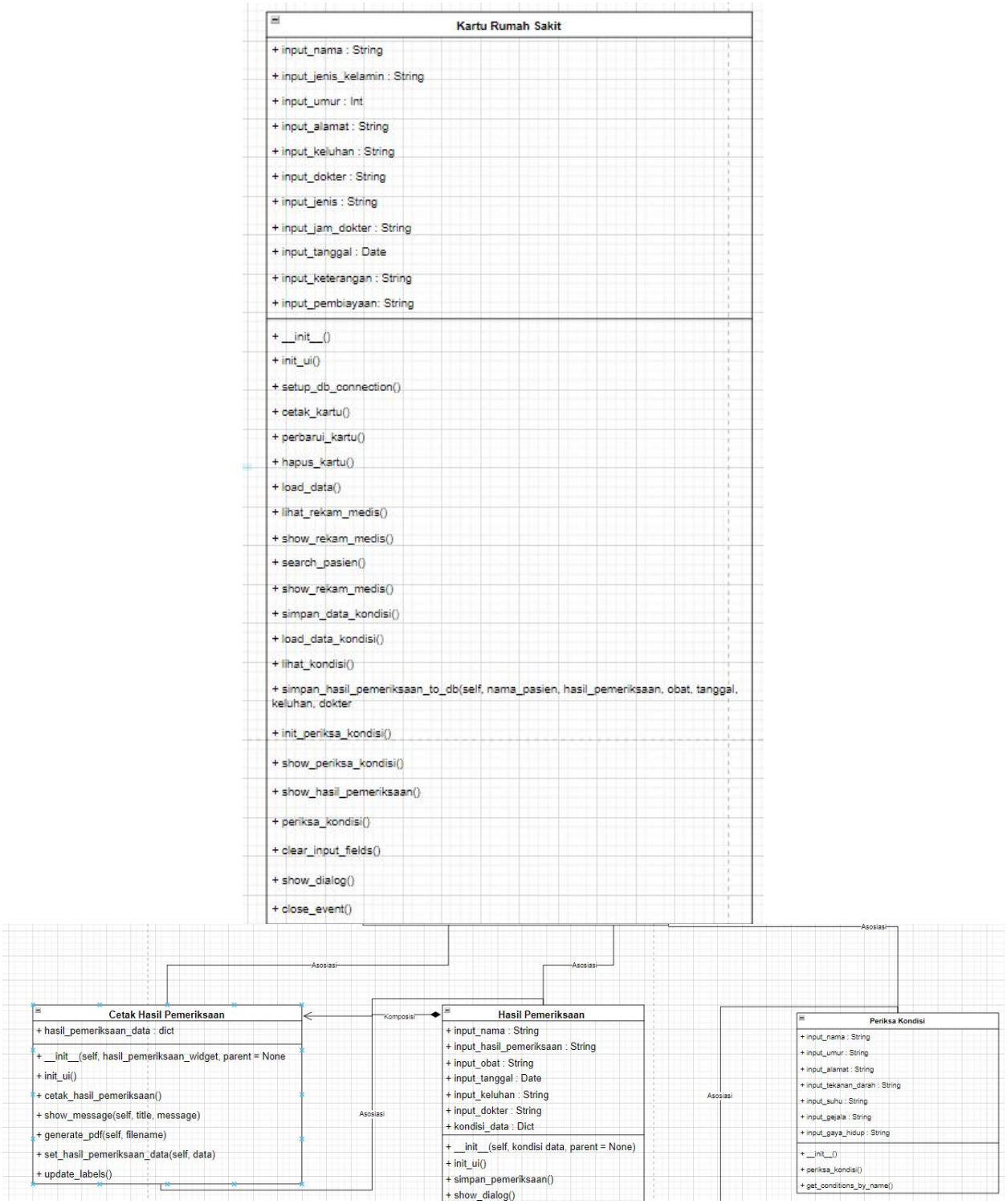
TAHUN 2023

DAFTAR ISI

FLOWCHART SISTEM




Delete
Input patient_name
IF not patient_name
Output "Error", "Nama pasien tidak valid."
Menghapus data pasien
Print
input nama / umur / alamat / tanggal_pemeriksaan
IF not nama or not umur or not alamat or not tanggal_pemeriksaan
Output "Error", "Harap lengkapi semua kolom"
Mencetak Kartu RumahSakit
Close Aplikasi
Close Koneksi Database



CLASS DIAGRAM

HASIL PEMROGRAMAN DAN PENJELASAN

- **Import Library dan Modul**

```
import sys
from PyQt5.QtWidgets import QApplication, QWidget, QLabel, QVBoxLayout, QLineEdit,
QPushButton, QComboBox, QMessageBox, QListWidget, QHBoxLayout, QDialog,
QFormLayout, QGridLayout, QListWidgetItem, QDateEdit,
QFileDialog,  QTableWidgetItem, QTableWidget
from PyQt5.QtCore import pyqtSignal, Qt, QDate
from PyQt5.QtGui import QFont
from PyQt5.QtCore import pyqtSignal, QObject
from reportlab.lib.pagesizes import letter
from reportlab.pdfgen import canvas
from PyQt5 import QtCore, QtGui, QtWidgets
import mysql.connector
```

- **Cetak Hasil Pemeriksaan**

```
class CetakHasilPemeriksaanWidget(QDialog):
    def __init__(self, hasil_pemeriksaan_data, parent=None):
        super(CetakHasilPemeriksaanWidget, self).__init__(parent)

        self.setWindowTitle('Cetak Hasil Pemeriksaan')
        self.setGeometry(600, 300, 400, 300)

        self.hasil_pemeriksaan_data = hasil_pemeriksaan_data
        self.init_ui()
```

- **Trigger button untuk Hasil Pemeriksaan ketika interaksi User**

```
    def init_ui(self):
        label_nama = QLabel(f'Nama Pasien: {self.hasil_pemeriksaan_data["nama"]}')
        label_hasil_pemeriksaan = QLabel(f'Hasil Pemeriksaan:
{self.hasil_pemeriksaan_data["hasil_pemeriksaan"]}')
        label_obat = QLabel(f'Obat yang akan diminum:
{self.hasil_pemeriksaan_data["obat"]}')



        btn_cetak = QPushButton('Cetak', self)
        btn_cetak.clicked.connect(self.cetak_hasil_pemeriksaan)

        layout = QVBoxLayout()
        layout.addWidget(label_nama)
```

```
        layout.addWidget(label_hasil_pemeriksaan)
        layout.addWidget(label_obat)

        layout.addWidget(btn_cetak)

        self.setLayout(layout)
```

