

Pengembangan Aplikasi Menu Kafe Menggunakan Bahasa Pemrograman C++ Berbasis Code::Blocks

Muhammad Wildan Ihsanmawla¹

¹Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Kota Semarang

Artikel Info

Kata kunci:

pemrograman C++
sistem pemesanan *cafe*
struktur data *array*
modularitas program
pembelajaran berbasis proyek

ABSTRAK

Perkembangan teknologi informasi telah mendorong terjadinya transformasi digital di berbagai sektor industri, termasuk usaha kafe yang membutuhkan sistem pemesanan terkomputerisasi guna meningkatkan efisiensi dan ketepatan operasional. Penelitian ini bertujuan untuk mengembangkan sebuah prototipe sistem pemesanan kafe menggunakan bahasa pemrograman C++ dengan menerapkan konsep-konsep fundamental pemrograman prosedural. Sistem dikembangkan menggunakan pendekatan perancangan *top-down* yang mengintegrasikan struktur data *array* untuk penyimpanan data menu dan transaksi, penggunaan fungsi dan prosedur untuk mendukung modularitas program, struktur percabangan untuk merepresentasikan logika bisnis, serta struktur perulangan untuk menangani operasi yang bersifat repetitif. Hasil implementasi berupa aplikasi berbasis antarmuka baris perintah dengan total 95 baris kode yang mampu menangani pemilihan kategori menu, pemesanan berulang dalam satu sesi transaksi, perhitungan otomatis total pembelian dengan penerapan sistem diskon bertingkat (10% untuk pembelian $\geq$ Rp100.000 dan 5% untuk pembelian $\geq$ Rp50.000), serta pencetakan struk pembayaran secara terperinci. Hasil pengujian menunjukkan bahwa sistem berhasil menjalankan fungsi utama sesuai kebutuhan dengan tingkat akurasi perhitungan yang baik pada skenario penggunaan normal. Sistem ini mampu mendemonstrasikan integrasi berbagai konsep dasar pemrograman dalam konteks aplikatif yang relevan dengan permasalahan bisnis sederhana. Penelitian ini memberikan kontribusi sebagai model pembelajaran pemrograman berbasis proyek serta menyediakan kerangka dasar untuk pengembangan sistem *point of sale* yang lebih komprehensif pada penelitian selanjutnya.

Penulis Korespondensi :

Muhammad Wildan Ihsanmawla,
Program Studi Teknik Informatika
Fakultas Ilmu Komputer
Universitas Dian Nuswantoro, Kota Semarang, 50131
Email: 111202516361@mhs.dinus.ac.id

1. PENDAHULUAN

Revolusi digital dalam dua dekade terakhir telah mengubah paradigma operasional berbagai sektor industri, termasuk industri jasa makanan dan minuman [1]. Transformasi dari sistem manual ke sistem terkomputerisasi menawarkan peningkatan efisiensi, akurasi, dan kecepatan operasional [2]. Seiring pertumbuhan pesat usaha *café* di Indonesia, kebutuhan akan sistem pemesanan yang efisien menjadi semakin penting guna meningkatkan kepuasan pelanggan dan efektivitas proses bisnis [3].

Namun demikian, banyak *café* skala kecil hingga menengah masih mengandalkan sistem pemesanan manual yang menimbulkan berbagai permasalahan [4]. Pencatatan pesanan secara manual berisiko menimbulkan kesalahan pencatatan, ketidakakuratan perhitungan harga, serta inkonsistensi data transaksi yang berpotensi menyebabkan kerugian finansial [5]. Selain itu, proses manual cenderung memperlambat pelayanan, terutama pada jam sibuk, sehingga berdampak negatif pada kualitas layanan.

Keterbatasan sistem manual juga terlihat pada proses perhitungan pembayaran, penerapan diskon, dan pengelolaan data penjualan [6]. Perhitungan manual meningkatkan risiko kesalahan matematis, sementara skema diskon yang kompleks sulit diterapkan secara konsisten. Kesalahan-kesalahan tersebut dapat berdampak pada menurunnya kepercayaan pelanggan serta potensi kerugian finansial bagi pelaku usaha. Selain itu, pada jam operasional yang sibuk, proses manual sering kali memperlambat pelayanan dan menimbulkan antrian panjang, sehingga memengaruhi pengalaman pelanggan secara keseluruhan. Di sisi lain, kesulitan dalam melakukan rekapitulasi dan analisis data penjualan menghambat pengambilan keputusan strategis oleh manajemen. Data transaksi yang tidak terdokumentasi secara terstruktur menyulitkan manajemen dalam melakukan evaluasi performa penjualan, analisis tren produk, maupun perencanaan strategi promosi. Padahal, dalam era persaingan bisnis yang semakin ketat, kemampuan untuk mengolah data menjadi informasi yang relevan merupakan faktor kunci dalam pengambilan keputusan strategis. Dengan demikian, implementasi sistem pemesanan berbasis komputer menjadi kebutuhan yang mendesak, terutama bagi usaha mikro, kecil, dan menengah (UMKM) yang ingin bertransformasi secara digital secara bertahap dan berbiaya terjangkau.

Dalam konteks pendidikan tinggi, khususnya bidang ilmu komputer dan sistem informasi, pengembangan aplikasi pemesanan *café* sederhana menjadi sarana pembelajaran yang relevan dan kontekstual [7]. Bahasa pemrograman C++ dipilih karena mendukung paradigma pemrograman prosedural dan berorientasi objek, serta efektif untuk mengajarkan konsep dasar seperti struktur data, kontrol alur program, dan modularitas kode [8]. Sebagai bahasa tingkat menengah, C++ memberikan keseimbangan antara kontrol terhadap performa dan abstraksi pemrograman [9]. Penggunaan *array*, fungsi dan prosedur, struktur percabangan, serta perulangan memberikan fondasi yang kuat bagi mahasiswa untuk memahami mekanisme kerja sistem informasi dan logika pemrosesan data secara sistematis [10].

