**LAPORAN**

**TUGAS AHIR**

**“ Program Bereorientasi Obyek ”**

Dosen Pengampu : M Bahrul Subkhi, M.Kom



**Disusun Oleh :**

**Reihan Rafi Rahmatuloh (2213020244)**

**Nugroho Bimantoro (2213020224)**

**Moch. Saefudin Yuhri (2213020205)**

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS NUSANTARA PGRI KEDIRI 2023

# LATAR BELAKANG

* Latar belakang pembuatan aplikasi parkir dapat bermacam-macam, tetapi beberapa alasan umum termasuk meningkatkan efisiensi pengelolaan parkir, memberikan kemudahan bagi pengguna untuk menemukan dan membayar tempat parkir, serta mengurangi kemacetan dan polusi udara akibat pencarian tempat parkir yang tidak efisien. Selain itu, aplikasi parkir juga dapat membantu mengoptimalkan pemanfaatan ruang parkir dan meningkatkan pengalaman pengguna dalam mengakses fasilitas parkir.
* Selain itu, latar belakang pembuatan aplikasi parkir juga melibatkan adaptasi terhadap perkembangan teknologi untuk memenuhi tuntutan zaman. Aplikasi parkir modern tidak hanya menyediakan informasi mengenai ketersediaan tempat parkir, tetapi juga dapat memanfaatkan sensor dan teknologi cerdas untuk mengelola parkir secara lebih efisien, mempercepat proses pembayaran, dan meningkatkan keamanan area parkir. Dengan demikian, aplikasi parkir berperan dalam menciptakan solusi terintegrasi untuk memecahkan permasalahan parkir perkotaan yang semakin kompleks.

DAFTAR ISI

[DAFTAR ISI 1](#_Toc154397165)

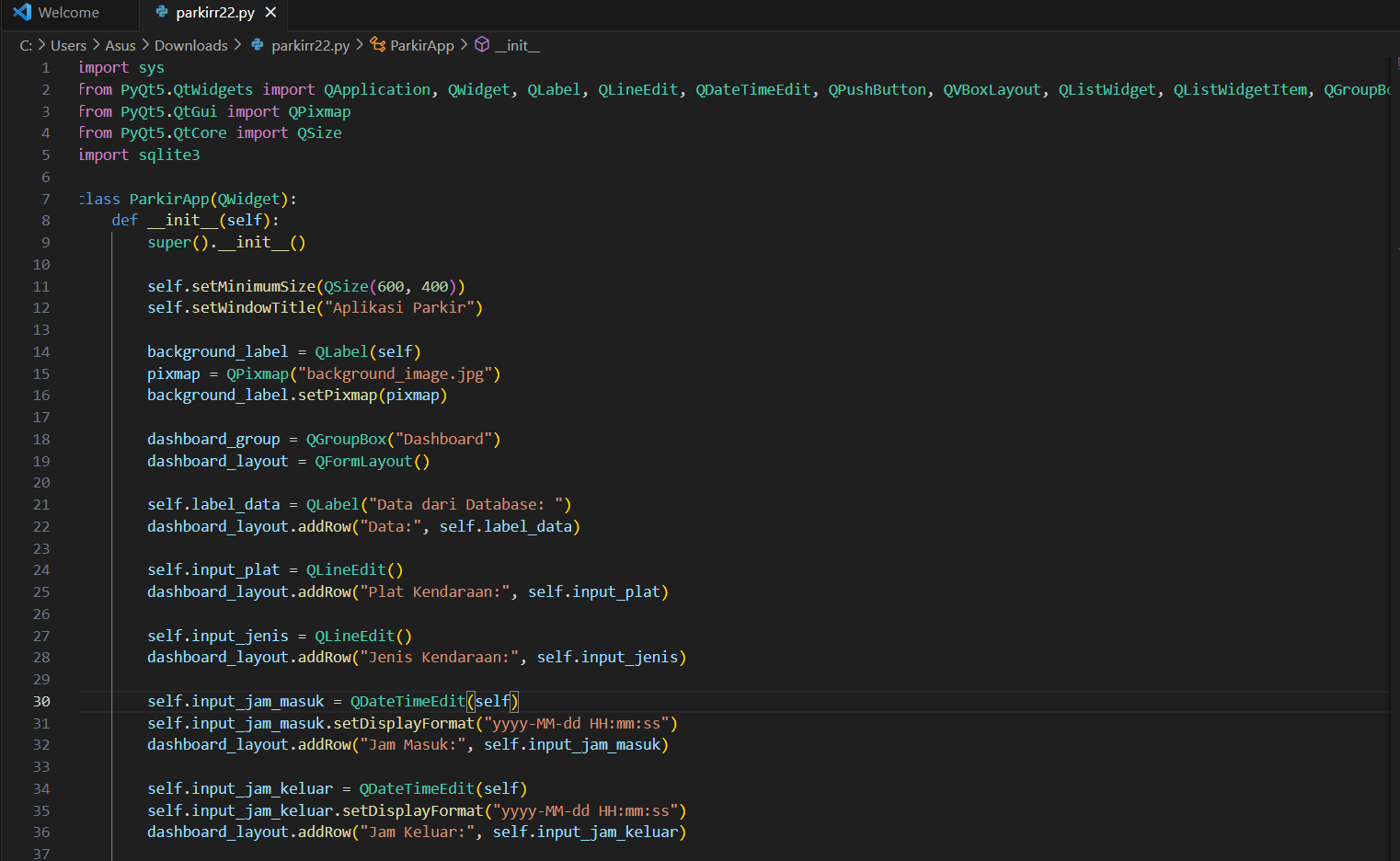
[BAB I 2](#_Toc154397166)

