

IMPLEMENTASI LIBRARY PYTHON DEQUEUE PADA ANTRIAN BANK MENGUNAKAN LOGIKA FIRST IN FIRST OUT

Atta Zulfahrizan, Muhammad Alby Savana HSB, Fhadillah br.Hutagalung, Fanny Ramadhani

Ilmu Komputer, Universitas Negeri Medan

Jalan Willem Iskandar, Pasar V Medan Estate, Percut Sei Tuan, Deli Serdang, Indonesia

latifhamzah106@gmail.com

ABSTRAK

Sistem antrian bank merupakan bagian penting dalam pelayanan kepada nasabah, namun seringkali menghadapi permasalahan seperti pembagian pelanggan yang tidak optimal dan tidak adanya estimasi waktu tunggu yang akurat. Kondisi ini menyebabkan waktu tunggu yang lama dan ketidakpuasan nasabah. Penelitian ini bertujuan untuk mengembangkan sistem antrian berbasis algoritma First In First Out (FIFO) menggunakan library Python deque untuk meningkatkan efisiensi layanan nasabah. Metode yang digunakan meliputi perancangan dan implementasi algoritma FIFO dengan struktur data deque, pengujian sistem, serta pengembangan fitur pembagian antrian berdasarkan estimasi waktu layanan. Hasil pengujian menunjukkan bahwa sistem yang dikembangkan mampu meningkatkan efisiensi administrasi, mengurangi kesalahan input data, dan memprioritaskan pelanggan berdasarkan waktu tunggu yang lebih singkat. Dengan demikian, sistem ini memberikan solusi yang efektif untuk pengelolaan antrian perbankan secara adil dan terstruktur.

Kata kunci : Struktur data, Sistem antrian, Dequeue, First in first out, First come first served, GUI.

1. PENDAHULUAN

Sistem antrian adalah bagian penting dari pelayanan di sektor perbankan. Sebagai institusi yang melayani berbagai kebutuhan nasabah, seperti penarikan, penyetoran, transfer, atau pembukaan rekening baru, bank menghadapi tantangan dalam mengelola antrian yang sering kali panjang. Pengelolaan antrian yang buruk dapat menyebabkan waktu tunggu yang lama, ketidakpuasan nasabah, dan penurunan kualitas layanan. Oleh karena itu, diperlukan sistem antrian yang efisien, terstruktur, dan adil untuk meningkatkan pengalaman nasabah.

Logika First In First Out (FIFO) adalah salah satu pendekatan yang paling umum digunakan dalam pengelolaan antrian. Dalam logika ini, pelanggan yang datang lebih dulu akan dilayani terlebih dahulu. Pendekatan FIFO memastikan keadilan dalam urutan layanan dan membantu bank memberikan pengalaman yang lebih baik bagi pelanggan. Untuk mengimplementasikan logika FIFO secara efisien, digunakan library Python deque.

Library deque (double-ended queue) menawarkan performa tinggi dalam operasi manipulasi elemen di kedua ujung struktur data, seperti penambahan dan penghapusan elemen. Dibandingkan dengan struktur data lainnya seperti list, deque lebih optimal dalam pengelolaan antrian karena kecepatan operasi yang konstan pada ujung-ujungnya. Proyek ini bertujuan untuk mengembangkan sistem antrian bank berbasis deque dengan fitur tambahan, seperti pembagian antrian berdasarkan estimasi waktu pelayanan dan antarmuka pengguna grafis (GUI) sederhana untuk mempermudah interaksi pengguna.

Penelitian ini menjelaskan implementasi sistem antrian bank menggunakan library Python deque, termasuk perancangan sistem, pengembangan fitur,

dan pengujian kinerja untuk memastikan efisiensi dan kemudahan penggunaan.