Penelitian ini mengembangkan sebuah *prototype* sistem pemesanan *café* berbasis *command-line interface* (CLI) menggunakan bahasa C++. Sistem dirancang untuk mengotomatisasi proses pemesanan, perhitungan total pembayaran, dan penerapan diskon, sehingga mampu mengurangi *human error* dan meningkatkan efisiensi operasional. Selain aspek teknis, penelitian ini menekankan integrasi konsep fundamental pemrograman dalam penyelesaian permasalahan bisnis nyata. Kontribusi utama penelitian ini terletak pada penerapan model pembelajaran berbasis proyek yang menghubungkan teori pemrograman dengan aplikasi praktis. Studi kasus pemesanan *café* memungkinkan mahasiswa memahami sinergi antara struktur data, algoritma kontrol, dan modularitas kode. *Prototype* yang dihasilkan juga dapat dikembangkan lebih lanjut menjadi sistem *point of sale* (POS) dengan fitur tambahan seperti inventori dan pelaporan penjualan.

Berdasarkan permasalahan tersebut, penelitian ini merumuskan fokus kajian meliputi perancangan arsitektur sistem pemesanan *café* menggunakan C++, implementasi *array* untuk pengelolaan data menu dan pesanan, penerapan modularitas melalui fungsi dan prosedur, serta penggunaan struktur percabangan dan perulangan untuk menangani logika bisnis dan operasi repetitif. Tujuan penelitian ini mencakup pengembangan *prototype* sistem pemesanan *café* berbasis CLI yang akurat dan efisien, serta demonstrasi penerapan konsep-konsep dasar pemrograman C++ dalam konteks aplikatif. Selain itu, penelitian ini menganalisis efektivitas pendekatan pemrograman prosedural, efisiensi algoritma, dan modularitas kode, khususnya dalam perhitungan transaksi dan penerapan skema diskon. Diharapkan, penelitian ini memberikan kontribusi akademis dan praktis. Secara akademis, penelitian ini menyediakan media pembelajaran kontekstual untuk meningkatkan pemahaman dan kemampuan berpikir komputasional mahasiswa. Secara praktis, *prototype* yang dikembangkan dapat menjadi solusi awal berbiaya rendah bagi UMKM *café* dalam mengadopsi sistem pemesanan terkomputerisasi, sekaligus menjadi dasar pengembangan sistem POS yang lebih komprehensif di masa mendatang.

2. METODE

Bab 2 ini disusun untuk memberikan landasan teoretis dan konseptual yang menjadi dasar dalam pelaksanaan penelitian. Pembahasan pada bab ini tidak hanya menguraikan konsep dan teori yang relevan dengan sistem pemesanan *café* dan pemrograman menggunakan bahasa C++, tetapi juga menjelaskan keterkaitan antar konsep yang digunakan dalam perancangan dan implementasi sistem. Landasan ini diperlukan agar proses pengembangan sistem dilakukan secara sistematis, terstruktur, dan sesuai dengan prinsip-prinsip rekayasa perangkat lunak.

Untuk mendukung tujuan tersebut, subbab-subbab berikut akan membahas secara rinci teori dan konsep yang berkaitan langsung dengan penelitian. Pembahasan diawali dengan kajian mengenai sistem pemesanan dan karakteristik operasional *café* sebagai konteks permasalahan, kemudian dilanjutkan dengan konsep dasar pemrograman C++ yang meliputi struktur data, struktur kontrol, serta modularitas program. Melalui pemaparan ini, diharapkan pembaca memperoleh pemahaman yang komprehensif mengenai dasar teoretis yang melandasi metode dan implementasi sistem yang akan dijelaskan pada bab-bab selanjutnya.

2.1 Bahasa Pemrograman C++

Bahasa pemrograman C++ merupakan hasil pengembangan dari bahasa C yang diperkenalkan oleh Bjarne Stroustrup pada tahun 1979 di Bell Laboratories [11]. Pengembangan ini bertujuan untuk memperluas kapabilitas bahasa C dengan menambahkan dukungan terhadap paradigma pemrograman berorientasi objek, tanpa menghilangkan karakteristik efisiensi dan kedekatan bahasa C terhadap perangkat keras [11]. Dengan pendekatan tersebut, C++ dirancang sebagai bahasa yang mampu menjembatani kebutuhan pemrograman tingkat rendah dan tingkat tinggi dalam satu kesatuan bahasa. Secara konseptual, C++ diklasifikasikan sebagai bahasa pemrograman tingkat menengah (*middle-level language*), karena menyediakan fasilitas pemrograman tingkat rendah seperti manipulasi memori dan *pointer*, sekaligus abstraksi tingkat tinggi melalui mekanisme kelas, objek, dan *template* [12]. Kombinasi ini memungkinkan pengembang untuk mengontrol penggunaan sumber daya sistem secara langsung, namun tetap mendukung pengembangan aplikasi berskala besar dan kompleks. Oleh karena itu, C++ dikenal sebagai bahasa yang memiliki performa tinggi serta efisiensi yang baik dalam penggunaan memori dan waktu eksekusi [13].

C++ mendukung berbagai paradigma pemrograman, termasuk pemrograman prosedural, pemrograman berorientasi objek, dan pemrograman generik [14]. Dukungan terhadap berbagai paradigma tersebut memberikan fleksibilitas dalam perancangan solusi perangkat lunak, sehingga pengembang dapat memilih pendekatan yang paling sesuai dengan karakteristik permasalahan yang dihadapi [15]. Dalam konteks pemrograman prosedural pada penelitian ini, C++ menyediakan mekanisme fungsi dan prosedur yang memungkinkan pemecahan program ke dalam modul-modul kecil yang relatif independen, sehingga meningkatkan keterbacaan, kemudahan pemeliharaan, dan potensi penggunaan ulang kode program. Keunggulan C++ dalam konteks pembelajaran pemrograman terletak pada kemampuannya memperkenalkan konsep-konsep fundamental secara eksplisit, seperti manajemen memori, penggunaan *pointer* dan *reference*, serta pemahaman struktur data dasar [16]. Berbeda dengan bahasa pemrograman tingkat tinggi yang banyak mengabstraksi detail teknis, C++ menuntut pemahaman menyeluruh terhadap mekanisme kerja program pada level yang lebih dekat dengan mesin [16]. Pendekatan ini berkontribusi dalam membangun fondasi berpikir komputasional yang kuat bagi mahasiswa atau pemula dalam bidang ilmu komputer.