- **Logika untuk mencetak hasil pemeriksaan ke PDF**

```
    def cetak_hasil_pemeriksaan(self):
        pdf_filename =
f"hasil_pemeriksaan_{self.hasil_pemeriksaan_data['nama']}.pdf"
        self.generate_pdf(pdf_filename)
        print("Melakukan pencetakan ke PDF...")
        self.accept()

    def show_message(self, title, message):
        msg = QMessageBox(self)
        msg.setWindowTitle(title)
        msg.setText(message)
        msg.exec_()

    def generate_pdf(self, filename):
        try:
            options = QFileDialog.Options()
            options |= QFileDialog.DontUseNativeDialog
            folder_path, _ = QFileDialog.getSaveFileName(self, "Save PDF", "",
"PDF Files (*.pdf);;All Files (*)", options=options)

            if folder_path:
                full_path = f"{folder_path}.pdf"  # Ensure the extension is added

                c = canvas.Canvas(full_path, pagesize=letter)
                c.setFont("Helvetica", 12)

                c.drawString(100, 750, f'Nama Pasien:
{self.hasil_pemeriksaan_data["nama"]}')
                c.drawString(100, 730, f'Hasil Pemeriksaan:
{self.hasil_pemeriksaan_data["hasil_pemeriksaan"]}')
                c.drawString(100, 710, f'Obat yang akan diminum:
{self.hasil_pemeriksaan_data["obat"]}')

                c.save()
                self.show_message("Pencetakan Berhasil", f"Hasil pemeriksaan
telah dicetak ke dalam file PDF: {full_path}")
            else:
```

```
            self.show_message("Batal", "Pencetakan dibatalkan.")
```

- **Pesan yang muncul Ketika terjadi eror saat mencetak**

```
        except Exception as e:
            print(f"Error: {e}")
            self.show_message("Error", f"Error saat mencetak ke PDF: {e}")
```

- **Tampilan Widget untuk Hasil Pemeriksaan**

```
class HasilPemeriksaanWidget(QDialog):
    kondisi_changed = pyqtSignal()
    hasil_pemeriksaan_saved = pyqtSignal(dict)

    def __init__(self, kondisi_data, kartu_rumah_sakit=None):
        super(HasilPemeriksaanWidget, self).__init__()
```

- **Akses Instance KartuRumahsakit**

```
        self.kondisi_data = kondisi_data
        self.kartu_rumah_sakit = kartu_rumah_sakit

        self.setWindowTitle('Hasil Pemeriksaan')
        self.setGeometry(600, 300, 400, 300)
        self.init_ui()


    def init_ui(self):
        label_nama = QLabel('Nama Pasien: ')
        self.input_nama = QLineEdit(self)

        label_hasil_pemeriksaan = QLabel('Hasil Pemeriksaan:')
        self.input_hasil_pemeriksaan = QLineEdit(self)

        label_obat = QLabel('Obat yang akan diminum:')
        self.input_obat = QLineEdit(self)

        label_tanggal = QLabel('Tanggal Pemeriksaan:')
        self.label_tanggal = QDateEdit(self)
        self.label_tanggal.setDate(QDate.currentDate())

        label_keluhan = QLabel('Keluhan:')
        self.input_keluhan = QLineEdit(self)

        nama_dokter = QLabel('Dokter:')
        self.nama_dokter = QComboBox(self)
```

```
        self.nama_dokter.addItems(['Dr. Yasuo, Sp. Jantung', 'Dr. Ahri, Sp. Mata',
'Dr. Riven, Sp. Tulang'])
```

- **Trigger Untuk menyimpan Hasil Pemeriksaan**

```
        btn_simpan_hasil = QPushButton('Simpan Hasil Pemeriksaan', self)
        btn_simpan_hasil.clicked.connect(self.simpan_pemeriksaan)

        layout = QVBoxLayout()
        layout.addWidget(label_nama)
        layout.addWidget(self.input_nama)
        layout.addWidget(label_hasil_pemeriksaan)
        layout.addWidget(self.input_hasil_pemeriksaan)
        layout.addWidget(label_obat)
        layout.addWidget(self.input_obat)
        layout.addWidget(label_tanggal)
        layout.addWidget(self.label_tanggal)
        layout.addWidget(label_keluhan)
        layout.addWidget(self.input_keluhan)
        layout.addWidget(nama_dokter)
        layout.addWidget(self.nama_dokter)
        layout.addWidget(btn_simpan_hasil)

        self.setLayout(layout)
```

- **Logika untuk Menyimpan**

```
    def simpan_pemeriksaan(self):
        nama_pasien = self.input_nama.text()
        hasil_pemeriksaan = self.input_hasil_pemeriksaan.text()
        obat = self.input_obat.text()
        tanggal = self.label_tanggal.text()
        keluhan = self.input_keluhan.text()
        dokter = self.nama_dokter.currentText()


        if not nama_pasien or not hasil_pemeriksaan or not obat or not tanggal or
not keluhan or not dokter:
            self.show_dialog(“Error”, “Harap lengkapi semua kolom.”)
            return
```

- **Menyimpan hasil pemeriksaan dan obat kedalam variable kelas**

```
        self.hasil_pemeriksaan_data = {
            "nama": nama_pasien,
            "hasil_pemeriksaan": hasil_pemeriksaan,
```

```
            "obat": obat,
            "tanggal": tanggal,
            "keluhan": keluhan,
            "dokter": dokter
        }


        # Simpan hasil_pemeriksaan dan obat ke database
        self.kartu_rumah_sakit.simpan_hasil_pemeriksaan_to_db(nama_pasien,
hasil_pemeriksaan, obat, tanggal, keluhan, dokter)

    # Emit sinyal bahwa kondisi telah berubah
        self.kondisi_changed.emit()
        self.hasil_pemeriksaan_saved.emit(self.hasil_pemeriksaan_data)

        # Tutup dialog
        self.accept()


    def show_dialog(self, title, message):
        msg = QMessageBox(self)
        msg.setWindowTitle(title)
        msg.setText(message)
        msg.exec_()
```

- **Widget untuk Periksa Kondisi Pasien**

```
class PeriksaKondisi(QWidget):
    kondisi_checked = pyqtSignal(dict)

    def __init__(self):
        super().__init__()

        self.init_ui()

    def init_ui(self):

        form_layout = QformLayout()
```