Latar belakang pada penelitian ini berfokus pada masalah penting dalam pengelolaan antrian di sektor perbankan. Sistem konvensional seringkali tidak dapat menangani masalah seperti pembagian pelanggan yang tidak optimal, estimasi waktu tunggu yang tidak akurat, dan proses pelayanan yang lambat. Ketidakpuasan pelanggan, waktu tunggu yang lama, dan kualitas layanan yang buruk secara keseluruhan dapat dihasilkan dari situasi seperti ini. Seiring dengan peningkatan permintaan pelanggan untuk layanan yang cepat dan efisien, solusi kreatif diperlukan untuk mengoptimalkan pengelolaan antrian. Algoritma First In First Out (FIFO), yang memastikan bahwa pelanggan dilayani sesuai dengan urutan kedatangan mereka, memberikan keadilan dalam pelayanan. Dalam penelitian ini, algoritma FIFO digunakan dengan library Python deque, yang memiliki keunggulan dalam efisiensi waktu operasi dengan kompleksitas $O(1)$ untuk penambahan.

2. TINJAUAN PUSTAKA

2.1. Konsep Dasar Struktur Data

Salah satu hal penting yang tidak dapat ditinggalkan dalam pemakaian komputer adalah data. Data dapat diperoleh dari berbagai sumber, misalkan hasil pengukuran laboratorium, hasil survei, hasil angket dan lain sebagainya. Komposisi data dan logika dari algoritma yang memanfaatkan data tersebut berhubungan sangat erat. Data sederhana dapat kita himpun ke dalam suatu struktur data yang memuat informasi tentang hubungan antara item yang terdapat di dalamnya. Dalam istilah ilmu komputer, struktur data adalah cara penyimpanan, pengorganisasian dan pengaturan data di dalam media penyimpanan

komputer sehingga data tersebut dapat digunakan secara efisien[1].

2.2. First in first out (FIFO)

First In First Out (FIFO) adalah metode penilaian persediaan yang menganggap barang yang pertama kali masuk diasumsikan keluar pertama kali pula. Pada umumnya perusahaan menggunakan metode ini, sebab metode ini perhitungannya sangat sederhana baik sistem fisik maupun sistem perpetual akan menghasilkan penilaian persediaan yang sama[2].

FIFO merupakan algoritma penerapan sistem antrian nonprioritas yang menggunakan struktur data tertentu. Algoritma FIFO banyak digunakan dalam berbagai solusi permasalahan kehidupan serta aplikasi dan teknologi. FIFO bekerja secara berurutan dan bergantian, memproses item sesuai urutan pemasukannya, yaitu yang pertama masuk akan diproses terlebih dahulu[3].

2.3. VS Code

Visual Studio Code adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi multiplatform, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemrograman Javascript, Typescript, dan Node. Js, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via marketplace Visual Studio Code seperti : C++, C#, Python, Go, Java, PHP, dst[4].

2.4. Graphical User Interface (GUI)

Graphical User Interface adalah jenis antarmuka pengguna yang memungkinkan pengguna untuk berinteraksi dengan komputer atau perangkat melalui elemen visual seperti icons, buttons, dan windows. Berbagai permasalahan seperti daya tarik GUI, seberapa intuitif penggunaannya dan 'keramahan penggunaannya' akan sangat penting untuk keberhasilan produk atau layanan berbasis perangkat lunak tertentu. Penggunaan Graphical User Interface berdampak pada peningkatan produktivitas dan efisiensi di berbagai industri. Secara keseluruhan, potensi ekonomi GUI dalam produk digital sangat signifikan, fenomena ini berkorelasi dengan pilihan 2,56 juta aplikasi yang dapat diunduh oleh pengguna smartphone dari Google Play Store dan sebanyak 1,85 juta aplikasi dapat di unduh dari App Store[5].

2.4. Queue

Queue atau antrian adalah suatu jenis struktur data yang dapat diproses dengan sifat FIFO (First In First Out), dimana elemen yang pertama kali masuk ke antrian akan keluar pertama kali. Ada dua jenis operasi yang bisa dilakukan di antrian: enqueue (memasukkan elemen baru ke dalam elemen) dan dequeue (adalah mengeluarkan satu elemen dari suatu antrian). Antrian dapat dibuat dengan menggunakan: Linear Array dan Circular Array[6].