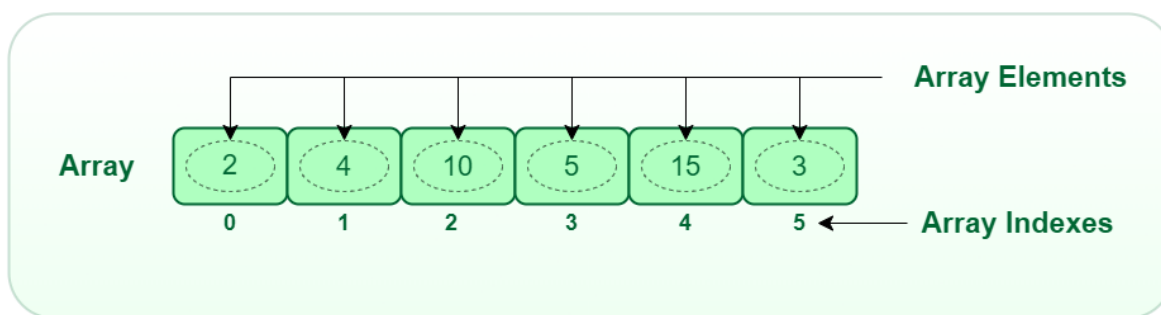


Gambar 1 Logo C++

Selain itu, C++ menyediakan Standard Template Library (STL) yang berisi kumpulan kelas dan fungsi generik untuk mengimplementasikan struktur data serta algoritma umum, seperti vector, list, map, dan berbagai algoritma pencarian serta pengurutan [17]. Meskipun demikian, dalam pembelajaran dasar pemrograman, implementasi manual struktur data dan algoritma sering kali tetap diajarkan untuk memperkuat pemahaman konseptual sebelum memanfaatkan abstraksi yang disediakan oleh pustaka standar. Dalam pengembangan aplikasi bisnis sederhana, seperti sistem pemesanan kafe, C++ menawarkan keunggulan berupa kecepatan eksekusi, efisiensi penggunaan memori, serta tingkat portabilitas yang tinggi [18]. Program C++ dapat dikompilasi dan dijalankan pada berbagai sistem operasi dengan dukungan *compiler* yang luas, seperti GNU Compiler Collection (GCC), Clang, dan Microsoft Visual C++ [18]. Dukungan terhadap standar C++ terkini juga menjamin kompatibilitas, keamanan, serta stabilitas aplikasi yang dikembangkan.

2.2 Struktur Data Array

Array merupakan salah satu struktur data fundamental dalam ilmu komputer yang digunakan untuk menyimpan sekumpulan elemen bertipe data homogen pada lokasi memori yang berurutan atau kontigu [19]. Sebagai tipe data abstrak, *array* menyediakan mekanisme akses elemen berdasarkan indeks tertentu dengan kompleksitas waktu $O(1)$, sehingga sangat efisien untuk operasi pengambilan data secara langsung (*direct access*) [19]. Karakteristik ini menjadikan *array* sebagai struktur data yang banyak digunakan dalam pengembangan perangkat lunak dasar [20]. Sifat alokasi memori yang kontigu pada *array* memungkinkan sistem dan kompiler melakukan optimasi dalam *offset* perhitungan alamat memori setiap elemen [21]. Setiap elemen *array* dapat diakses melalui perhitungan *offset* berdasarkan indeks, tanpa memerlukan proses penelusuran elemen satu per satu. Namun demikian, keterbatasan utama *array* terletak pada ukuran yang bersifat statis dan harus ditentukan sejak awal, sehingga kurang fleksibel untuk kebutuhan data yang dinamis. Dalam bahasa pemrograman C++, *array* dideklarasikan dengan menentukan tipe data elemen serta ukuran *array* secara eksplisit. Mekanisme pengindeksan pada C++ menggunakan sistem indeks berbasis nol (*zero-based indexing*), di mana elemen pertama berada pada indeks 0 dan elemen terakhir berada pada indeks $n - 1$ [21]. Pemahaman terhadap mekanisme pengindeksan ini sangat penting untuk menghindari kesalahan akses di luar batas *array* (*out-of-bounds error*), yang dapat menyebabkan perilaku program tidak terdefinisi.



Gambar 2 Struktur Array [22]

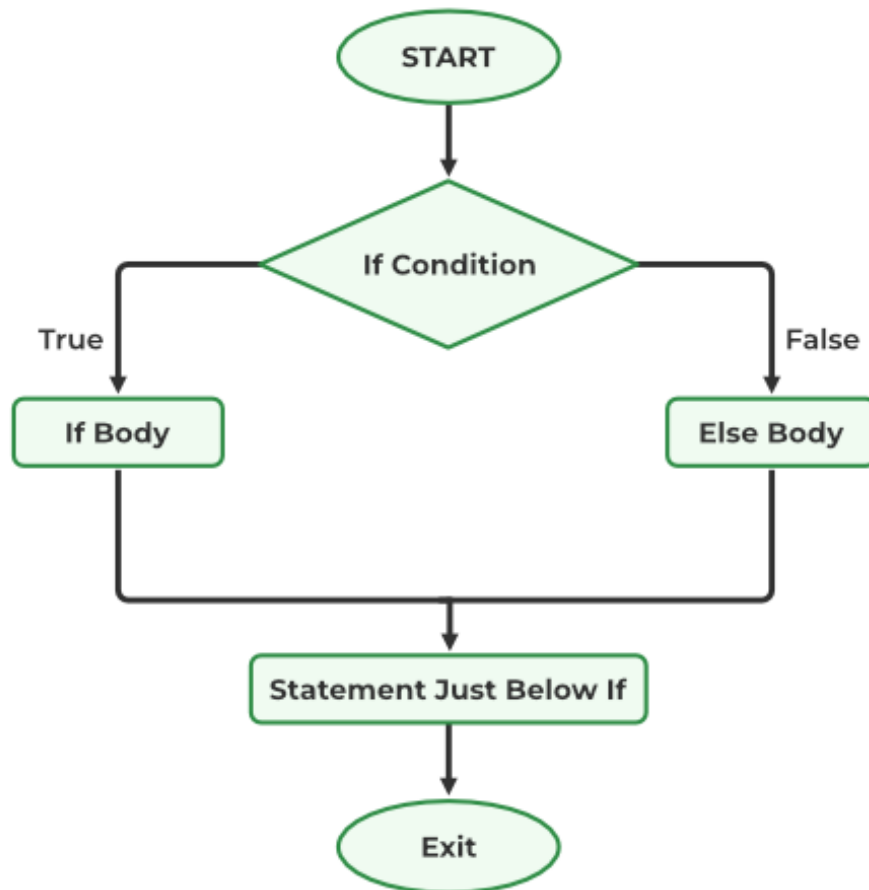
Dalam sistem pemesanan kafe yang dikembangkan pada penelitian ini, *array* dimanfaatkan untuk menyimpan data menu dan data transaksi secara terstruktur. *Array* bertipe *string* digunakan untuk menyimpan daftar nama menu makanan dan minuman, sedangkan *array* bertipe numerik digunakan untuk menyimpan informasi harga masing-masing menu. Korespondensi antara nama menu dan harga dijaga melalui penggunaan indeks yang sama pada kedua *array* tersebut. Selain itu, data transaksi pesanan disimpan menggunakan pendekatan *parallel arrays*, di mana setiap atribut pesanan seperti nama item, jumlah pesanan, dan sub total yang disimpan dalam *array* terpisah dengan indeks yang saling berkorespondensi. Pendekatan ini dipilih karena kesederhanaan implementasinya serta kemudahannya untuk dipahami dalam konteks pembelajaran pemrograman dasar. Meskipun kurang fleksibel dibandingkan struktur data yang lebih kompleks seperti *struct* atau *class*, metode ini dinilai cukup efisien untuk skala data kecil hingga menengah.