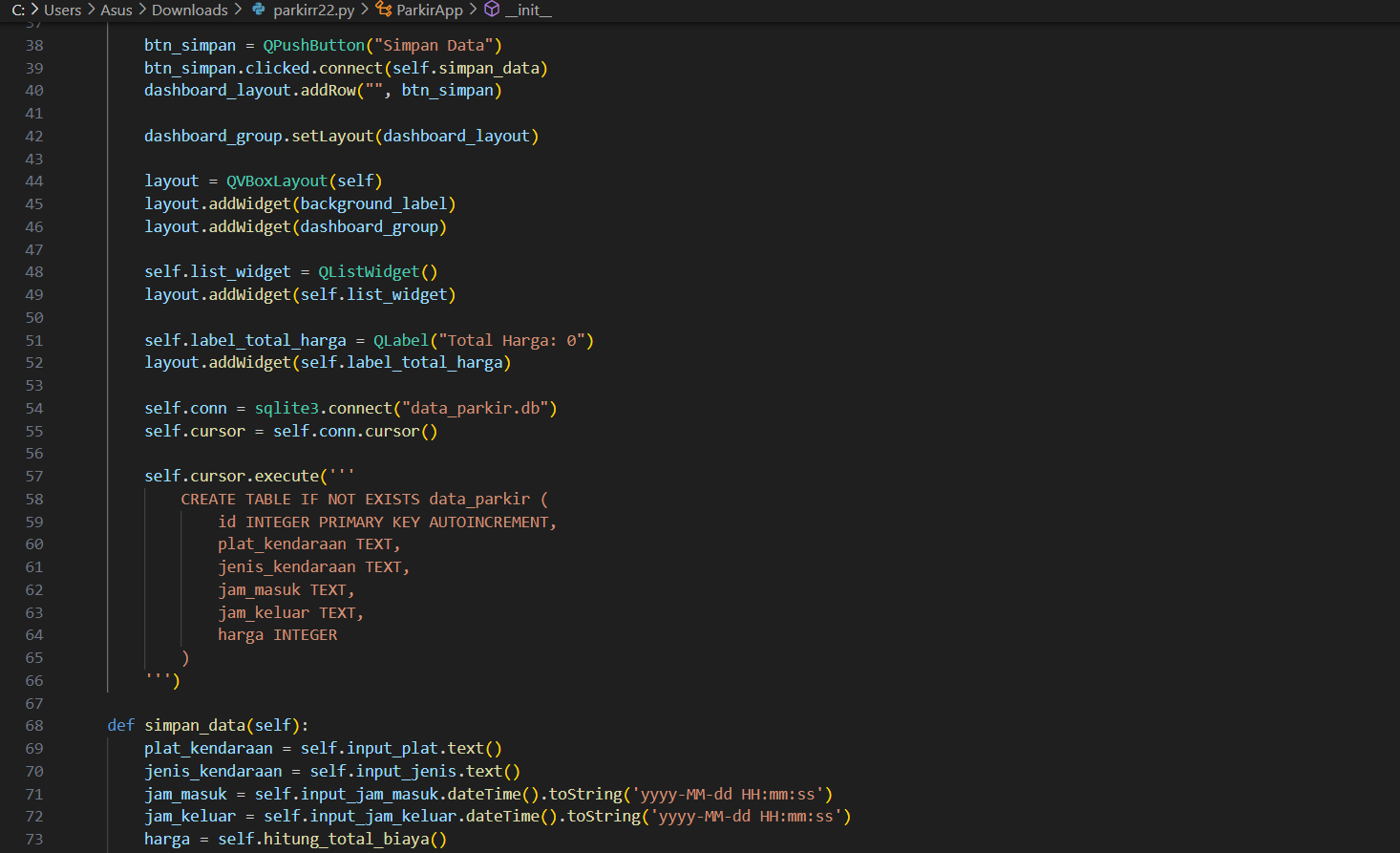
[1.1 Hasil Program dan Penjelasan 2](#_Toc154397167)