2.5. Pemrograman Python

Python merupakan salah satu bahasa pemrograman yang banyak digunakan oleh perusahaan besar maupun para developer untuk mengembangkan berbagai macam aplikasi berbasis desktop, web dan mobile. Python diciptakan oleh Guido van Rossum di Belanda pada tahun 1990 dan namanya diambil dari acara televisi kesukaan Guido Monty Python's Flying Circus. Van Rossum mengembangkan Python sebagai hobi, kemudian Python menjadi bahasa pemrograman yang dipakai secara luas dalam industri dan pendidikan karena sederhana, ringkas, sintak intuitif dan memiliki pustaka yang luas[7].

2.5. Penelitian Terdahulu

- a. Judul: Implementasi Metode Queue Pada Sistem Antrian Online Berbasis Web Studi Kasus UPTD Puskesmas Sananwetan.
- b. Penulis: Rizaldy Gunawan, H. Yuana, S. Kirom (2023).
- c. Ringkasan: Penelitian ini membangun sistem antrian online berbasis web dengan logika FIFO untuk mengelola pasien di puskesmas. Sistem menggunakan struktur data Queue untuk memasukkan pasien secara urut berdasarkan waktu kedatangan dan mendistribusikan mereka ke layanan yang tersedia.
- d. Kesamaan: Penelitian ini berfokus pada pengelolaan antrian menggunakan algoritma FIFO, dengan tujuan untuk mengurangi dampak negatif waktu tunggu yang lama melalui pembagian antrian yang lebih efisien. Pendekatan ini serupa dengan penelitian yang dilakukan, di mana antrian dibagi berdasarkan estimasi waktu layanan guna meningkatkan efisiensi dan keadilan dalam pelayanan.
- e. Perbedaan: Penelitian ini diimplementasikan pada sistem berbasis web, sedangkan penelitian yang dilakukan menggunakan sistem berbasis GUI desktop untuk mendukung interaksi pengguna secara langsung.

3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan kualitatif untuk memahami proses, dan implementasi sistem antrian bank berbasis Python yang mengadopsi algoritma First In First Out (FIFO). Pendekatan ini dilakukan untuk memahami secara mendalam bagaimana sistem tersebut dirancang, diimplementasikan, dan berkontribusi terhadap efisiensi layanan di bank. Program ini dirancang untuk sistem antrian bank yang adil dan terstruktur imana yang pertama datang lah yang seharusnya dilayani pertama.

3.1. Identifikasi Masalah

Penelitian ini berfokus pada kebutuhan sistem antrian yang mampu mengelola pelanggan secara efisien dalam antrian perbankan. Penelitian ini

bertujuan untuk mengatasi inefisiensi dalam penanganan antrian bank yang masih menggunakan cara lama yang diidentifikasi terdapat beberapa masalah:

- Tidak optimalnya pembagian customer ke tiap resepsionis.
- Tidak adanya estimasi waktu tunggu pada setiap customer.

Maka dari itu, kehadiran sistem antrian yang di automasi serta menyediakan estimasi waktu tunggu tiap customer, dan pembagian pelanggan ke resepsionis secara otomatis akan sangat mengoptimalkan efisiensi dari antrian bank.

3.2. Perancangan Algoritma First In First Out

Berdasarkan identifikasi masalah, system antrian dengan algoritma FIFO (First In First Out) dirancang untuk melayani customer secara berurut sesuai dengan waktu kedatangannya. Dalam penelitian ini, library Python deque dari modul collection digunakan sebagai struktur data utama. Deque dipilih karena mendukung operasi penambahan (append) dan penghapusan (popleft) elemen dengan efisiensi waktu $O(1)$.