2.3 Modularitas Program: Fungsi dan Prosedur

Konsep modularitas merupakan prinsip fundamental dalam rekayasa perangkat lunak yang menekankan pembagian program ke dalam unit-unit fungsional yang lebih kecil, dan dapat digunakan kembali [23]. Pendekatan modular pada penelitian ini bertujuan untuk mengurangi kompleksitas program secara keseluruhan serta meningkatkan keteraturan struktur kode [24]. Dalam bahasa pemrograman C++, modularitas diwujudkan melalui penggunaan fungsi dan prosedur [25]. Pada penelitian ini, fungsi didefinisikan sebagai sub program yang menerima parameter masukan, melakukan pemrosesan tertentu, dan mengembalikan nilai sebagai hasil keluaran. Dengan menggunakan fungsi, logika pemrosesan dapat dikapsulkan sehingga detail implementasi tersembunyi dari bagian program yang memanggilnya. Prosedur yang diimplementasikan sebagai fungsi bertipe void, digunakan untuk sub program yang tidak mengembalikan nilai. Prosedur umumnya digunakan untuk operasi yang bersifat menghasilkan efek samping, seperti menampilkan keluaran ke layar atau memodifikasi data melalui parameter referensi. Dalam sistem pemesanan kafe, fungsi digunakan untuk melakukan perhitungan total dan diskon, sedangkan prosedur digunakan untuk menampilkan menu dan mencetak struk pembayaran.

2.4 Struktur Kontrol Percabangan

Struktur kontrol percabangan memungkinkan program untuk mengeksekusi alur instruksi yang berbeda berdasarkan hasil evaluasi suatu kondisi logika [26]. Dalam bahasa C++, percabangan diimplementasikan melalui konstruksi `if`, `if-else`, `if-else if-else`, dan `switch-case`. Pada sistem pemesanan kafe, struktur percabangan digunakan dalam berbagai skenario, seperti validasi *input* pengguna, pemilihan kategori menu, serta penerapan logika diskon bertingkat. Validasi input bertujuan untuk memastikan bahwa pilihan pengguna berada dalam rentang yang valid, sehingga mencegah kesalahan akses data.



Gambar 3 Logika If-Else Pada Pemrograman [27]

Logika diskon diimplementasikan menggunakan struktur `if-else if` untuk memastikan bahwa hanya satu tingkat diskon yang diterapkan sesuai dengan total pembelian. Selain itu, percabangan juga digunakan untuk menangani kondisi khusus, seperti pesanan kosong atau *input* yang tidak sesuai. Penerapan struktur percabangan yang tepat meningkatkan keandalan sistem serta mencegah terjadinya kesalahan eksekusi.

2.5 Struktur Kontrol Perulangan

Struktur kontrol perulangan memungkinkan eksekusi suatu blok instruksi secara berulang selama kondisi tertentu terpenuhi, sehingga mengurangi duplikasi kode dan mendukung pengolahan data dalam jumlah variatif [28]. Bahasa C++ menyediakan tiga jenis perulangan utama, yaitu `for`, `while`, dan `do-while`. Dalam sistem pemesanan kafe, perulangan digunakan untuk menampilkan daftar menu, memproses beberapa pesanan dalam satu transaksi, serta menghitung akumulasi sub total pesanan. Penggunaan `for` digunakan untuk iterasi *array* dengan jumlah elemen yang telah diketahui, sedangkan `while` dan `do-while` digunakan untuk mengelola alur interaksi pengguna hingga proses pemesanan selesai. Penggunaan struktur perulangan yang tepat memungkinkan program berjalan secara efisien dan terkontrol. Selain itu, penentuan kondisi penghentian

perulangan yang jelas sangat penting untuk mencegah terjadinya *infinite loop* yang dapat menyebabkan program berhenti merespons.

3. PEMBAHASAN HASIL

Bab 3 menyajikan pembahasan hasil penelitian yang diperoleh dari proses perancangan, implementasi, dan pengujian sistem pemesanan *café* berbasis *command-line interface* menggunakan bahasa pemrograman C++. Pada bab ini, hasil implementasi tidak hanya dipaparkan secara deskriptif, tetapi juga dianalisis untuk menilai kesesuaian antara rancangan sistem, penerapan konsep pemrograman, dan tujuan penelitian yang telah dirumuskan sebelumnya.

Untuk mempermudah pemahaman, subbab-subbab berikut akan menguraikan pembahasan secara sistematis, dimulai dari penjelasan struktur dan alur kerja sistem, implementasi masing-masing komponen fungsional, hingga analisis hasil pengujian dan evaluasi kinerja sistem. Melalui pembahasan ini, diharapkan pembaca dapat memperoleh gambaran yang jelas mengenai bagaimana konsep teoretis yang telah dijelaskan pada bab sebelumnya diimplementasikan dalam bentuk sistem nyata serta sejauh mana sistem tersebut mampu menjawab permasalahan yang menjadi fokus penelitian.

