**LAPORAN**

**TUGAS AKHIR**

Mata Kuliah Pemrograman Berorientasi Objek



Dosen Pengampu:

M. Bahrul Subkhi, M.Kom

Disusun oleh:

|  |  |
| --- | --- |
| 1. Cahyo Septian Nugroho | 2213020153 |
| 1. Rafael Yonathan Timotius | 2213020155 |
| 1. Yustitio Caesar | 2213020143 |

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK DAN ILMU KOMPUTER**

**UNIVERSITAS NUSANTARA PGRI KEDIRI**

**TAHUN 2023**

**DAFTAR ISI**

Daftar Isi**2**

Flowchart Sistem3

Class Diagram4

Hasil Program dan Penjelasan5

Daftar Pustaka6

**FLOWCHART SISTEM**

**A diagram of a flowchart

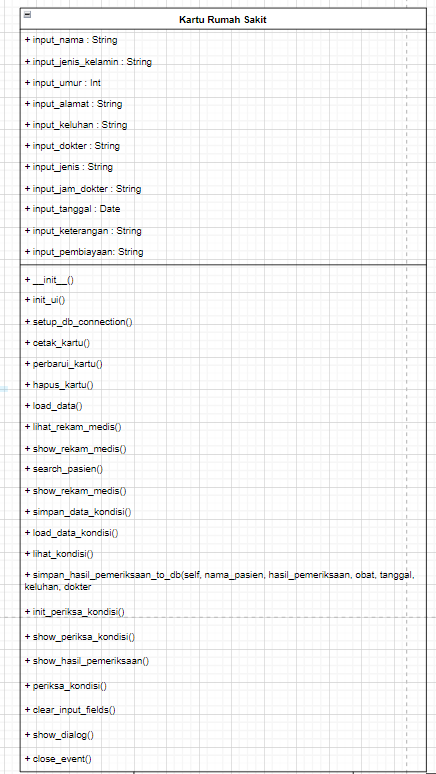
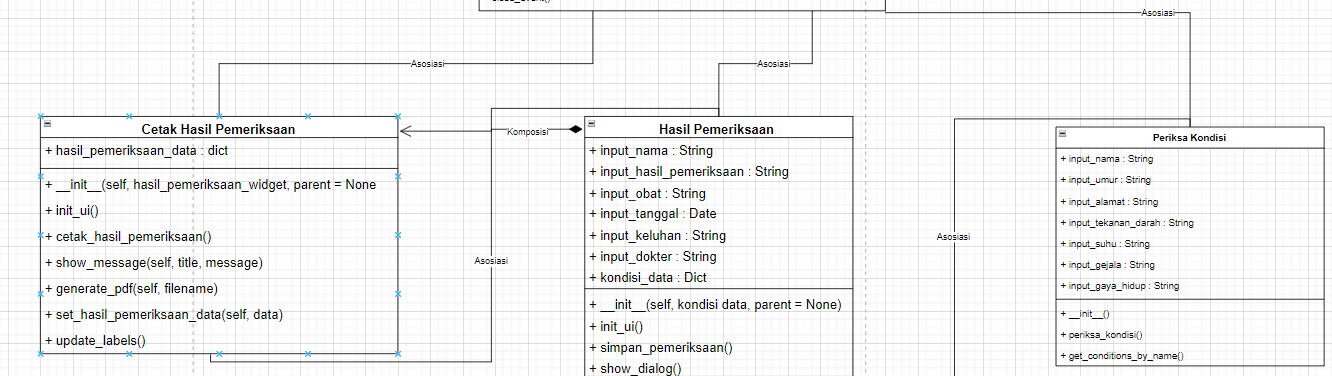
Description automatically generated**

**A diagram of a data flow

Description automatically generated**

**A diagram of a workflow

Description automatically generated**

**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\_kondisi)

        self.periksa\_kondisi\_widget.kondisi\_checked.connect(self.handle\_kondisi\_checked)

        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.