**PRAKTIKUM**

**ALGORITMA DAN PEMROGRAMAN LANJUT**

**SEMESTER GENAP T.A. 2024/2025**

**LAPORAN PROYEK AKHIR**



**DISUSUN OLEH :**

**NAMA/NIM : Adiel Khairullah / 123240221**

**Liu Luqyana / Liu Luqyana**

**KELAS/PLUG : IF-K**

**NAMA ASISTEN : Panji Arif Jafarudin**

**Bagas Duta Prasetya**

**PROGRAM STUDI INFORMATIKA**

**JURUSAN INFORMATIKA**

**FAKULTAS TEKNIK INDUSTRI**

**UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”**

**YOGYAKARTA**

**2025**

# **HALAMAN PENGESAHAN**

**LAPORAN PROYEK AKHIR**



Disusun

oleh :

*Adiel Khairullah 123240221*

*Liu Luqyana 123240223*

Telah Diperiksa dan Disetujui oleh Asisten Praktikum Algoritma dan Pemrograman

Pada Tanggal : …

**Asisten**

**Praktikum**

**Panji Arif Jafarudin**

**NIM. 123220091**

**Asisten**

**Praktikum**

**Bagas Duta Prasetya**

**NIM. 123220000**

# **KATA PENGANTAR**

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa yang senantiasa mencurahkan rahmat dan hidayah-Nya sehingga kami dapat menyelesaikan Praktikum Pemrograman Web serta laporan akhir Pemrograman Web. Adapun laporan ini berisi tentang projek akhir semester dari hasil pembelajaran selama praktikum berlangsung.

Tidak lupa ucapan terima kasih kepada asisten dosen yang selalu membimbing dan mengajari kami dalam melaksanakan praktikum dan dalam menyusun laporan ini. Laporan ini masih sangat jauh dari kesempurnaan, oleh karena itu kritik serta saran yang membangun kami harapkan untuk menyempurnakan laporan akhir ini.

Atas perhatian dari semua pihak yang membantu penulisan ini, kami ucapkan terima kasih. Semoga laporan ini dapat dipergunakan seperlunya.

Yogyakarta, 30 Mei 2025

Penyusun

# **DAFTAR ISI**

[**HALAMAN PENGESAHAN 1**](#_30j0zll)

[**KATA PENGANTAR 2**](#_1fob9te)

[**DAFTAR ISI 3**](#_3znysh7)

[**BAB I 1**](#_2et92p0)

[**BAB II 2**](#_gx1sot4w5n0)

[**BAB III 3**](#_4d34og8)

[3.1. Implementasi Materi 3](#_2s8eyo1)

[3.2. Penjelasan Fitur 3](#_17dp8vu)

[**BAB IV 5**](#_26in1rg)

[4.1. Jadwal Pengerjaan 5](#_lnxbz9)

[4.2. Pembagian Tugas 5](#_35nkun2)

[**BAB V 6**](#_1ksv4uv)

[5.1 Kesimpulan 6](#_44sinio)

[5.2 Saran 6](#_2jxsxqh)

[**DAFTAR PUSTAKA 7**](#_z337ya)

# **BAB I**

**Perpustakaan**

Konsep dari program ini yaitu dirancang sebagai sebuah aplikasi sistem manajemen data buku perpustakaanmenggunakan bahasa pemrograman **C++**. Tujuan utama program adalah untuk membantu pengguna dalam melakukan pencatatan dan pengelolaan koleksi buku secara efisien, termasuk proses peminjaman dan pengembalian buku.

Struktur data yang digunakan mengandalkan struct dan array untuk menyimpan kumpulan data buku. Program ini dibangun dengan pendekatan modular melalui pemanfaatan fungsi-fungsi agar kode lebih terorganisir dan mudah dipelihara.

**Fitur-Fitur Utama :**

**1. Penambahan Data Buku**

Pengguna dapat menambahkan satu atau beberapa data buku secara sekaligus. Informasi yang dicatat meliputi:

* Kode buku
* Judul buku
* Nama pengarang
* Tahun terbit

Secara default, setiap buku yang baru ditambahkan akan berstatus "tersedia".