3.3. Keunggulan penggunaan deque

Keunggulan dari penggunaan deque ini yaitu sebagai berikut:

- Efisiensi**
Dibandingkan dengan list, deque memiliki performa lebih baik untuk operasi yang melibatkan elemen diujung awal atau akhir, sehingga cocok digunakan untuk algoritma FIFO
- Sederhana dan Praktis**
Fungsi bawaan seperti `append()` dan `popleft()` mempermudah implementasi algoritma FIFO tanpa perlu pengelolaan indeks secara manual
- Kompleksitas yang Stabil**
Semua operasi utama pada deque (penambahan dan penghapusan elemen) memiliki kompleksitas $O(1)$, sehingga cocok untuk menangani antrian dengan skala besar.

3.4. Tahapan Implementasi Algoritma FIFO

Tahapan dalam mengimplementasikan algoritma FIFO dalam penelitian ini yaitu:

- Inisialisasi Sistem Antrian**
Antrian menggunakan 2 resepsionis yang diinisialisasi menggunakan deque sebagai struktur data utama. Setiap elemen antrian menyimpan nomor antrian, jenis layanan yang dipilih, juga estimasi waktu tunggu customer.
- Menambahkan pelanggan ke Dalam Antrian**
Saat pelanggan baru saja selesai memilih layanan dan mengisi form yang diberikan, data mereka ditambahkan ke ujung belakang deque menggunakan sintaks `append()`. Sistem juga memberikan kode antrian dan menghitung estimasi waktu tunggu berdasarkan pelanggan yang ada di depannya.
- Menghapus Pelanggan dari Antrian**

Setelah pelanggan selesai dilayani, data nya akan dihapus dengan menggunakan sintaks `popleft()`. Waktu tunggu customer dibelakangnya juga diperbarui secara otomatis.

- Pembagian Pelanggan ke Dua Antrian**
Untuk meningkatkan efisiensi dan meminimalisir efek dari convoy, yaitu customer yang hanya akan memakan waktu sebentar terjebak menunggu lama dibelakang customer yang memakan waktu sangat lama, sistem membagi pelanggan ke dua resepsionis berdasarkan waktu tunggu yang lebih cepat.
- Pengecekan Kondisi Antrian**
Sebelum pelanggan diproses, sistem memeriksa apakah deque tidak kosong, guna menghindari kesalahan saat memproses elemen.

3.5. Pengelolaan Estimasi Waktu Pelayanan

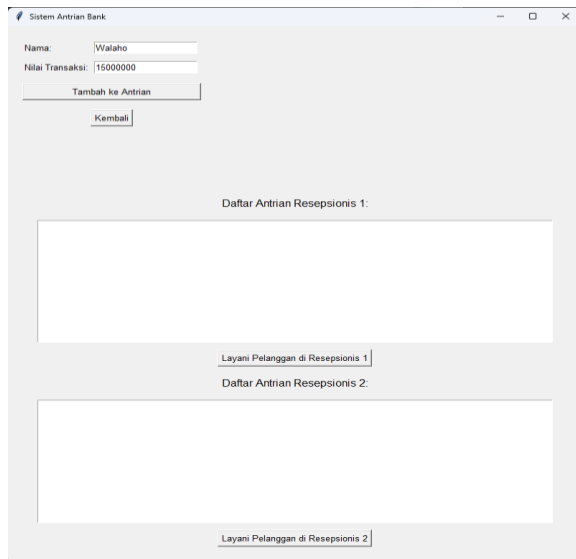
Sistem antrian bank yang dirancang menggunakan algoritma FIFO pada penelitian ini juga mencakup pengaturan waktu layanan berdasarkan jenis transaksi. Setiap jenis layanan memiliki waktu standar yang telah ditentukan, dengan penyesuaian tambahan untuk transaksi dalam jumlah besar. Berikut adalah rincian estimasi waktu yang diterapkan dalam sistem:

- Penyetoran dan Penarikan**
Waktu standar untuk layanan penyetoran dan penarikan adalah 5 menit. Jika jumlah transaksi mencapai Rp10 juta, maka waktu pelayanan akan bertambah 5 menit untuk setiap kelipatan Rp10 juta.
- Konsultasi**
Layanan konsultasi membutuhkan waktu standar 10 menit.
- Pembukaan Rekening Baru**
Proses pembukaan rekening baru memakan waktu lebih lama, yaitu 20 menit, mengingat adanya proses administrasi dan verifikasi data.
- Transfer Antar Bank atau Pembayaran Tagihan**
Layanan transfer antar bank dan pembayaran tagihan diberikan waktu standar 5 menit.
- Kehilangan Kartu**
Penanganan kehilangan kartu memerlukan waktu hingga 20 menit karena melibatkan verifikasi identitas dan proses pelaporan. Jadi pada saat penempatan customer ke dalam antrian, akan dipilih antrian yang estimasi waktu tunggu nya lebih singkat daripada antrian yang satunya lagi, juga misalkan kedua antriannya memiliki waktu tunggu yang sama, maka akan diprioritaskan ke antrian 1.

4. HASIL DAN PEMBAHASAN

Hasil penelitian menunjukkan bahwa sistem pengisian formulir dengan fitur pengisian data otomatis berhasil meningkatkan efisiensi dalam proses pengisian data pada layanan administrasi. Sistem ini juga mampu mengurangi kesalahan penginputan,

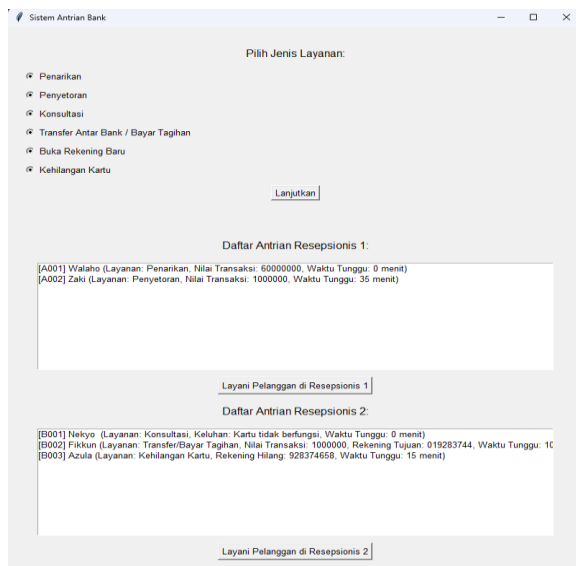
karena data yang dimasukkan dapat dihapus atau diperbaiki kembali sebelum dikirimkan ke sistem.



Gambar 1. Tampilan GUI Penginputan data.

Secara keseluruhan, sistem pengisian data yang telah dimodernisasi ini mampu mengurangi waktu yang dibutuhkan pengguna untuk mengisi formulir dibandingkan metode manual. Selain itu, sistem ini efektif dalam meminimalkan kesalahan input, seperti salah ketik atau kelalaian dalam pengisian data.

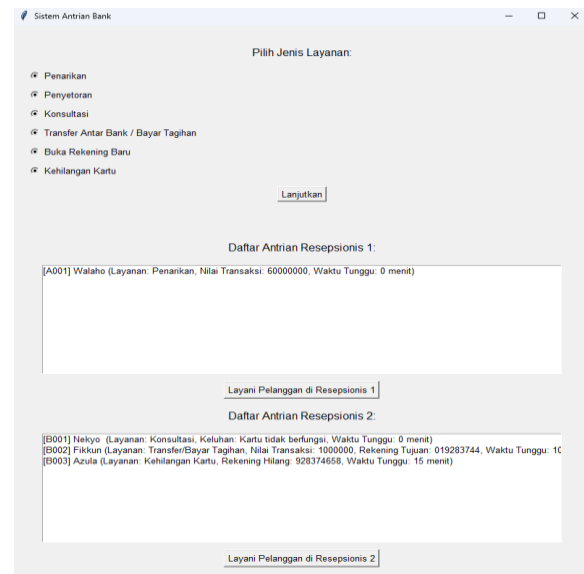
Sebagai bagian dari penelitian, dilakukan pengujian terhadap algoritma First In First Out (FIFO) pada sistem antrian yang dirancang. Pengujian ini bertujuan untuk mengevaluasi apakah sistem dapat membagi antrian berdasarkan waktu tunggu yang paling cepat.