- **Pengaturan untuk Layout**

```
        self.setGeometry(550, 300, 900, 300)
        self.setWindowTitle('Periksa Kondisi Pasien')

        label_nama = QLabel('Nama:')
```

```
        label_umur = QLabel('Umur:')
        label_alamat = QLabel('Alamat:')
        label_tekanan_darah = QLabel('Tekanan Darah:')
        label_suhu = QLabel('Suhu:')
        label_gejala = QLabel('Gejala:')
        label_gaya_hidup = QLabel('Gaya Hidup:')


        self.input_nama = QLineEdit(self)
        self.input_umur = QLineEdit(self)
        self.input_alamat = QLineEdit(self)
        self.input_tekanan_darah = QLineEdit(self)
        self.input_suhu = QLineEdit(self)
        self.input_gejala = QLineEdit(self)
        self.input_gaya_hidup = QLineEdit(self)
        btn_periksa_kondisi = QPushButton('Simpan Kondisi', self)
        btn_periksa_kondisi.clicked.connect(self.periksa_kondisi)


        form_layout = QFormLayout()
        form_layout.addRow(label_nama, self.input_nama)
        form_layout.addRow(label_umur, self.input_umur)
        form_layout.addRow(label_alamat, self.input_alamat)
        form_layout.addRow(label_tekanan_darah, self.input_tekanan_darah)
        form_layout.addRow(label_suhu, self.input_suhu)
        form_layout.addRow(label_gejala, self.input_gejala)
        form_layout.addRow(label_gaya_hidup, self.input_gaya_hidup)

        self.pasien_list_periksa_kondisi = QListWidget()

        layout = QVBoxLayout()
        layout.addLayout(form_layout)
        layout.addWidget(self.pasien_list_periksa_kondisi)
        layout.addWidget(btn_periksa_kondisi)
        self.setLayout(layout)


    def periksa_kondisi(self):
        nama = self.input_nama.text()
        umur = self.input_umur.text()
        alamat = self.input_alamat.text()
        tekanan_darah = self.input_tekanan_darah.text()
        suhu = self.input_suhu.text()
        gejala = self.input_gejala.text()
        gaya_hidup = self.input_gaya_hidup.text()
```

```
        data = {
            'nama': nama,
            'umur': umur,
            'alamat': alamat,
            'tekanan_darah': tekanan_darah,
            'suhu': suhu,
            'gejala': gejala,
            'gaya_hidup': gaya_hidup
        }

        self.kondisi_checked.emit({'data': data})
```

- **Mengambil data kondisi**

```
    def get_conditions_by_name(self, patient_name):
        conditions = []

        return conditions
```

- **Widget untuk Kartu Rumah Sakit**

```
class KartuRumahSakit(QObject):
    data_changed = pyqtSignal()
    kondisi_changed = pyqtSignal()
    def __init__(self):
        super().__init__()
        self.setup_db_connection()
        self.setupUi(MainWindow)
        self.pasien_list_kartu_rs = QTableWidget()
        self.load_data()
        self.load_data_kondisi()
        self.kondisi_data = {}
```