3.1 Hasil Implementasi Sistem

Implementasi sistem pemesanan kafe menggunakan bahasa pemrograman C++ berhasil direalisasikan dengan mengintegrasikan seluruh konsep dasar pemrograman yang telah dibahas pada bab sebelumnya. Sistem yang dikembangkan berbentuk aplikasi berbasis *command-line interface* (CLI) yang bersifat fungsional dan mampu menangani alur transaksi secara menyeluruh, mulai dari pemilihan menu, pencatatan pesanan, perhitungan otomatis, hingga pencetakan struk pembayaran digital. Hasil implementasi menunjukkan bahwa pendekatan pemrograman prosedural dengan pemanfaatan struktur data *array*, fungsi, prosedur, struktur percabangan, dan perulangan mampu menghasilkan solusi yang efektif untuk mendukung proses pemesanan pada lingkungan kafe sederhana.

```
PROGRAM PEMESANAN CAFE

===== MENU CAFE =====
1. Minuman
2. Makanan
3. Keluar
Pilih kategori (1-3):
```

Gambar 4 Antarmuka Sistem

Program yang dihasilkan terdiri dari sekitar 95 baris kode yang disusun secara modular dengan pembagian tanggung jawab yang relatif jelas antar bagian program. Implementasi memanfaatkan pustaka `<iostream>` untuk menangani proses masukan dan keluaran standar, serta penggunaan namespace `std` untuk menyederhanakan sintaksis pemrograman. Seluruh sistem dibangun dalam satu berkas sumber (*single-file architecture*), yang dinilai sesuai dengan ruang lingkup aplikasi pembelajaran dengan tingkat kompleksitas menengah dan tujuan utama untuk demonstrasi konsep dasar pemrograman.

Dari sisi fungsionalitas, sistem telah memenuhi seluruh kebutuhan yang ditetapkan pada tahap perancangan. Sistem mampu menampilkan menu dalam dua kategori utama, yaitu minuman dan makanan, dengan masing-masing kategori menyediakan tiga item pilihan. Setiap item dilengkapi informasi harga yang jelas serta sistem penomoran untuk memudahkan proses pemilihan. Selain itu, sistem mendukung pemesanan lebih dari satu item dalam satu sesi transaksi, melakukan perhitungan sub total secara otomatis untuk setiap pesanan, menghitung total keseluruhan pembelian, menerapkan sistem diskon bertingkat berdasarkan total belanja, serta menghasilkan struk pembayaran yang memuat rincian pesanan dan perhitungan finansial secara lengkap.

3.2 Analisis Komponen Penyimpanan Data

Komponen penyimpanan data pada sistem ini diimplementasikan menggunakan *array* statis yang dideklarasikan sebagai variabel global agar dapat diakses oleh seluruh fungsi dalam program. *Array* string minuman[3] = {"Es Teh", "Kopi Susu", "Jus Jeruk"} digunakan untuk menyimpan nama menu minuman, sedangkan *array* paralel int hargaMinuman[3] = {15000, 20000, 18000} digunakan untuk menyimpan informasi harga yang berkorespondensi dengan masing-masing item minuman melalui mekanisme indeks yang sama.

Pendekatan serupa diterapkan pada kategori makanan dengan deklarasi string makanan[3] = {"Nasi Goreng", "Mie Goreng", "Kentang Goreng"} dan int hargaMakanan[3] = {25000, 20000, 15000}. Pemisahan data menu ke dalam kategori yang berbeda memberikan sejumlah keuntungan dari sudut pandang perancangan perangkat lunak. Pertama, pemisahan ini memungkinkan pengelolaan data yang lebih terstruktur dan independen. Kedua, pendekatan tersebut memudahkan pengembangan lanjutan apabila diperlukan penambahan kategori menu baru. Ketiga, struktur data ini merepresentasikan model domain bisnis kafe yang pada umumnya mengelompokkan produk berdasarkan jenisnya.

Tabel 1 Implementasi *Array* Pada Menu Kafe

main.cpp
string minuman[3] = {"Es Teh", "Kopi Susu", "Jus Jeruk"};
int hargaMinuman[3] = {15000, 20000, 18000};
string makanan[3] = {"Nasi Goreng", "Mie Goreng", "Kentang Goreng"};
int hargaMakanan[3] = {25000, 20000, 15000};
string pesananNama[20];
int pesananJumlah[20];
int pesananSubtotal[20];
int indexPesanan = 0;

Untuk menyimpan data transaksi, sistem menggunakan tiga *array* paralel dengan kapasitas maksimum 20 elemen, yaitu pesananNama[20] untuk menyimpan nama item yang dipesan, pesananJumlah[20] untuk menyimpan jumlah pesanan, dan pesananSubtotal[20] untuk menyimpan nilai sub total hasil perhitungan harga dan jumlah. Kapasitas ini dipilih berdasarkan estimasi jumlah maksimum item berbeda yang mungkin dipesan dalam satu transaksi, sehingga memberikan ruang yang cukup tanpa alokasi memori yang berlebihan. Variabel indexPesanan digunakan sebagai penunjuk indeks sekaligus penghitung jumlah pesanan yang telah dicatat. Variabel ini diinisialisasi dengan nilai nol dan akan bertambah setiap kali pesanan baru dimasukkan. Penggunaan satu variabel indeks untuk seluruh *array* paralel memastikan sinkronisasi data, di mana setiap indeks merepresentasikan satu entitas pesanan dengan atribut yang saling berkaitan. Dari sisi penggunaan memori, alokasi *array* statis pada sistem ini relatif kecil dan tidak menimbulkan beban berarti. Total penggunaan memori berada pada kisaran beberapa *kilobyte*, sehingga sangat efisien bahkan untuk sistem dengan sumber daya terbatas. Meskipun pendekatan *array* paralel tidak sefleksibel penggunaan struktur data seperti *struct* atau *class*, metode ini memberikan kesederhanaan yang bernilai dalam konteks pembelajaran pemrograman dasar.