**2. Menampilkan Daftar Buku**

Fitur ini menyajikan seluruh koleksi buku dalam bentuk tabel yang rapi dan informatif. Informasi yang ditampilkan mencakup:

* Nomor urut
* Kode buku
* Judul
* Pengarang
* Tahun terbit
* Status (tersedia / dipinjam oleh siapa)

**3. Peminjaman Buku**

Pengguna dapat meminjam buku dengan cara menginputkan nama peminjam dan kode buku yang ingin dipinjam. Sistem akan mengecek ketersediaan buku dan, jika tersedia, akan mencatat bahwa buku tersebut sedang dipinjam serta menyimpan nama peminjam.

**4. Pengembalian Buku**

Fitur ini memungkinkan pengguna untuk mengembalikan buku yang telah dipinjam. Pengguna cukup memasukkan kode buku, kemudian sistem akan memperbarui status buku menjadi "tersedia" dan menghapus nama peminjam.

**5. Pengurutan Data Buku (Sorting)**

Program menyediakan kemampuan untuk mengurutkan data buku berdasarkan kriteria berikut:

* Kode buku
* Judul buku
* Tahun terbit

Dengan pilihan metode pengurutan:

* **Bubble Sort** (metode sederhana)
* **Shell Sort** (lebih efisien pada data besar)

Pengurutan dapat dilakukan dalam urutan:

* **Ascending (menaik)**
* **Descending (menurun)**

**Topik dan Konsep Pemrograman yang Digunakan**

**Struct dan Array of Struct**

Digunakan untuk merepresentasikan dan menyimpan kumpulan data buku secara terstruktur.

**Fungsi (Modular Programming)**

Seluruh operasi utama dipisahkan dalam fungsi-fungsi tersendiri untuk meningkatkan keterbacaan, efisiensi, dan pemeliharaan kode.

**Percabangan dan Perulangan**

Penggunaan struktur if-else dan switch-case untuk kontrol alur program, serta perulangan for dan while untuk mengelola proses input dan pemrosesan data.

**Algoritma Pengurutan (Sorting)**

* **Bubble Sort**: digunakan untuk pengurutan sederhana.
* **Shell Sort**: digunakan untuk efisiensi pengurutan yang lebih tinggi.

**Input dan Output**

Interaksi dengan pengguna dilakukan melalui cin, getline, dan cout. Format tampilan tabel ditingkatkan dengan penggunaan fungsi setw dari library <iomanip>

# **BAB II**

**TUJUAN PROYEK AKHIR**

Aplikasi manajemen data buku perpustakaan ini dikembangkan dengan tujuan utama untuk menyederhanakan, mempercepat, dan meningkatkan efisiensi proses pengelolaan data buku di lingkungan perpustakaan. Melalui digitalisasi sistem pencatatan, aplikasi ini hadir sebagai solusi untuk menggantikan metode manual yang rentan terhadap kesalahan, lambat, dan kurang terorganisir.

Secara garis besar, aplikasi ini memiliki beberapa tujuan penting sebagai berikut:

**1. Mewujudkan Sistem Pencatatan Buku yang Rapi dan Terstruktur**

Aplikasi ini memungkinkan pustakawan untuk mencatat dan menyimpan data buku secara sistematis, mencakup informasi seperti kode buku, judul, pengarang, dan tahun terbit. Dengan struktur data yang terorganisir, proses pencarian maupun pengelompokan buku menjadi jauh lebih mudah dan cepat.

**2. Mempermudah Pengelolaan Proses Peminjaman dan Pengembalian**

Fitur peminjaman dan pengembalian buku disediakan untuk memantau status ketersediaan buku serta mencatat siapa peminjamnya. Hal ini membantu mengurangi risiko kehilangan data, keterlambatan pengembalian, maupun terjadinya peminjaman ganda.

**3. Mendukung Akses Data yang Cepat dan Akurat**

Dengan dukungan fitur pencarian dan pengurutan (sorting) berdasarkan berbagai kriteria, seperti judul buku atau tahun terbit, pengguna dapat mengakses data yang dibutuhkan dengan efisien, tanpa harus memeriksa data secara manual satu per satu.