Gambar 2. Percobaan Algoritma.

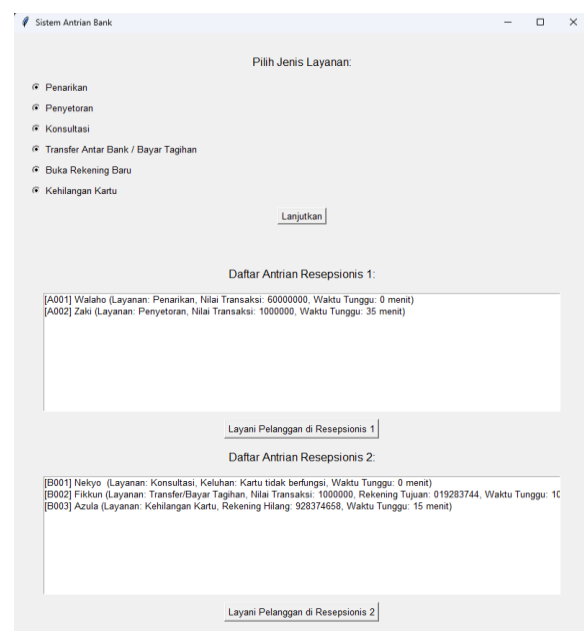
Pada gambar di atas, terlihat simulasi di mana dua pelanggan telah dimasukkan ke dalam dua antrian. Pada antrian 1, terdapat pelanggan Walaho yang memilih layanan penarikan dengan nilai transaksi

sebesar 60 juta rupiah. Layanan ini memiliki estimasi waktu awal 5 menit, tetapi karena nilai transaksi melebihi 10 juta rupiah, waktu estimasi tambahan sebesar 25 menit ditambahkan. Total waktu tunggu untuk antrian 1 menjadi 30 menit. Sementara itu, pada antrian 2, terdapat pelanggan Nekyo yang memilih layanan konsultasi dengan estimasi waktu tunggu 10 menit. Dengan membandingkan total waktu tunggu kedua antrian, terlihat bahwa antrian 2 memiliki waktu tunggu yang lebih singkat. Oleh karena itu, pelanggan baru akan secara otomatis diarahkan ke antrian 2.



Gambar 3. Bukti algoritma sudah berjalan dengan baik.

Pada gambar tersebut ketika total waktu tunggu pada antrian 2 melebihi total waktu tunggu pada antrian 1, pelanggan berikutnya secara otomatis akan dialihkan kembali ke antrian 1.



Gambar 4. Pembagian antrian berjalan baik.

Pada gambar tersebut etika layanan pelanggan selesai, tombol "Layani Pelanggan" dapat digunakan untuk menyelesaikan pelayanan pada antrian tertentu sesuai nomor antriannya.

```
class AntrianBank:
> def __init__(self): ...
>
> def isEmpty(self, receptionist): ...
>
> def tambahCustomer(self, receptionist, customer): ...
>
> def getTotalWaitingTime(self, receptionist):
    total_time = 0
    for customer in self.queues[receptionist]:
        total_time += self.serviceTime[customer.serviceType] + customer.additionalTime
    return total_time
```

Gambar 5. kode estimasi waktu.

Penggunaan algoritma First In First Out sering menghadapi masalah yang disebut Convoy Effect, di mana pelanggan dengan waktu layanan singkat terjebak di belakang pelanggan dengan waktu layanan yang sangat panjang. Namun, sistem yang membagi antrian menjadi dua berdasarkan estimasi waktu tunggu paling singkat dapat membantu mengurangi dampak dari masalah ini.

```
def addCustomerToQueue(self, name, service_type, amount, additional_time):
    waiting_time_1 = self.bank.getTotalWaitingTime(1)
    waiting_time_2 = self.bank.getTotalWaitingTime(2)
```

Gambar 6. Kode pembagian customer ke antrian berdasarkan estimasi waktu.

