

APLIKASI CEK KESEHATAN SEDERHANA

Dosen pengampu :

Pahrul Irfan, S.Kom., M.Kom.



Disusun oleh :

QHAULAN SYAQHILA (F1D022152)

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS MATARAM

2025

A. DESKRIPSI SINGKAT DARI APLIKASI CEK KESEHATAN SEDDERHANA

Program yang ditampilkan merupakan aplikasi berbasis PyQt5 bernama "Aplikasi Cek Kesehatan Sederhana" yang bertujuan untuk memberikan evaluasi awal terhadap status kesehatan pengguna berdasarkan beberapa input data pribadi, seperti nama, berat badan, umur, tinggi badan, kebiasaan merokok, dan tingkat aktivitas fisik. Aplikasi ini menggunakan antarmuka grafis (GUI) yang dirancang di Qt Designer dan dimuat melalui file "Project Mini.ui".

Ketika pengguna mengisi data dan menekan tombol "Cek", program akan memvalidasi input yang diberikan. Jika input valid, aplikasi akan menghitung Body Mass Index (BMI) pengguna dan menampilkan status kesehatan yang dikategorikan sebagai "SEHAT" atau "PERLU PERHATIAN". Hasil tersebut ditampilkan bersama dengan kategori BMI, seperti "Kurang Berat Badan", "Normal", "Kelebihan Berat Badan", atau "Obesitas", beserta saran terkait dari BMI.

Selain itu, aplikasi memberikan saran yang disesuaikan dengan kondisi pengguna berdasarkan faktor-faktor seperti umur, kebiasaan merokok, dan aktivitas fisik. Sebagai contoh, jika pengguna jarang berolahraga atau merokok, aplikasi akan menyarankan untuk meningkatkan aktivitas fisik atau berhenti merokok. Teks hasil juga diberi warna, seperti hijau untuk status "SEHAT" dan merah untuk "PERLU PERHATIAN". Aplikasi ini juga dilengkapi dengan menu bantuan yang menampilkan informasi mengenai aplikasi.

B. LANGKAH – LANGKAH PEMBUATAN PROYEK

1. Membuat repository.
2. Menentukan apa yang ingin dibuat.
3. Mendesain UI menggunakan QtDesigner.
4. Mulai melakukan codingan.
5. Menyabungkan keduanya (file .ui dan python).
6. Melakukan testing.
7. Membuat laporan.
8. Push pada github.
9. Kumpulin tugas di berajah.

C. PENJELASAN FUNGSI UTAMA YANG DIGUNAKAN DALAM APLIKASI

```
def __init__(self):
```

```
.....
```

Script di atas merupakan konstrktor dari kelas "HealthChecker" yang digunakan untuk menginisialisasi aplikasi. *Script* ini juga memuat UI dari file "Project Mini.ui" menggunakan "loadUi", menyimpan judul aplikasi ("setWindowTitle"), dan mengecek serta memvalidasi apakah semua widget yang diperlukan tersedia di file ".ui". *Script* ini menghubungkan komponen ke fungsinya masing – masing menggunakan signals dan slots seperti "valueChanged" pada slider di fungsi "slider_changed", "clicked"

pada tombol di fungsi “check_health”, dan aksi menu di fungsi “show_about” serta menyembunyikan “output_result” di awal program.

```
def slider_changed(self):
    self.slider digeser = True
    self.label_tinggi.setText(f"{self.slider_tinggi.value()} cm")
```

Script di atas dijalankan saat slinder tinggi badan di geser dan dapat menampilkan nilai tinggi badan terkini di label (dalam satuan cm).

```
def check_health(self):
    nama = self.inputNama.text().strip()
    berat = self.inputBerat.text().strip()
    umur = self.spinUmur.value()
    tinggi = self.slider_tinggi.value()
    merokok_ya = self.radio_ya.isChecked()
    merokok_tidak = self.radio_tidak.isChecked()
    aktivitas = self.combo_aktivitas.currentText()

    if not nama:
        QMessageBox.warning(self, "Peringatan", "Nama tidak boleh kosong!")
        return
    if not berat:
        QMessageBox.warning(self, "Peringatan", "Berat tidak boleh kosong!")
        return
    try:
        berat = float(berat)
    except ValueError:
        QMessageBox.warning(self, "Peringatan", "Berat harus berupa angka!")
        return
    if umur == 0:
        QMessageBox.warning(self, "Peringatan", "Umur belum diisi!")
        return
    if not self.slider digeser:
        QMessageBox.warning(self, "Peringatan", "Tinggi badan belum diatur!")
        return
    if not (merokok_ya or merokok_tidak):
        QMessageBox.warning(self, "Peringatan", "Silakan pilih status merokok!")
        return
    if aktivitas == "-- Pilih --":
        QMessageBox.warning(self, "Peringatan", "Silakan pilih aktivitas fisik!")
        return

    tinggi_meter = tinggi / 100
    bmi = berat / (tinggi_meter ** 2)

    if bmi < 18.5:
        bmi_category = "Kurang Berat Badan"
        bmi_saran = "Perhatikan asupan makanan yang bergizi."
    elif bmi < 24.9:
        bmi_category = "Normal"
        bmi_saran = "Pertahankan gaya hidup sehat Anda!"
    elif bmi < 29.9:
```

```

        bmi_category = "Kelebihan Berat Badan"
        bmi_saran = "Cobalah untuk menurunkan berat badan dengan
diet dan olahraga."
    else:
        bmi_category = "Obesitas"
        bmi_saran = "Konsultasikan dengan dokter untuk program
penurunan berat badan."

    merokok = merokok_ya
    saran = []

```

Script di atas adalah fungsi inti logika aplikasi yang mengambil dan memvalidasi input seperti nama, berat badan, umur, tinggi badan, status merokok, dan Tingkat aktivitas fisik. Kemudian semua input divalidasi agar tidak ada yang dikosongkan atau terlewatkan. Kemudian menentukan kategori BMI (*Body Mass Index*) dan saran berdasarkan nilai BMI.

```

if umur > 40:
    saran.append("Rutin periksa kesehatan.")
    if merokok:
        saran.append("Disarankan untuk mengurangi atau
menghentikan kebiasaan merokok untuk menjaga kesehatan paru-paru dan
mencegah risiko penyakit kronis seperti kanker dan penyakit
jantung.")
    else:
        saran.append("Pertahankan rutinitas aktivitas fisik Anda
agar kesehatan tubuh tetap terjaga dan risiko penyakit dapat
diminimalkan.")

```

Script di atas ini digunakan untuk menganalisis kesehatan berdasarkan factor lain seperti, jika usia > 40 maka disarankan periksa kesehatan rutin; jika merokok maka disarankan untuk berhenti; dan jika tidak merokok disarankan untuk mempertahankan kebiasaan baik tersebut.

```

status = "SEHAT" if not merokok and aktivitas == "Rutin" and umur <
40 else "PERLU PERHATIAN"
color = "#2ecc71" if status == "SEHAT" else "#e74c3c"

```

Script di atas berguna untuk mempengaruhi warna teks hasil evaluasi yang dimana jika warnanya hijau (#2ecc71) akan dinyatakan sehat, dan jika berwarna merah (#e74c3c) maka diminta untuk memperhatikan kesehatannya.

Dan tampilan akhirnya akan menampilkan semua data dan saran – saran pada output (`output_result`), termasuk nama pengguna, status kesehatan, nilai dan katagori BMI, dan saran kesehatan tambahan.

D. HASIL