3.3 Analisis Komponen Fungsi dan Prosedur

Sistem mengimplementasikan dua prosedur utama dan satu fungsi perhitungan untuk mendukung modularitas program. Prosedur tampilMenu() bertanggung jawab menampilkan antarmuka menu utama yang menyediakan pilihan kategori kepada pengguna. Prosedur ini menggunakan serangkaian perintah keluaran untuk menampilkan menu dengan pemisah visual, sehingga struktur tampilan menjadi lebih jelas dan mudah dipahami oleh pengguna. Prosedur tampilMenu() dirancang mengikuti prinsip *single responsibility*, di mana prosedur ini hanya menangani aspek tampilan tanpa mencampurkan logika bisnis atau manipulasi data. Prosedur ini tidak menerima parameter maupun mengembalikan nilai, sehingga berfungsi murni sebagai komponen presentasi. Prosedur ini dipanggil berulang kali dalam perulangan utama program untuk menampilkan ulang menu setelah setiap proses pemesanan.

Tabel 2 Implementasi Komponen Fungsi dan Prosedur

main.cpp
void tampilMenu() {

```

cout << "\n===== MENU CAFE =====\n";
cout << "1. Minuman\n";
cout << "2. Makanan\n";
cout << "3. Keluar\n";
}

void footer() {
    cout << "===== Toko Kopi Rengganis =====< endl;
    cout << "Social:" << endl;
    cout << "IG: @tokokopirengganis" << endl;
    cout << "WA: 08112516361" << endl;
}

int hitungTotal(int harga, int jumlah) {
    return harga * jumlah;
}

```

Prosedur kedua, yaitu footer(), berfungsi menampilkan bagian penutup yang memuat informasi kafe seperti nama usaha dan pesan penutup. Prosedur ini dipanggil pada akhir eksekusi program sebagai penutup transaksi. Informasi yang ditampilkan masih bersifat statis dan dituliskan langsung di dalam kode, yang dapat diterima untuk prototipe, namun pada sistem produksi sebaiknya dikelola melalui konfigurasi terpisah. Fungsi hitungTotal(int harga, int jumlah) berperan sebagai fungsi murni yang melakukan perhitungan sub total dengan mengalikan harga satuan dengan jumlah pesanan. Fungsi ini tidak memiliki efek samping dan hasil keluarannya sepenuhnya ditentukan oleh parameter masukan. Karakteristik ini menjadikan fungsi mudah diuji, dapat digunakan ulang, serta meningkatkan kejelasan kode karena maksud perhitungan tersampaikan secara eksplisit melalui nama fungsi.

3.4 Analisis Logika Program dan Alur Kendali

Fungsi utama (main) mengendalikan alur eksekusi program menggunakan struktur perulangan do-while yang memungkinkan pengguna melakukan pemesanan berulang dalam satu sesi. Sejumlah variabel dideklarasikan untuk menyimpan masukan pengguna, akumulasi total belanja, nilai diskon, serta penanda pengulangan program. Di dalam perulangan utama, sistem menampilkan menu kategori dan memproses pilihan pengguna menggunakan struktur percabangan if-else. Ketika pengguna memilih kategori minuman atau makanan, sistem menampilkan daftar menu terkait dengan memanfaatkan perulangan for untuk mengakses elemen *array*. Penomoran menu ditampilkan menggunakan indeks *array* yang disesuaikan agar lebih intuitif bagi pengguna. Setelah pengguna memilih item dan jumlah pesanan, sistem melakukan validasi sederhana terhadap pilihan yang dimasukkan. Apabila valid, sistem menghitung sub total menggunakan fungsi hitungTotal(), menyimpan data pesanan ke dalam *array* transaksi, serta menampilkan sub total sebagai umpan balik langsung kepada pengguna. Apabila pilihan tidak valid, sistem menampilkan pesan kesalahan dan mengizinkan pengguna mengulangi proses.

Tabel 3 Implementasi Perhitungan Diskon

main.cpp
<pre> if (total >= 100000) { diskon = total * 10 / 100; } else if (total >= 50000) { diskon = total * 5 / 100; } else { diskon = 0; } totalBayar = total - diskon; </pre>

Perhitungan diskon dilakukan setelah proses pemesanan selesai dengan menggunakan struktur percabangan bertingkat. Diskon sebesar 10% diberikan untuk total belanja minimal Rp100.000, diskon 5%

untuk total belanja minimal Rp50.000, dan tidak ada diskon untuk total di bawah nilai tersebut. Struktur ini memastikan hanya satu tingkat diskon yang diterapkan sesuai prioritas yang benar.

3.5 Evaluasi Output dan Antarmuka Pengguna

Keluaran sistem dirancang secara bertahap mengikuti alur transaksi, dimulai dari tampilan judul program, menu kategori, daftar menu, hingga struk pembayaran akhir. Penggunaan pemisah visual dan format yang konsisten membantu meningkatkan keterbacaan dan pemahaman pengguna terhadap informasi yang ditampilkan.

```

===== MENU CAFE =====
1. Minuman
2. Makanan
3. Keluar
Pilih kategori (1-3): 2

--- MENU MAKANAN ---
1. Nasi Goreng (Rp 25000)
2. Mie Goreng (Rp 20000)
3. Kentang Goreng (Rp 15000)
Pilih makanan: 1
Jumlah: 2
Subtotal: Rp 50000

Pesan lagi? (y/n): n

===== STRUK PEMBAYARAN =====
1. Es Teh x3 = Rp 45000
2. Nasi Goreng x2 = Rp 50000
-----
Total          : Rp 95000
Potongan       : Rp 4750
Total Bayar    : Rp 90250
Terima kasih sudah berkunjung!

===== Toko Kopi Rengganis =====
Social:
IG: @tokokopirengganis
WA: 08112516361

```

Gambar 5 Output Program

Struk pembayaran disajikan dalam format terstruktur yang mencakup daftar pesanan, sub total, diskon, dan total pembayaran akhir. Informasi ini ditampilkan secara transparan sehingga pengguna dapat memahami proses perhitungan yang dilakukan oleh sistem. Meskipun antarmuka berbasis teks memiliki keterbatasan dari sisi visual, pendekatan ini tetap efektif untuk tujuan pembelajaran dan kompatibel pada berbagai platform.