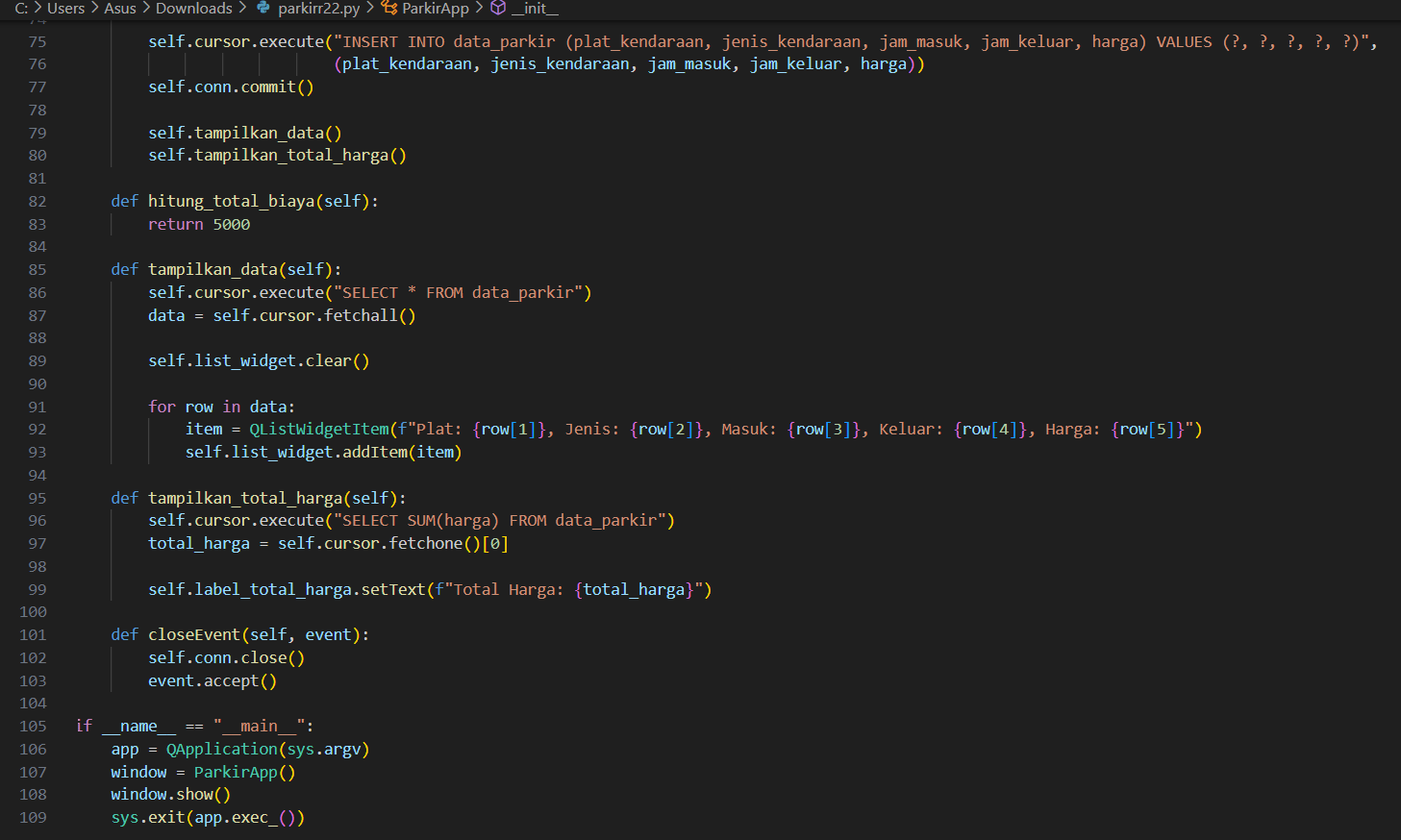
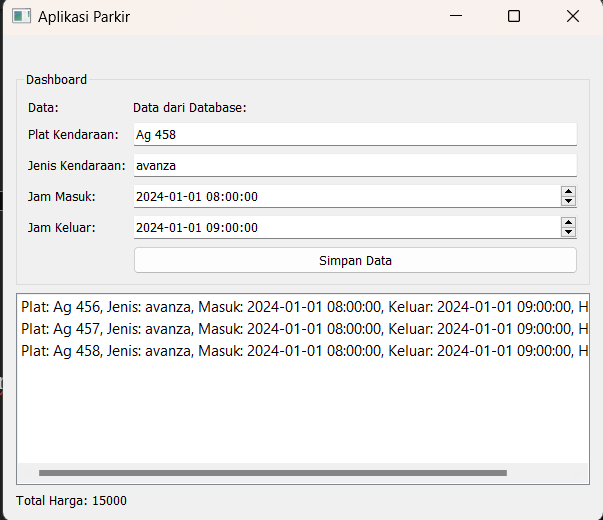
[1.2 Daftar Pustaka 3](#_Toc154397168)