- **Setting tampilan**

```
    def setupUi(self, MainWindow):
        MainWindow.setObjectName("Kartu Rumah Sakit")
        MainWindow.resize(890, 590)
        MainWindow.setStyleSheet("QWidget {\n"
"                background-color: #f0f0f0; /* Set the background color */\n"
"            }")

        self.pasien_list_periksa_kondisi = QTableWidget()
        self.kondisi_changed.connect(self.load_data_kondisi)
        self.input_nama = QLineEdit()
```

```
        self.input_umur = QLineEdit()
        self.input_alamat = QLineEdit()
        self.input_keluhan = QLineEdit()
        self.jenis_kelamin = QComboBox()
        self.cmb_pembiayaan = QComboBox()
        self.dokter = QComboBox()
        self.jenis = QComboBox()
        self.jam = QComboBox()
        self.input_tanggal = QDateEdit()
        self.keterangan = QComboBox()

        self.pasien_list_kartu_rs = QTableWidget()
        self.centralwidget = QtWidgets.QWidget(MainWindow)
        self.centralwidget.setObjectName("centralwidget")
        self.formLayoutWidget = QtWidgets.QWidget(self.centralwidget)
        self.formLayoutWidget.setGeometry(QtCore.QRect(10, 10, 471, 341))
        self.formLayoutWidget.setObjectName("formLayoutWidget")
        self.formLayout = QtWidgets.QFormLayout(self.formLayoutWidget)
        self.formLayout.setContentsMargins(0, 0, 0, 0)
        self.formLayout.setObjectName("formLayout")
        self.namaLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.namaLabel.setFont(font)
        self.namaLabel.setObjectName("namaLabel")
        self.formLayout.setWidget(0, QtWidgets.QFormLayout.LabelRole,
self.namaLabel)
        self.namaLineEdit = QtWidgets.QLineEdit(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.namaLineEdit.setFont(font)
        self.namaLineEdit.setText("")
        self.namaLineEdit.setObjectName("namaLineEdit")
        self.formLayout.setWidget(0, QtWidgets.QFormLayout.FieldRole,
self.namaLineEdit)
        self.umurLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.umurLabel.setFont(font)
        self.umurLabel.setObjectName("umurLabel")
        self.formLayout.setWidget(1, QtWidgets.QFormLayout.LabelRole,
self.umurLabel)
        self.umurLineEdit = QtWidgets.QLineEdit(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
```

```
        self.umurLineEdit.setFont(font)
        self.umurLineEdit.setToolTip("")
        self.umurLineEdit.setText("")
        self.umurLineEdit.setObjectName("umurLineEdit")
        self.formLayout.setWidget(1, QtWidgets.QFormLayout.FieldRole,
self.umurLineEdit)
        self.jenisKelaminLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.jenisKelaminLabel.setFont(font)
        self.jenisKelaminLabel.setObjectName("jenisKelaminLabel")
        self.formLayout.setWidget(2, QtWidgets.QFormLayout.LabelRole,
self.jenisKelaminLabel)
        self.alamatLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.alamatLabel.setFont(font)
        self.alamatLabel.setObjectName("alamatLabel")
        self.formLayout.setWidget(3, QtWidgets.QFormLayout.LabelRole,
self.alamatLabel)
        self.alamatLineEdit = QtWidgets.QLineEdit(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.alamatLineEdit.setFont(font)
        self.alamatLineEdit.setText("")
        self.alamatLineEdit.setObjectName("alamatLineEdit")
        self.formLayout.setWidget(3, QtWidgets.QFormLayout.FieldRole,
self.alamatLineEdit)
        self.keluhanLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.keluhanLabel.setFont(font)
        self.keluhanLabel.setObjectName("keluhanLabel")
        self.formLayout.setWidget(4, QtWidgets.QFormLayout.LabelRole,
self.keluhanLabel)
        self.keluhanLineEdit = QtWidgets.QLineEdit(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.keluhanLineEdit.setFont(font)
        self.keluhanLineEdit.setText("")
        self.keluhanLineEdit.setObjectName("keluhanLineEdit")
        self.formLayout.setWidget(4, QtWidgets.QFormLayout.FieldRole,
self.keluhanLineEdit)
        self.pembiayaanLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
```

```
        font.setPointSize(12)
        self.pembiayaanLabel.setFont(font)
        self.pembiayaanLabel.setObjectName("pembiayaanLabel")
        self.formLayout.setWidget(5, QtWidgets.QFormLayout.LabelRole,
self.pembiayaanLabel)
        self.dokterLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.dokterLabel.setFont(font)
        self.dokterLabel.setObjectName("dokterLabel")
        self.formLayout.setWidget(6, QtWidgets.QFormLayout.LabelRole,
self.dokterLabel)
        self.jenisLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.jenisLabel.setFont(font)
        self.jenisLabel.setObjectName("jenisLabel")
        self.formLayout.setWidget(7, QtWidgets.QFormLayout.LabelRole,
self.jenisLabel)
        self.jamKerjaDokterLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.jamKerjaDokterLabel.setFont(font)
        self.jamKerjaDokterLabel.setObjectName("jamKerjaDokterLabel")
        self.formLayout.setWidget(8, QtWidgets.QFormLayout.LabelRole,
self.jamKerjaDokterLabel)
        self.keteranganLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.keteranganLabel.setFont(font)
        self.keteranganLabel.setObjectName("keteranganLabel")
        self.formLayout.setWidget(9, QtWidgets.QFormLayout.LabelRole,
self.keteranganLabel)
        self.tanggalPemeriksaanLabel = QtWidgets.QLabel(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.tanggalPemeriksaanLabel.setFont(font)
        self.tanggalPemeriksaanLabel.setObjectName("tanggalPemeriksaanLabel")
        self.formLayout.setWidget(10, QtWidgets.QFormLayout.LabelRole,
self.tanggalPemeriksaanLabel)
        self.comboBox = QtWidgets.QComboBox(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.comboBox.setFont(font)
        self.comboBox.setObjectName("comboBox")
```

```
        self.comboBox.addItem("")
        self.comboBox.addItem("")
        self.formLayout.setWidget(2, QtWidgets.QFormLayout.FieldRole,
self.comboBox)
        self.comboBox_2 = QtWidgets.QComboBox(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.comboBox_2.setFont(font)
        self.comboBox_2.setObjectName("comboBox_2")
        self.comboBox_2.addItem("")
        self.comboBox_2.addItem("")
        self.formLayout.setWidget(5, QtWidgets.QFormLayout.FieldRole,
self.comboBox_2)
        self.comboBox_3 = QtWidgets.QComboBox(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.comboBox_3.setFont(font)
        self.comboBox_3.setObjectName("comboBox_3")
        self.comboBox_3.addItem("")
        self.comboBox_3.addItem("")
        self.comboBox_3.addItem("")
        self.formLayout.setWidget(6, QtWidgets.QFormLayout.FieldRole,
self.comboBox_3)
        self.comboBox_4 = QtWidgets.QComboBox(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.comboBox_4.setFont(font)
        self.comboBox_4.setObjectName("comboBox_4")
        self.comboBox_4.addItem("")
        self.comboBox_4.addItem("")
        self.formLayout.setWidget(7, QtWidgets.QFormLayout.FieldRole,
self.comboBox_4)
        self.comboBox_5 = QtWidgets.QComboBox(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.comboBox_5.setFont(font)
        self.comboBox_5.setObjectName("comboBox_5")
        self.comboBox_5.addItem("")
        self.comboBox_5.addItem("")
        self.comboBox_5.addItem("")
        self.formLayout.setWidget(8, QtWidgets.QFormLayout.FieldRole,
self.comboBox_5)
        self.dateEdit = QtWidgets.QDateEdit(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
```

```
        self.dateEdit.setFont(font)
        self.dateEdit.setObjectName("dateEdit")
        self.formLayout.setWidget(9, QtWidgets.QFormLayout.FieldRole,
self.dateEdit)
        self.comboBox_6 = QtWidgets.QComboBox(self.formLayoutWidget)
        font = QtGui.QFont()
        font.setPointSize(12)
        self.comboBox_6.setFont(font)
        self.comboBox_6.setObjectName("comboBox_6")
        self.comboBox_6.addItem("")
        self.comboBox_6.addItem("")
        self.comboBox_6.addItem("")
        self.formLayout.setWidget(10, QtWidgets.QFormLayout.FieldRole,
self.comboBox_6)
        self.tableWidget = QtWidgets.QTableWidget(self.centralwidget)
        self.tableWidget.setGeometry(QtCore.QRect(10, 350, 871, 192))
        self.tableWidget.setBaseSize(QtCore.QSize(10, 3))
        font = QtGui.QFont()
        font.setPointSize(8)
        self.tableWidget.setFont(font)
        self.tableWidget.setAutoScrollMargin(16)
        self.tableWidget.setObjectName("tableWidget")
        self.tableWidget.setColumnCount(11)
        self.tableWidget.setRowCount(1)
        item = QtWidgets.QTableWidgetItem()
        self.tableWidget.setVerticalHeaderItem(0, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(0, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(1, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(2, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(3, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(4, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(5, item)
        item = QtWidgets.QTableWidgetItem()
```

```
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(6, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(7, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(8, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        font = QtGui.QFont()
        font.setPointSize(8)
        item.setFont(font)
        self.tableWidget.setHorizontalHeaderItem(9, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setHorizontalHeaderItem(10, item)
        item = QtWidgets.QTableWidgetItem()
        font = QtGui.QFont()
        font.setPointSize(8)
        item.setFont(font)
        self.tableWidget.setItem(0, 0, item)
        item = QtWidgets.QTableWidgetItem()
        font = QtGui.QFont()
        font.setPointSize(8)
        item.setFont(font)
        self.tableWidget.setItem(0, 1, item)
        item = QtWidgets.QTableWidgetItem()
        self.tableWidget.setItem(0, 2, item)
        item = QtWidgets.QTableWidgetItem()
        self.tableWidget.setItem(0, 3, item)
        item = QtWidgets.QTableWidgetItem()
        self.tableWidget.setItem(0, 4, item)
        item = QtWidgets.QTableWidgetItem()
        self.tableWidget.setItem(0, 5, item)
        item = QtWidgets.QTableWidgetItem()
        self.tableWidget.setItem(0, 6, item)
        item = QtWidgets.QTableWidgetItem()
        self.tableWidget.setItem(0, 7, item)
        item = QtWidgets.QTableWidgetItem()
        self.tableWidget.setItem(0, 8, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        font = QtGui.QFont()
        font.setPointSize(8)
```

```
        item.setFont(font)
        self.tableWidget.setItem(0, 9, item)
        item = QtWidgets.QTableWidgetItem()
        item.setTextAlignment(QtCore.Qt.AlignCenter)
        self.tableWidget.setItem(0, 10, item)
        self.pushButton = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton.setGeometry(QtCore.QRect(490, 50, 381, 41))
        font = QtGui.QFont()
        font.setPointSize(-1)
        font.setUnderline(False)
        font.setStrikeOut(False)
        self.pushButton.setFont(font)
        self.pushButton.setLayoutDirection(QtCore.Qt.LeftToRight)
```