**4. Meningkatkan Efektivitas dan Produktivitas Pengelola Perpustakaan**

Aplikasi ini dirancang untuk meminimalkan waktu dan tenaga yang dibutuhkan dalam kegiatan administrasi perpustakaan. Dengan otomatisasi proses, pekerjaan yang sebelumnya membutuhkan waktu lama kini dapat diselesaikan hanya dalam hitungan detik.

**5. Menjadi Media Pembelajaran Konsep Dasar Pemrograman**

Selain berfungsi sebagai aplikasi fungsional, program ini juga bertujuan sebagai sarana latihan dalam mengimplementasikan berbagai konsep dasar pemrograman C++, seperti :

* Struktur data (struct dan array)
* Fungsi dan modularitas kode
* Sorting dan pencarian data
* Penggunaan percabangan dan perulangan
* Pengolahan input/output dari pengguna

# **BAB III**

**PEMBAHASAN**

## Implementasi Materi

**3.1.1 Struct**

|  |
| --- |
| struct Menu {  string nama;  int harga;  int stok;  }; |

**3.1.2 Pointer**

|  |
| --- |
| void tampilkanMenu(Menu \*menu, int jumlahMenu) {  for (int i = 0; i < jumlahMenu; i++) {  cout << i+1 << ". " << menu[i].nama << " - Rp" << menu[i].harga << " (" << menu[i].stok << " tersedia)" << endl;  }  } |

**3.1.3 Sorting (Pengurutan)**

|  |
| --- |
| void bubbleSort(Menu \*menu, int jumlahMenu) {  for (int i = 0; i < jumlahMenu - 1; i++) {  for (int j = 0; j < jumlahMenu - i - 1; j++) {  if (menu[j].harga > menu[j + 1].harga) {  swap(menu[j], menu[j + 1]);  }  }  }  } |

**3.1.4 Searching (Pencariantan)**

|  |
| --- |
| int cariMenu(Menu \*menu, int jumlahMenu, string keyword) {  for (int i = 0; i < jumlahMenu; i++) {  if (menu[i].nama.find(keyword) != string::npos) {  return i;  }  }  return -1;  } |

**3.1.5 File**

|  |
| --- |
| ofstream file("struk.txt");  file << "===== Struk Pembelian =====" << endl;  file << "Total: Rp" << totalHarga << endl;  file.close(); |

## Penjelasan Fitur

3.2.1 Fitur 1

Fungsi ini meminta input dari pengguna untuk mengisi data buku (kode, judul, pengarang, tahun). Data dimasukkan ke dalam array daftarBuku[] dan ditandai sebagai tersedia.

|  |
| --- |
| for(int i = 0; i < jumlah; i++) {          cout << "\n==========";          cout << "\nBuku ke-" << i + 1;          cout << "\n==========\n";          cout << "Masukkan Kode Buku   : "; getline(cin, daftarBuku[jmlhBuku].kode);          cout << "Masukkan Judul Buku   : "; getline(cin, daftarBuku[jmlhBuku].judul);          cout << "Masukkan Pengarang    : "; getline(cin, daftarBuku[jmlhBuku].pengarang);          cout << "Masukkan Tahun Terbit : "; cin >> daftarBuku[jmlhBuku].tahunTerbit;          cin.ignore();          daftarBuku[jmlhBuku].tersedia = true;          daftarBuku[jmlhBuku].peminjam = "-";          jmlhBuku++;      } |

3.2.2 Fitur 2

Fungsi ini mencetak seluruh daftar buku dalam array, menampilkan semua informasi dan status “Tersedia” atau “Dipinjam”.

|  |
| --- |
| for (int i = 0; i < jmlhBuku; i++) {          cout << setw(5) << i + 1  // Nomor urut               << setw(15) << daftarBuku[i].kode // Kode Buku               << setw(20) << daftarBuku[i].judul // Judul Buku               << setw(20) << daftarBuku[i].pengarang // Pengarang               << setw(15) << daftarBuku[i].tahunTerbit // Tahun Terbit               << setw(15) << (daftarBuku[i].tersedia ? "Tersedia" : "Dipinjam oleh " + daftarBuku[i].peminjam) // Status               << endl;      }      cout << "------------------------------------------------------------------------------------------\n\n "; |

3.2.3 Fitur 3

|  |
| --- |
| if (daftarBuku[i].kode == kodeCari) {              ditemukan = true;              if (daftarBuku[i].tersedia) {                  daftarBuku[i].peminjam = namaPeminjam;                  daftarBuku[i].tersedia = false;                  cout << "Buku dengan judul \"" << daftarBuku[i].judul << "\" berhasil dipinjam oleh " << namaPeminjam << "!\n";                  cout << "======================================================\n\n";              } else {                  cout << "Maaf, buku sedang dipinjam oleh " << daftarBuku[i].peminjam << "!\n\n";              }              break;          } |

Memungkinkan pengguna meminjam buku berdasarkan kode buku.