# BAB I

# Hasil Program Dan Penjelasan







Penjelasan

1. \*Impor Modul:\*

- Mengimpor modul sys untuk akses variabel yang digunakan oleh interpreter.

- Mengimpor beberapa kelas dan modul dari PyQt5 untuk membangun antarmuka pengguna (UI).

- Mengimpor sqlite3 untuk mengakses dan mengelola database SQLite.

2. \*\*Definisi Kelas (AplikasiParkir):\*\*

- Mewarisi dari kelas QWidget untuk membuat jendela aplikasi utama.

- Mendefinisikan konstruktor \_\_init\_\_ di mana elemen-elemen UI dan koneksi database diatur.

3. \*Pengaturan UI:\*

- Menetapkan ukuran minimum dan judul jendela.

- Membuat label gambar latar belakang menggunakan QLabel dan mengatur pixmap-nya.

- Membuat QGroupBox untuk dashboard dengan berbagai kolom input, label, dan tombol "Simpan Data".

- Mengatur manajer tata letak (QFormLayout dan QVBoxLayout) untuk mengorganisir widget.

4. \*Koneksi Database:\*

- Menghubungkan ke database SQLite (data\_parkir.db) dan membuat objek kursor.

- Mendefinisikan struktur tabel (data\_parkir) jika belum ada, termasuk kolom untuk detail kendaraan, waktu masuk/keluar, dan harga.

5. \*\*Entri Data (simpan\_data):\*\*

- Mendapatkan nilai dari inputan (plat kendaraan, jenis kendaraan, jam masuk, jam keluar).

- Menghitung harga total dengan memanggil fungsi hitung\_total\_biaya.

- Menyimpan data ke dalam database menggunakan pernyataan SQL INSERT.

- Memanggil fungsi untuk menampilkan data terkini dan total harga.

6. \*\*Perhitungan Harga (hitung\_total\_biaya):\*\*

- Mengembalikan nilai tetap (5000) sebagai contoh perhitungan harga.

7. \*\*Menampilkan Data (tampilkan\_data):\*\*

- Mengeksekusi pernyataan SQL SELECT untuk mengambil semua data dari tabel.

- Membersihkan widget daftar (QListWidget).

- Membuat item untuk setiap baris data dan menambahkannya ke daftar.

8. \*\*Menampilkan Total Harga (tampilkan\_total\_harga):\*\*

- Mengeksekusi pernyataan SQL SELECT untuk menghitung total harga dari tabel.

- Mengupdate label total harga dengan nilai yang diambil dari database.

9. \*\*Penanganan Penutupan Aplikasi (closeEvent):\*\*

- Menutup koneksi database saat aplikasi ditutup.

10. \*\*Eksekusi Aplikasi (\_\_main\_\_):\*\*

- Membuat objek QApplication untuk menginisialisasi aplikasi.

- Membuat objek AplikasiParkir sebagai jendela utama.

- Menampilkan jendela dan menjalankan aplikasi.