- **Mengatur Align Layout dan background**

```
        self.pushButton.setStyleSheet("QPushButton {\n"
"                background-color: #4CAF50; /* Green background color */\n"
"                color: white;\n"
"                border: none;\n"
"                padding: 8px 16px;\n"
"                text-align: center;\n"
"                text-decoration: none;\n"
"                font-size: 12px;\n"
"                margin: 4px 2px;\n"
"                border-radius: 4px; /* Rounded corners */\n"
"            }\n"
"")
        self.pushButton.setAutoDefault(False)
        self.pushButton.setFlat(False)
        self.pushButton.setObjectName("pushButton")
        self.pushButton.clicked.connect(self.cetak_kartu)
        self.pushButton_2 = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton_2.setGeometry(QtCore.QRect(490, 100, 381, 41))
        font = QtGui.QFont()
        font.setPointSize(-1)
        font.setUnderline(False)
        font.setStrikeOut(False)
        self.pushButton_2.setFont(font)
        self.pushButton_2.setLayoutDirection(QtCore.Qt.LeftToRight)
        self.pushButton_2.setStyleSheet("QPushButton {\n"
"                background-color: #4CAF50; /* Green background color */\n"
"                color: white;\n"
"                border: none;\n"
"                padding: 8px 16px;\n"
```

```
"                text-align: center;\n"
"                text-decoration: none;\n"
"                font-size: 12px;\n"
"                margin: 4px 2px;\n"
"                border-radius: 4px; /* Rounded corners */\n"
"            }\n"
"")
        self.pushButton_2.setAutoDefault(False)
        self.pushButton_2.setFlat(False)
        self.pushButton_2.setObjectName("pushButton_2")
        self.pushButton_2.clicked.connect(self.perbarui_kartu)
        self.pushButton_3 = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton_3.setGeometry(QtCore.QRect(490, 150, 381, 41))
        font = QtGui.QFont()
        font.setPointSize(-1)
        font.setUnderline(False)
        font.setStrikeOut(False)
        self.pushButton_3.setFont(font)
        self.pushButton_3.setLayoutDirection(QtCore.Qt.LeftToRight)
        self.pushButton_3.setStyleSheet("QPushButton {\n"
"                background-color: #4CAF50; /* Green background color */\n"
"                color: white;\n"
"                border: none;\n"
"                padding: 8px 16px;\n"
"                text-align: center;\n"
"                text-decoration: none;\n"
"                font-size: 12px;\n"
"                margin: 4px 2px;\n"
"                border-radius: 4px; /* Rounded corners */\n"
"            }\n"
"")
        self.pushButton_3.setAutoDefault(False)
        self.pushButton_3.setFlat(False)
        self.pushButton_3.setObjectName("pushButton_3")
        self.pushButton_3.clicked.connect(self.hapus_kartu)
        self.pushButton_4 = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton_4.setGeometry(QtCore.QRect(490, 200, 381, 41))
        font = QtGui.QFont()
        font.setPointSize(-1)
        font.setUnderline(False)
        font.setStrikeOut(False)
        self.pushButton_4.setFont(font)
        self.pushButton_4.setLayoutDirection(QtCore.Qt.LeftToRight)
        self.pushButton_4.setStyleSheet("QPushButton {\n"
"                background-color: #4CAF50; /* Green background color */\n"
```

```
"                color: white;\n"
"                border: none;\n"
"                padding: 8px 16px;\n"
"                text-align: center;\n"
"                text-decoration: none;\n"
"                font-size: 12px;\n"
"                margin: 4px 2px;\n"
"                border-radius: 4px; /* Rounded corners */\n"
"            }\n"
"")
        self.pushButton_4.setAutoDefault(False)
        self.pushButton_4.setFlat(False)
        self.pushButton_4.setObjectName("pushButton_4")
        self.pushButton_4.clicked.connect(self.init_periksa_kondisi)
        self.pushButton_5 = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton_5.setGeometry(QtCore.QRect(490, 250, 381, 41))
        font = QtGui.QFont()
        font.setPointSize(-1)
        font.setUnderline(False)
        font.setStrikeOut(False)
        self.pushButton_5.setFont(font)
        self.pushButton_5.setLayoutDirection(QtCore.Qt.LeftToRight)
        self.pushButton_5.setStyleSheet("QPushButton {\n"
"                background-color: #4CAF50; /* Green background color */\n"
"                color: white;\n"
"                border: none;\n"
"                padding: 8px 16px;\n"
"                text-align: center;\n"
"                text-decoration: none;\n"
"                font-size: 12px;\n"
"                margin: 4px 2px;\n"
"                border-radius: 4px; /* Rounded corners */\n"
"            }\n"
"")
        self.pushButton_5.setAutoDefault(False)
        self.pushButton_5.setFlat(False)
        self.pushButton_5.setObjectName("pushButton_5")
        self.pushButton_5.clicked.connect(self.lihat_kondisi)
        self.pushButton_6 = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton_6.setGeometry(QtCore.QRect(490, 300, 381, 41))
        font = QtGui.QFont()
        font.setPointSize(-1)
        font.setUnderline(False)
        font.setStrikeOut(False)
        self.pushButton_6.setFont(font)
```

```
        self.pushButton_6.setLayoutDirection(QtCore.Qt.LeftToRight)
        self.pushButton_6.setStyleSheet("QPushButton {\n"
"                background-color: #4CAF50; /* Green background color */\n"
"                color: white;\n"
"                border: none;\n"
"                padding: 8px 16px;\n"
"                text-align: center;\n"
"                text-decoration: none;\n"
"                font-size: 12px;\n"
"                margin: 4px 2px;\n"
"                border-radius: 4px; /* Rounded corners */\n"
"            }\n"
"")
        self.pushButton_6.setAutoDefault(False)
        self.pushButton_6.setFlat(False)
        self.pushButton_6.setObjectName("pushButton_6")
        self.pushButton_6.clicked.connect(self.show_hasil_pemeriksaan)
        self.textEdit = QLineEdit(self.centralwidget)
        self.textEdit.setGeometry(QtCore.QRect(490, 10, 381, 31))
        palette = QtGui.QPalette()
        brush = QtGui.QBrush(QtGui.QColor(148, 148, 148))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.WindowText, brush)
        brush = QtGui.QBrush(QtGui.QColor(240, 240, 240))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Button, brush)
        brush = QtGui.QBrush(QtGui.QColor(240, 240, 240))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Base, brush)
        brush = QtGui.QBrush(QtGui.QColor(240, 240, 240))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Active, QtGui.QPalette.Window, brush)
        brush = QtGui.QBrush(QtGui.QColor(148, 148, 148))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.WindowText,
brush)
        brush = QtGui.QBrush(QtGui.QColor(240, 240, 240))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Button, brush)
        brush = QtGui.QBrush(QtGui.QColor(240, 240, 240))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Base, brush)
        brush = QtGui.QBrush(QtGui.QColor(240, 240, 240))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Inactive, QtGui.QPalette.Window, brush)
```

```
        brush = QtGui.QBrush(QtGui.QColor(120, 120, 120))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.WindowText,
brush)
        brush = QtGui.QBrush(QtGui.QColor(240, 240, 240))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Button, brush)
        brush = QtGui.QBrush(QtGui.QColor(240, 240, 240))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Base, brush)
        brush = QtGui.QBrush(QtGui.QColor(240, 240, 240))
        brush.setStyle(QtCore.Qt.SolidPattern)
        palette.setBrush(QtGui.QPalette.Disabled, QtGui.QPalette.Window, brush)
        self.textEdit.setPalette(palette)
        self.textEdit.setObjectName("textEdit")
        self.textEdit.setPlaceholderText('Cari berdasarkan Nama / Tanggal
Pemeriksaan')
        self.textEdit.returnPressed.connect(self.search_pasien)

        MainWindow.setCentralWidget(self.centralwidget)
        self.menubar = QtWidgets.QMenuBar(MainWindow)
        self.menubar.setGeometry(QtCore.QRect(0, 0, 890, 21))
        self.menubar.setObjectName("menubar")
        MainWindow.setMenuBar(self.menubar)
        self.statusbar = QtWidgets.QStatusBar(MainWindow)
        self.statusbar.setObjectName("statusbar")
        MainWindow.setStatusBar(self.statusbar)


        self.retranslateUi(MainWindow)
        QtCore.QMetaObject.connectSlotsByName(MainWindow)

    def retranslateUi(self, MainWindow):
        _translate = QtCore.QCoreApplication.translate
        MainWindow.setWindowTitle(_translate("MainWindow", "MainWindow"))
        self.namaLabel.setText(_translate("MainWindow", "Nama "))
        self.umurLabel.setText(_translate("MainWindow", "Umur"))
        self.jenisKelaminLabel.setText(_translate("MainWindow", "Jenis Kelamin"))
        self.alamatLabel.setText(_translate("MainWindow", "Alamat"))
        self.keluhanLabel.setText(_translate("MainWindow", "Keluhan"))
        self.pembiayaanLabel.setText(_translate("MainWindow", "Pembiayaan"))
        self.dokterLabel.setText(_translate("MainWindow", "Dokter"))
        self.jenisLabel.setText(_translate("MainWindow", "Jenis"))
        self.jamKerjaDokterLabel.setText(_translate("MainWindow", "Pilih Jam
Dokter"))
```

```
        self.keteranganLabel.setText(_translate("MainWindow", "Tanggal
Pemeriksaan"))
        self.tanggalPemeriksaanLabel.setText(_translate("MainWindow",
"Keterangan"))
        self.comboBox.setItemText(0, _translate("MainWindow", "Laki - Laki"))
        self.comboBox.setItemText(1, _translate("MainWindow", "Perempuan"))
        self.comboBox_2.setItemText(0, _translate("MainWindow", "BPJS"))
        self.comboBox_2.setItemText(1, _translate("MainWindow", "Umum"))
        self.comboBox_3.setItemText(0, _translate("MainWindow", "Dr. Yasuo, Sp.
Jantung"))
        self.comboBox_3.setItemText(1, _translate("MainWindow", "Dr. Ahri, Sp.
Mata"))
        self.comboBox_3.setItemText(2, _translate("MainWindow", "Dr. Riven, Sp.
Tulang"))
        self.comboBox_4.setItemText(0, _translate("MainWindow", "Baru Daftar"))
        self.comboBox_4.setItemText(1, _translate("MainWindow", "Sudah Pernah
Daftar"))
        self.comboBox_5.setItemText(0, _translate("MainWindow", "07 : 00"))
        self.comboBox_5.setItemText(1, _translate("MainWindow", "09 : 00"))
        self.comboBox_5.setItemText(2, _translate("MainWindow", "10 : 00"))
        self.comboBox_6.setItemText(0, _translate("MainWindow", "Baru Daftar
Kartu Rumah Sakit"))
        self.comboBox_6.setItemText(1, _translate("MainWindow", "Kartu Rumah
Sakit Hilang"))
        self.comboBox_6.setItemText(2, _translate("MainWindow", "Kartu Rumah
Sakit Rusak"))
        item = self.tableWidget.verticalHeaderItem(0)
        item.setText(_translate("MainWindow", "1"))
        item = self.tableWidget.horizontalHeaderItem(0)
        item.setText(_translate("MainWindow", "Nama"))
        item = self.tableWidget.horizontalHeaderItem(1)
        item.setText(_translate("MainWindow", "Umur"))
        item = self.tableWidget.horizontalHeaderItem(2)
        item.setText(_translate("MainWindow", "Jenis Kelamin"))
        item = self.tableWidget.horizontalHeaderItem(3)
        item.setText(_translate("MainWindow", "Alamat"))
        item = self.tableWidget.horizontalHeaderItem(4)
        item.setText(_translate("MainWindow", "Keluhan"))
        item = self.tableWidget.horizontalHeaderItem(5)
        item.setText(_translate("MainWindow", "Pembiayaan"))
        item = self.tableWidget.horizontalHeaderItem(6)
        item.setText(_translate("MainWindow", "Dokter"))
        item = self.tableWidget.horizontalHeaderItem(7)
        item.setText(_translate("MainWindow", "Jenis"))
        item = self.tableWidget.horizontalHeaderItem(8)
```

```
        item.setText(_translate("MainWindow", "Pilih Jam Dokter"))
        item = self.tableWidget.horizontalHeaderItem(9)
        item.setText(_translate("MainWindow", "Tanggal Pemeriksaan"))
        item = self.tableWidget.horizontalHeaderItem(10)
        item.setText(_translate("MainWindow", "Keterangan"))
        __sortingEnabled = self.tableWidget.isSortingEnabled()
        self.tableWidget.setSortingEnabled(False)
        self.tableWidget.setSortingEnabled(__sortingEnabled)
        self.pushButton.setText(_translate("MainWindow", "Buat Kartu Rumah
Sakit"))
        self.pushButton_2.setText(_translate("MainWindow", "Perbarui Kartu Rumah
Sakit"))
        self.pushButton_3.setText(_translate("MainWindow", "Delete Kartu Rumah
Sakit"))
        self.pushButton_4.setText(_translate("MainWindow", "Periksa Kondisi"))
        self.pushButton_5.setText(_translate("MainWindow", "Lihat Kondisi"))
        self.pushButton_6.setText(_translate("MainWindow", "Hasil Pemeriksaan"))
```