Dengan implementasi pembagian antrian ini, sistem dapat memprioritaskan pelanggan berdasarkan estimasi waktu tunggu, sehingga memberikan efisiensi yang lebih baik dalam melayani pelanggan dan mengurangi efek negatif dari Convoy Effect. Secara keseluruhan, struktur data deque yang diimplementasikan pada penelitian ini memberikan efisiensi tinggi pada operasi penambahan dan penghapusan elemen di kedua ujungnya dengan kompleksitas waktu $O(1)$. Hal ini disebabkan oleh desain deque yang memungkinkan manipulasi langsung tanpa memerlukan pergeseran elemen, menjadikannya pilihan ideal untuk berbagai aplikasi seperti pengelolaan antrian. Meskipun efisien, deque memiliki keterbatasan pada penggunaan memori tambahan dan pencarian elemen yang tetap memerlukan waktu $O(n)$.

5. KESIMPULAN DAN SARAN

Sebagai kesimpulan, penelitian ini berhasil mengembangkan sistem antrian bank berbasis deque

dengan logika First In First Out (FIFO) yang memberikan efisiensi tinggi pada pengelolaan layanan administrasi dan antrian. Sistem ini memberikan pengalaman pengguna yang lebih baik dengan memprioritaskan pelanggan secara adil dan efisien, sekaligus meningkatkan performa operasional secara keseluruhan. Sebagai saran, penelitian selanjutnya dapat mengintegrasikan fitur tambahan seperti pelacakan waktu nyata untuk memberikan estimasi waktu tunggu yang lebih akurat dan dinamis.

DAFTAR PUSTAKA

- [1] S. S. Winarsih and A. W. Wahono, "Implementasi Dan Pengujian Struktur Data Berbasis Acuan Untuk Program Aplikasi Mengungkap Kepribadian Berdasarkan Tanggal Lahir Dan Nama," *J. Teknol. Inf. dan Komun.*, vol. 10, no. 2, p. 11, 2022, doi: 10.30646/tikomsin.v10i2.631.
- [2] W. Alakel, "Sistem Informasi Akuntansi Persediaan Obat Metode First in First Out (Studi Kasus: Rumah Sakit Bhayangkara Polda Lampung)," *J. Tekno Kompak*, vol. 13, no. 1, p. 36, 2019, doi: 10.33365/jtk.v13i1.269.
- [3] P. Gas *et al.*, "Kata Kunci: FIFO; Agile; Blackbox Testing; PGN; Website ;," vol. 10, pp. 1–10, 2024.
- [4] M. B. Nendya, B. Susanto, G. I. W. Tamtama, and T. J. Wijaya, "Desain Level Berbasis Storyboard Pada Perancangan Game Edukasi Augmented Reality Tap The Trash," *Fountain Informatics J.*, vol. 8, no. 1, pp. 1–6, 2023, doi: 10.21111/fij.v8i1.8836.
- [5] E. Saputro, A. Syarief, and R. D. W. Utomo, "Graphical User Interface dalam Perspektif Pelindungan Desain Industri di Indonesia," *J. Seni Reka Ranc.*, vol. 6, no. 2, pp. 255–268, 2023, doi: 10.25105/jsrr.v6i2.17553.
- [6] Rizaldy Gunawan, H. Yuana, and S. Kirom, "Implementasi Metode Queue Pada Sistem Antrian Online Berbasis Web Studi Kasus Uptd Puskesmas Sananwetan," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 7, no. 3, pp. 1538–1545, 2023, doi: 10.36040/jati.v7i3.6687.
- [7] Muhammad Romzi and B. Kurniawan, "Pembelajaran Pemrograman Python Dengan Pendekatan Logika Algoritma," *JTIM J. Tek. Inform. Mahakarya*, vol. 03, no. 2, pp. 37–44, 2020.