3.6 Analisis Kelebihan dan Keterbatasan Sistem

Sistem yang dikembangkan memiliki sejumlah kelebihan, antara lain kesederhanaan implementasi, keterbacaan kode yang baik, efisiensi penggunaan sumber daya, serta kemudahan pengembangan lanjutan untuk tujuan pembelajaran. Sistem ini efektif dalam mendemonstrasikan konsep dasar pemrograman dalam konteks permasalahan nyata. Di sisi lain, sistem memiliki keterbatasan pada aspek skalabilitas, persistensi data, antarmuka pengguna, serta keamanan. Penggunaan *array* statis membatasi jumlah data yang dapat ditangani,

dan tidak adanya penyimpanan permanen membuat sistem hanya cocok untuk penggunaan sesi tunggal. Keterbatasan ini perlu diperhatikan apabila sistem dikembangkan lebih lanjut ke arah aplikasi produksi.

4. KESIMPULAN

Berdasarkan hasil penelitian dan implementasi sistem pemesanan kafe menggunakan bahasa pemrograman C++, dapat disimpulkan bahwa seluruh tujuan penelitian telah tercapai dengan baik. Penelitian ini berhasil menunjukkan integrasi konsep-konsep fundamental pemrograman yang meliputi arsitektur prosedural, struktur data *array*, modularitas fungsi dan prosedur, serta struktur kendali ke dalam solusi aplikatif yang relevan dengan permasalahan bisnis sederhana. Arsitektur sistem yang dibangun menerapkan prinsip modularitas dan pemisahan tanggung jawab, menghasilkan kode yang ringkas, mudah dipahami, dan sesuai untuk tujuan pembelajaran, sekaligus memiliki potensi untuk dikembangkan lebih lanjut. Dari sisi implementasi teknis, penggunaan *array* paralel terbukti efektif untuk mengelola data menu dan transaksi dengan efisiensi waktu dan memori yang baik, meskipun memerlukan pengelolaan indeks yang cermat. Pemanfaatan fungsi dan prosedur berhasil memisahkan logika penyajian, komputasi, dan alur program, sehingga mendukung keterbacaan, kemudahan pengujian, serta *code reusability*. Selain itu, penerapan struktur percabangan dan perulangan mampu merepresentasikan logika bisnis secara akurat, mulai dari validasi masukan, pemilihan menu, penerapan diskon bertingkat, hingga pengelolaan pemesanan berulang dalam satu sesi penggunaan. Sebagai pengembangan ke depan, sistem ini masih memiliki ruang peningkatan, khususnya pada aspek validasi masukan, skalabilitas struktur data, penyimpanan data permanen, dan peningkatan antarmuka pengguna. Penggunaan struktur data dinamis, penerapan penyimpanan berbasis berkas atau basis data, serta refaktorisasi kode akan meningkatkan keandalan dan fleksibilitas sistem. Dengan pengembangan lanjutan tersebut, *prototype* ini berpotensi berkembang menjadi solusi pemesanan kafe yang lebih komprehensif, sekaligus tetap mempertahankan nilai utamanya sebagai media pembelajaran pemrograman berbasis proyek yang efektif dan aplikatif.

REFERENCES

- [1] W. Kurniady, E. Sofiati, and S. Sindrawati, "Strategi Inovasi Bisnis Food & Beverage dan Peran Digital Bisnis dalam Pengembangan Kewirausahaan di Hanan Academy," *J. Ekon. Dan Stat. Indones.*, vol. 5, no. 1, pp. 123–133, May 2025, doi: 10.11594/jesi.05.01.14.
- [2] M. Faris Hafizh, R. Dhever Hani, A. Nur Kholishah, and I. Farida Adi Prawira, "STRATEGI TRANSFORMASI DIGITAL DI ERA INDUSTRI 4.0: BLUEPRINT BISNIS, PENERAPAN TEKNOLOGI, DAN PERAN KRITIS PEMERINTAH DALAM MENINGKATKAN DAYA SAING BISNIS FOOD AND BEVERAGE (F&B)," *Ekon. Bisnis*, vol. 23, no. 1, pp. 1–8, Jun. 2024, doi: 10.32722/eb.v23i1.6383.
- [3] E. Hadinata, D. Irwan, A. M. Elhanafi, and D. Handoko, "Pelatihan Strategi Marketing Pemanfaatan Content Creator Pada Bisnis Food and Beverage Studi Kasus Bisnis Madu Hutan," *J. Has. Pengabd. Masy. JURIBMAS*, vol. 4, no. 1, pp. 137–148, Jul. 2025, doi: 10.62712/juribmas.v4i1.460.
- [4] A. Dera, Sutrisno, and R. Febrianti, "The Effect of Current Ratio, Price to Book Value, and Debt to Equity Ratio on Return on Assets (in Food and Beverage Companies Listed on the Indonesia Stock Exchange in 2021–2023)," *Ilmu Ekon. Manaj. Dan Akunt.*, vol. 6, no. 2, pp. 102–123, Sep. 2025, doi: 10.37012/ileka.v6i2.2969.
- [5] S. Sudiyono, A. D. Baihaqie, V. Hapsari, and A. T. Hapsari, "Efektivitas Pencatatan Keuangan Digital Bagi Pelaku UMKM Sektor Food and Beverage Sentra Industri Makanan Banyubiru Kabupaten Semarang," *Edusight J. Pengabd. Masy.*, vol. 2, no. 1, Feb. 2025, doi: 10.69726/edujpm.v2i1.77.
- [6] A. N. Rais *et al.*, "Implementasi Sistem Informasi Food and Beverage Online Shop Dengan Metode Waterfall Yang Dimodifikasi," *EVOLUSI J. Sains Dan Manaj.*, vol. 10, no. 1, Mar. 2022, doi: 10.31294/evolusi.v10i1.12053.
- [7] Danu Abilsyah Aimar and Tantry Hidayati Sinaga, "Penerapan Sistem Informasi Penjualan Dessert Berbasis Web dengan Pendekatan POAC," *Neptunus J. Ilmu Komput. Dan Teknol. Inf.*, vol. 3, no. 3, pp. 201–218, Aug. 2025, doi: 10.61132/neptunus.v3i3.989.
- [8] M. Mundhir, "Implementasi Perhitungan Arah Kiblat Menggunakan Metode Vincenty dalam Bahasa Pemrograman C++," *Astroislamica J. Islam. Astron.*, vol. 4, no. 2, pp. 199–219, Dec. 2025, doi: 10.47766/astroislamica.v4i2.6005.
- [9] Adawiyah Ritonga and Yahfizham Yahfizham, "Studi Literatur Perbandingan Bahasa Pemrograman C++ dan Bahasa Pemrograman Python pada Algoritma Pemrograman," *J. Tek. Inform. Dan Teknol. Inf.*, vol. 3, no. 3, pp. 56–63, Nov. 2023, doi: 10.55606/jutiti.v3i3.2863.
- [10] S. Z. Akhmad, "Analisis Keefektifan Bahasa Pemrograman C++ Sebagai Dasar Pembelajaran Untuk Mahasiswa Teknik Informatika," *Pelita Teknol.*, vol. 18, no. 1, pp. 33–37, Mar. 2023, doi: 10.37366/pelitatekno.v18i1.3335.
- [11] P. Shah, A. Bhatt, and R. Kuksal, "C Programming: Your First Step in C Programming," ASP Books (A division of AcademicSP), May 2025, doi: 10.63686/978-81-987694-8-0.
- [12] "A Comparative Study of the Significance of Different Programming Languages," *Acad. J. Comput. Inf. Sci.*, vol. 8, no. 2, 2025, doi: 10.25236/AJCIS.2025.080206.