- **Settup koneksi ke DataBase kartu_rs**

```
    def setup_db_connection(self):
        try:
            self.db_connection = mysql.connector.connect(
                host="localhost",
                user="root",
                password="rafael$kuning",
                database="kartu_rs"
            )
```

- **Pesan yang muncul Ketika terjadi eror saat konek ke DB**

```
        except mysql.connector.Error as err:
            print(f"Error connecting to the database: {err}")
            sys.exit(1)

    def cetak_kartu(self):
        nama = self.namaLineEdit.text()
        umur = self.umurLineEdit.text()
        jenis_kelamin = self.comboBox.currentText()
        alamat = self.alamatLineEdit.text()
        keluhan = self.keluhanLineEdit.text()
        pembiayaan = self.comboBox_2.currentText()
        dokter = self.comboBox_3.currentText()
        jenis = self.comboBox_4.currentText()
        jam = self.comboBox_5.currentText()
        keterangan = self.comboBox_6.currentText()
        tanggal_pemeriksaan = self.dateEdit.text()
```

```
        font = QFont()
        font.setPointSize(14)


        kartu_teks = f"Nama: {nama}\nUmur: {umur}\nJenis Kelamin:
{jenis_kelamin}\nAlamat: {alamat}\nKeluhan: {keluhan}\nPembiayaan:
{pembiayaan}\nDokter: {dokter}\nJenis: {jenis}\nJam Kerja Dokter:
{jam}\nKeterangan {keterangan}\nTangal Pemeriksaan: {tanggal_pemeriksaan}"

        self.show_dialog('KartuRumahSakit', kartu_teks)

        try:
            cursor = self.db_connection.cursor()
            sql = "INSERT INTO informasi (nama, umur, keluhan, pembiayaan, dokter,
jenis, jam_kerja_dokter, keterangan,jenis_kelamin, alamat, tanggal_pemeriksaan)
VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)"
            values = (nama, umur, keluhan, pembiayaan, dokter, jenis, jam,
keterangan, jenis_kelamin, alamat, tanggal_pemeriksaan)
            cursor.execute(sql, values)
            self.db_connection.commit()
            cursor.close()
            self.show_dialog("Kartu Rumah Sakit", kartu_teks)
        except mysql.connector.Error as err:
            print(f"Error: {err}")
            self.show_dialog("Error", f"Error: {err}")
            return

        self.clear_input_fields()
        self.data_changed.emit()
        self.load_data()
```