3.2.4 Fitur 4

Potongan kode berikut adalah bagian dari fungsi pengembalian buku dalam program manajemen perpustakaan. Tujuannya adalah untuk mengecek apakah buku dengan kode tertentu ada dalam daftar, dan jika iya, apakah sedang dipinjam agar bisa dikembalikan.

|  |
| --- |
| for (int i = 0; i < jmlhBuku; i++) {          if (daftarBuku[i].kode == kode) {              ditemukan = true;              if (!daftarBuku[i].tersedia) {                  daftarBuku[i].tersedia = true;                  daftarBuku[i].peminjam = "-";                  cout << "Buku berhasil dikembalikan!\n";              } else {                  cout << "Buku ini belum dipinjam.\n";              }              break;          }      } |

3.2.5 Fitur 5

Buku bisa diurutkan berdasarkan:

* Judul
* Kode
* Tahun terbit
* Dengan algoritma Bubble Sort dan Shell Sort, baik secara ascending maupun descending.

Bubble Sort :

|  |
| --- |
| if ((ascending && daftarBuku[j].judul > daftarBuku[j + 1].judul ) ||  (!ascending && daftarBuku[j].judul < daftarBuku[j + 1].judul)) {  swap(daftarBuku[j], daftarBuku[j + 1]);  } |

Shell Sort

|  |
| --- |
| for (j = i; j >= gap && ((ascending && daftarBuku[j - gap].kode > temp.kode) ||  (!ascending && daftarBuku[j - gap].kode < temp.kode)); j -= gap) {  daftarBuku[j] = daftarBuku[j - gap];  } |

## Screenshot Program

Masukan screenshot program selengkap mungkin, beri caption tiap gambar.

## Source Code Lengkap

|  |
| --- |
| *Format: source code pro 10pt* |

# **BAB IV**

**JADWAL PENGERJAAN TUGAS DAN PEMBAGIAN TUGAS**

## Jadwal Pengerjaan

**Tabel 4.1** Tabel Jadwal Pengerjaan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Kegiatan** | **November** | | | |
| **1** | **2** | **3** | **4** |
| 1. | Mengkaji Masalah |  |  |  |  |
| 2. | Perancangan Sistem Program |  |  |  |  |
| 3. | … |  |  |  |  |
| 4. | … |  |  |  |  |

## Pembagian Tugas

Kami membagi tugas sesuai dengan tanggung jawab masing masing setiap kegiatan.

**Tabel 4.2** Tabel Pembagian Tugas

|  |  |  |
| --- | --- | --- |
| **No.** | **Kegiatan** | **Penanggung Jawab** |
| 1. | Mengkaji Masalah |  |
| 2. | Perancangan Sistem Program |  |
|  | … |  |
|  | .. |  |

# 

# **BAB V**

**KESIMPULAN DAN SARAN**

## Kesimpulan

….

## Saran

…..

# **DAFTAR PUSTAKA**

**Tambahkan link repo github disini**

Gunakan format APA 7 untuk sitasi yang lain.

**KETENTUAN :**

1. Huruf Times New Roman 12 (Hitam)
2. Keterangan tabel dan gambar Times New Roman 11 (Hitam)
3. Isi tabel Times New Roman 10 (Hitam)
4. Isi source code Source code pro 10 (Hitam)
5. Margin Top Left 3, Right Bottom 2,5
6. Spacing 1,5 dan remove space before and after paragraph
7. Penomoran halaman pada setiap awal bab di bawah tengah, setiap isi bab di atas kanan, dan halaman cover tanpa halaman.
8. Tulisan dalam format laporan ini yang diberi warna merah hanya berupa tanda dan dapat disesuaikan.