- [13] M. Roynard, E. Carlinet, and T. Géraud, "A Modern C++ Point of View of Programming in Image Processing," in *Proceedings of the 21st ACM SIGPLAN International Conference on Generative Programming: Concepts and Experiences*, Auckland New Zealand: ACM, Nov. 2022, pp. 164–171. doi: 10.1145/3564719.3568692.
- [14] K. A. Onyango and G. M. Wambugu, "Comparative Analysis on the Evaluation of the Complexity of C, C++, Java, PHP and Python Programming Languages based on Halstead Software Science," Mar. 2023, Accessed: Jan. 19, 2026. [Online]. Available: <http://repository.mut.ac.ke:8080/xmlui/handle/123456789/6419>
- [15] A. Nazri, H. Liyana, I. Hasanah, and Z. Ibrahim, "Tales of C++ Worlds: C++ Programming Language Game-Based Learning," *Multidiscip. Appl. Res. Innov.*, vol. 5, no. 1, pp. 223–231, Jan. 2024.
- [16] J. Reinders, B. Ashbaugh, J. Brodman, M. Kinsner, J. Pennycook, and X. Tian, *Data Parallel C++: Programming Accelerated Systems Using C++ and SYCL*. Springer Nature, 2023. doi: 10.1007/978-1-4842-9691-2.
- [17] J. Othman, S. A. Mohamed Yusoff, A. F. Rosmani, and M. S. Abu Bakar, "Structural differences of constructor methods in object-oriented paradigms: a comparative study using C++, Java and Python / Jamal Othman ... [et al.]," *Boundaries Multidimens. Horiz. E-Learn.*, vol. 9, pp. 165–170, Mar. 2025.
- [18] M. Roynard, "Generic programming in modern C++ for Image Processing," Theses, Sorbonne Université, 2022. Accessed: Jan. 19, 2026. [Online]. Available: <https://theses.hal.science/tel-03922670>
- [19] H. Song *et al.*, "Experimental Demonstration of an Integrated Broadband Pixel-Array Structure Generating Two Tunable Orbital-Angular-Momentum Mode Values and Carrying 100-Gbit/s QPSK Data," *Conf. Lasers Electro-Opt., p. SM4C.3*, 2021, doi: 10.1364/CLEO_SI.2021.SM4C.3.
- [20] J. Yu, H. Hwang, H. Kim, and W. Y. Choi, "Vertically Stackable Memcapacitor Crossbar Array Based on NAND Flash Array Structure," *2024 IEEE Int. Electron Devices Meet. IEDM*, pp. 1–4, Dec. 2024, doi: 10.1109/IEDM50854.2024.10873480.
- [21] B. Harrison, J. Necaie, A. Projansky, and J. D. Whitfield, "A Sierpinski Triangle Data Structure for Efficient Array Value Update and Prefix Sum Calculation," 2024, doi: 10.48550/ARXIV.2403.03990.
- [22] "Implement Arrays in different Programming Languages," GeeksforGeeks. Accessed: Jan. 19, 2026. [Online]. Available: <https://www.geeksforgeeks.org/dsa/implement-arrays-in-different-programming-languages/>
- [23] X. Chen and X. Wang, "Computational Thinking Training and Deep Learning Evaluation Model Construction Based on Scratch Modular Programming Course," *Comput. Intell. Neurosci.*, vol. 2023, no. 1, p. 3760957, Jan. 2023, doi: 10.1155/2023/3760957.
- [24] R. Saha *et al.*, "Modular programming of interaction and geometric specificity enables assembly of complex DNA origami nanostructures," *Nat. Commun.*, vol. 16, no. 1, p. 11392, Dec. 2025, doi: 10.1038/s41467-025-66195-9.
- [25] P. Zhao, Y. Zhang, Y. Chen, and Y. Jia, "Modular Programming in C for MCU," *2025 5th Int. Conf. Electron. Inf. Eng. Comput. Sci. EIECS*, pp. 1254–1259, Sep. 2025, doi: 10.1109/EIECS67708.2025.11283466.
- [26] K. Kunz, J. Wacker, T. Wenzel, A. Schumacher, and K. Tsarava, "Programming Structures and Their Misconceptions Across Modalities: The Pilot Study of a Code Tracing Assessment For Talented Primary School Students," *Proc. 25th Koli Call. Int. Conf. Comput. Educ. Res.*, pp. 1–12, Nov. 2025, doi: 10.1145/3769994.3770040.
- [27] "C if else Statement," GeeksforGeeks. Accessed: Jan. 19, 2026. [Online]. Available: <https://www.geeksforgeeks.org/c/c-if-else-statement/>
- [28] A. Meyer and K. Winklmann, "program looping in regular dynamic logic," 2022. Accessed: Jan. 19, 2026. [Online]. Available: <https://www.semanticscholar.org/paper/program-looping-in-regular-dynamic-logic-Meyer-Winklmann/cac3365b3e67c95907d5b502dae5d9b787c564a3>