- **Membuat Fungsi Update**

```
    def perbarui_kartu(self):
        nama = self.namaLineEdit.text()
        umur = self.umurLineEdit.text()
        jenis_kelamin = self.comboBox.currentText()
        alamat = self.alamatLineEdit.text()
        keluhan = self.keluhanLineEdit.text()
        pembiayaan = self.comboBox_2.currentText()
        dokter = self.comboBox_3.currentText()
        jenis = self.comboBox_4.currentText()
        jam = self.comboBox_5.currentText()
        keterangan = self.comboBox_6.currentText()
```

```
        tanggal_pemeriksaan = self.dateEdit.text()

        try:
            cursor = self.db_connection.cursor()
            sql = "UPDATE informasi SET umur=%s, keluhan=%s, pembiayaan=%s,
dokter=%s, jenis=%s, jam_kerja_dokter=%s, keterangan=%s, jenis_kelamin = %s,
alamat = %s, tanggal_pemeriksaan = %s WHERE nama=%s"
            values = (umur, keluhan, pembiayaan, dokter, jenis, jam,
keterangan,jenis_kelamin, alamat, tanggal_pemeriksaan, nama)
            cursor.execute(sql, values)
            self.db_connection.commit()
            cursor.close()
            self.show_dialog("Perbarui Kartu Rumah Sakit", "Data berhasil
diperbarui.")
        except mysql.connector.Error as err:
            print(f"Error: {err}")
            self.show_dialog("Error", f"Error: {err}")
            return

        self.load_data()
        self.clear_input_fields()
        self.data_changed.emit()
```

- **Membuat Fungsi Delete**

```
    def hapus_kartu(self):
        try:
            cursor = self.db_connection.cursor()
            sql = "DELETE FROM informasi"
            cursor.execute(sql)
            self.db_connection.commit()
            cursor.close()
            self.show_dialog("Hapus Semua Data", "Semua data berhasil dihapus.")
            self.data_changed.emit()
            self.load_data()
        except mysql.connector.Error as err:
            print(f"Error: {err}")
            self.show_dialog("Error", f"Error: {err}")
```

- **Membuat Fungsi Load data**

```
    def load_data(self):
        try:
            cursor = self.db_connection.cursor()
            sql = "SELECT nama, umur, jenis_kelamin, alamat, keluhan, pembiayaan,
dokter, jenis, jam_kerja_dokter, keterangan, tanggal_pemeriksaan FROM informasi"
```

```
            cursor.execute(sql)
            result = cursor.fetchall()
            cursor.close()

            self.tableWidget.setRowCount(0)  # Clear existing data

            # Mengisi tabel dengan data
            for row in result:
                rowPosition = self.tableWidget.rowCount()
                self.tableWidget.insertRow(rowPosition)

                for col, value in enumerate(row):
                    item = QTableWidgetItem(str(value))
                    self.tableWidget.setItem(rowPosition, col, item)

        except mysql.connector.Error as err:
            print(f"Error: {err}")
            self.show_dialog("Error", f"Error: {err}")
```

- **Membuat Fungsi Simpan / Save**

```
    def simpan_data_kondisi(self, data):
        try:
            cursor = self.db_connection.cursor()
            sql = "INSERT INTO kondisi_pasien (nama_pasien, umur, alamat,
tekanan_darah, suhu, gejala, gaya_hidup) VALUES (%s, %s, %s, %s, %s, %s, %s)"
            values = (
                data['data']['nama'],
                data['data']['umur'],
                data['data']['alamat'],
                data['data']['tekanan_darah'],
                data['data']['suhu'],
                data['data']['gejala'],
                data['data']['gaya_hidup']
            )
            cursor.execute(sql, values)
            self.db_connection.commit()
            cursor.close()
            self.show_dialog("Data Kondisi", "Data kondisi berhasil disimpan.")
            self.kondisi_changed.emit()
        except mysql.connector.Error as err:
            print(f"Error: {err}")
            self.show_dialog("Error", f"Error: {err}")
            self.load_data_kondisi()
```

```
    def load_data_kondisi(self):

        try:
            self.pasien_list_periksa_kondisi = QTableWidget()
            cursor = self.db_connection.cursor()
            sql = "SELECT nama_pasien, umur, alamat, tekanan_darah, suhu, gejala,
gaya_hidup FROM kondisi_pasien"
            cursor.execute(sql)
            result = cursor.fetchall()
            cursor.close()

        # Clear existing items
            self.pasien_list_periksa_kondisi.clear()

        # Set up the table widget
            self.pasien_list_periksa_kondisi.setRowCount(len(result))
            self.pasien_list_periksa_kondisi.setColumnCount(7)
            self.pasien_list_periksa_kondisi.setHorizontalHeaderLabels(["Nama",
"Umur", "Alamat", "Tekanan Darah", "Suhu", "Gejala", "Gaya Hidup"])

        # Add new items to the table widget
            for row_index, row in enumerate(result):
                for col_index, col_value in enumerate(row):
                    item = QTableWidgetItem(str(col_value))
                    self.pasien_list_periksa_kondisi.setItem(row_index, col_index,
item)

        except mysql.connector.Error as err:
            print(f"Error: {err}")
            self.show_dialog("Error", f"Error: {err}")
```

- **Membuat Fungsi Search**

```
    def search_pasien(self):
        search_text = self.textEdit.text()
        try:
            cursor = self.db_connection.cursor()
            sql = “SELECT nama, umur, jenis_kelamin,  lamat, keluhan, pembiayaan,
dokter, jenis, jam_kerja_dokter, tanggal_pemeriksaan, keterangan FROM informasi
WHERE nama LIKE %s OR tanggal_pemeriksaan LIKE %s”
            cursor.execute(sql, (f’%{search_text}%’, f’%{search_text}%’))
            result = cursor.fetchall()
            cursor.close()

            self.display_search_result(result)
```

- **Pesan eror Ketika isi able tidak sesuai**

```
        except mysql.connector.Error as err:
            print(f"Error: {err}")
            self.show_dialog("Error", f"Error: {err}")

    def display_search_result(self, result):
        search_result_dialog = Qdialog()
        search_result_dialog.setWindowTitle("Pasien")

        layout = QVBoxLayout(search_result_dialog)

        # Create a table widget
        table = QtableWidget()
        table.setColumnCount(11)  # Number of columns

        # Set table headers
        headers = ["Nama", "Umur", "Jenis Kelamin", "Alamat", "Keluhan",
"Pembiayaan", "Dokter", "Jenis", "Jam Kerja Dokter", "Tanggal Pemeriksaan",
"Keterangan"]
        table.setHorizontalHeaderLabels(headers)

        # Populate the table with data
        table.setRowCount(len(result))
        for I, row in enumerate(result):
            for j in range(11):
                item = QtableWidgetItem(str(row[j]))
                table.setItem(I, j, item)

        layout.addWidget(table)

        search_result_dialog.exec_()
```

- **Membuat fungsi show untuk melihat kondisi**

```
    def lihat_kondisi(self):
        lihat_kondisi_dialog = None

        try:
            # Panggil load_data_kondisi untuk memastikan data kondisi terkini
            self.load_data_kondisi()

            # Create a new widget to display the conditions
            lihat_kondisi_dialog = QDialog()

            # Create layout for the widget
```

```
            layout = QVBoxLayout()
            layout.addWidget(self.pasien_list_periksa_kondisi)
            lihat_kondisi_dialog.setLayout(layout)

            # Show the widget as a modal dialog
            result = lihat_kondisi_dialog.exec_()

            # Handle the result to avoid issues when the dialog is closed
            if result == Qdialog.Accepted:
                # Any additional actions or cleanup
                pass
            elif result == Qdialog.Rejected:
                # Handle rejection or cleanup if needed
                pass

        except Exception as e:
            print(f”Error: {e}”)
            # Handle the error appropriately, such as showing an error message

        finally:
            if lihat_kondisi_dialog:
                lihat_kondisi_dialog.deleteLater()
```

- **Menyimpan hasil pemeriksaan yang telah dibuat kedalam Database**

```
    def simpan_hasil_pemeriksaan_to_db(self, nama_pasien, hasil_pemeriksaan, obat,
tanggal, keluhan, dokter):
        try:
            cursor = self.db_connection.cursor()
            sql = "INSERT INTO hasil_pemeriksaan (nama_pasien, hasil_pemeriksaan,
obat, tanggal_pemeriksaan, keluhan, dokter) VALUES (%s, %s, %s, %s, %s, %s)"
            values = (nama_pasien, hasil_pemeriksaan, obat, tanggal, keluhan,
dokter)
            cursor.execute(sql, values)
            self.db_connection.commit()
            cursor.close()
            self.show_dialog("Simpan Hasil Pemeriksaan", "Hasil pemeriksaan
berhasil disimpan.")
        except mysql.connector.Error as err:
            print(f"Error: {err}")
            self.show_dialog("Error", f"Error: {err}")


    def init_periksa_kondisi(self):
```

```
        self.periksa_kondisi_widget = PeriksaKondisi()
        self.periksa_kondisi_widget.kondisi_checked.connect(self.simpan_data_kond
isi)
        self.periksa_kondisi_widget.kondisi_checked.connect(self.handle_kondisi_c
hecked)
        self.periksa_kondisi_widget.show()
        self.pasien_list_periksa_kondisi = QListWidget()

    def handle_kondisi_checked(self, data):
        print("Data Kondisi Checked:", data)

    # def show_periksa_kondisi(self):
    #     # Show the PeriksaKondisi widget when the button is clicked


    def show_hasil_pemeriksaan(self):
    # Menentukan nama pasien dari entri pengguna atau data lainnya
        nama_pasien = self.input_nama.text()

    # Membuat instance HasilPemeriksaanWidget dengan kondisi_data yang sesuai
        hasil_pemeriksaan_widget = HasilPemeriksaanWidget({'nama': nama_pasien})
        hasil_pemeriksaan_widget.kartu_rumah_sakit = self  # Tetapkan objek
KartuRumahSakit sebagai orang tua

    # Connect the signal kondisi_changed to the slot in KartuRumahSakit
        hasil_pemeriksaan_widget.kondisi_changed.connect(self.load_data_kondisi)

    # Execute the dialog
        result = hasil_pemeriksaan_widget.exec_()

    # Handle the result if needed
        if result == QDialog.Accepted:
            print("Hasil Pemeriksaan dialog accepted.")
        # Perform actions upon accepting the dialog, if needed
            cetak_widget = CetakHasilPemeriksaanWidget({'nama': nama_pasien,
'hasil_pemeriksaan': 'contoh hasil', 'obat': 'contoh obat'})
            cetak_widget.exec_()
        else:
            print("Hasil Pemeriksaan dialog rejected or closed.")

    # Tidak perlu lagi menggunakan parent(), langsung akses atribut
self.kartu_rumah_sakit
        hasil_pemeriksaan_widget.kondisi_changed.connect(self.load_data_kondisi)

    def clear_input_fields(self):
```

```
        self.input_nama.clear()
        self.input_umur.clear()
        self.input_keluhan.clear()
        self.cmb_pembiayaan.setCurrentIndex(0)
        self.dokter.setCurrentIndex(0)
        self.jenis.setCurrentIndex(0)
        self.jam.setCurrentIndex(0)
        self.keterangan.setCurrentIndex(0)
```

- **Untuk menampilkan Dialog**

```
    def show_dialog(self, title, message):
        dialog = QMessageBox(self.centralwidget)  # Gunakan central widget
sebagai parent
        dialog.setIcon(QMessageBox.Information)
        dialog.setWindowTitle(title)
        dialog.setText(message)
        dialog.exec_()

    def closeEvent(self, event):
        self.db_connection.close()
        super().closeEvent(event)

class ParentWidget(QWidget):
    def __init__(self):
        super().__init__()

        self.kartu_rs = KartuRumahSakit()
        self.kartu_rs.setupUi()

        layout = QVBoxLayout(self)
        layout.addWidget(self.kartu_rs)
```

- **Run Progam**

```
def main():
    app = QApplication(sys.argv)
    parent_widget = ParentWidget()
    parent_widget.show()
    sys.exit(app.exec_())


if __name__ == '__main__':
    app = QtWidgets.QApplication(sys.argv)
    MainWindow = QtWidgets.QMainWindow()
    ui = KartuRumahSakit()
```

```
    ui.setupUi(MainWindow)
    MainWindow.show()
    sys.exit(app.exec_())
```

DAFTAR PUSTAKA

Ambler, Scott W. (2004). *"Introduction to UML 2 Class Diagrams."* Agile Modeling.

Fowler, Martin. (2004). *"UML Distilled: A Brief Guide to the Standard Object Modeling Language*." Addison-Wesley.

Summerfield, M. (2013). *"Rapid GUI Programming with Python and Qt: The Definitive Guide to PyQt Programming."* Prentice Hall.