

LAPORAN TUGAS BESAR
PRAKTIKUM ALGORITMA DAN PEMROGRAMAN
APLIKASI PENGELOLAAN DATA *CROWDFUNDING* SEDERHANA

Disusun untuk memenuhi tugas mata kuliah:

Praktikum Algoritma dan Pemrograman

Dosen pengampu:

Aaz Muhammad Hafidz Azis



Kelas: DS-48-02

disusun oleh kelompok Rendang Keju:

Muhammad Al Abrar 103052400034

Annisa Putri Nur Sa'id 103052400032

Muhammad Ammar Ridho 103052400009

PRODI S1 DATA SAINS
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY BANDUNG

2025

BAB 1

PENDAHULUAN

1. Latar Belakang

Tugas Besar ini merupakan implementasi dari materi yang telah dipelajari selama satu semester dalam mata kuliah Algoritma dan Pemrograman. Dalam tugas ini, kelompok kami diminta untuk membuat sebuah aplikasi sistem *crowdfunding* menggunakan bahasa pemrograman Go (*Golang*). Sistem *crowdfunding* merupakan aplikasi yang cukup kompleks untuk menerapkan berbagai konsep pemrograman yang telah dipelajari, namun masih dapat dipahami dan diimplementasikan.

2. Tujuan

Tujuan dari tugas besar ini adalah:

- a) Menerapkan konsep-konsep pemrograman dasar yang telah dipelajari dalam konteks nyata
- b) Mengimplementasikan struktur data *array* dan *struct* dalam bahasa *Go*
- c) Menerapkan algoritma pencarian dan pengurutan sederhana pada data aplikasi
- d) Membuat aplikasi yang dapat mengelola data pengguna, kampanye, dan donasi secara terintegrasi
- e) Memahami penggunaan *flow control* dan fungsi dalam pemrograman *Go*
- f) Meningkatkan kemampuan kerjasama kelompok dalam pengembangan *software*

3. Ruang Lingkup

Aplikasi yang dibuat adalah sistem *crowdfunding* sederhana yang memungkinkan:

- a) Pendaftaran dan *login* pengguna dengan dua peran berbeda (*admin* dan donatur)
- b) Pembuatan dan pengelolaan kampanye penggalangan dana oleh *admin*
- c) Sistem donasi yang memungkinkan donatur memberikan kontribusi
- d) Fitur pencarian dan pengurutan data untuk kemudahan navigasi
- e) Laporan donasi dan prediksi pencapaian target untuk analisis

BAB 2

DESKRIPSI TUGAS BESAR

1. Gambaran Umum Aplikasi

Sistem *Crowdfunding* yang dikembangkan kelompok kami adalah aplikasi berbasis terminal yang memungkinkan pengguna untuk membuat kampanye penggalangan dana dan menerima donasi dari donatur. Aplikasi ini dibuat menggunakan bahasa pemrograman *Go* dengan pendekatan pemrograman prosedural yang telah dipelajari dalam mata kuliah. Sistem ini dirancang untuk memberikan pengalaman yang mudah digunakan meskipun berbasis *command-line interface*.

2. Struktur Data

Struktur Data yang digunakan dalam proses pembuatan program *crowdfunding*:

a) Struktur Data Pengguna

Kelompok kami menggunakan struktur data untuk menyimpan informasi pengguna yang mencakup *email*, *username*, *password*, dan peran. Struktur ini memungkinkan sistem untuk membedakan akses antara *admin* yang dapat membuat kampanye dan donatur yang dapat memberikan donasi.

b) Struktur Data Kampanye

Untuk menyimpan informasi kampanye penggalangan dana, kami menggunakan struktur data yang mencakup identifikasi unik, judul, kategori, deskripsi, target dana, jumlah yang terkumpul, persentase *progress*, dan status kampanye. Struktur ini memungkinkan *tracking progress* secara *real-time*.

c) Struktur Data Donasi

Setiap transaksi donasi dicatat menggunakan struktur data yang mencatat ID kampanye tujuan, nama donatur, dan jumlah donasi. Hal ini memungkinkan pelacakan dan pelaporan yang akurat.

3. Fitur-Fitur Aplikasi

Fitur dan Fungsi pada program *Crowdfunding*:

a) Sistem Autentikasi

Kelompok kami mengimplementasikan sistem autentikasi yang komprehensif dengan pendaftaran pengguna yang mencakup validasi *email*, *username*, dan *password* yang kuat. Sistem *login* memungkinkan pengguna masuk menggunakan email atau username dengan *password* yang telah terdaftar. Kami juga menerapkan *role-based access* yang membedakan akses antara *admin* dan donatur.

b) Manajemen Kampanye

Fitur manajemen kampanye memungkinkan *admin* untuk membuat kampanye baru dengan informasi lengkap. Sistem pencarian kampanye dapat dilakukan berdasarkan ID, judul, atau kategori untuk memudahkan navigasi. Kami juga menyediakan fitur

pengurutan kampanye berdasarkan *progress* dalam urutan *ascending* atau *descending*. Setiap kampanye menampilkan informasi detail termasuk *progress bar visual* yang memudahkan pemahaman status pencapaian.

c) Sistem Donasi

Proses donasi dirancang agar mudah digunakan oleh donatur untuk berdonasi ke kampanye yang masih aktif. Sistem ini dilengkapi dengan validasi untuk mencegah donasi yang melebihi target kampanye. Semua transaksi donasi tercatat dengan rapi untuk keperluan pelaporan dan audit.

d) Laporan dan Analisis

Kelompok kami menyediakan fitur laporan donasi yang menampilkan semua transaksi dengan berbagai opsi pengurutan. Sistem dapat menghitung total keseluruhan donasi dan memberikan prediksi jumlah transaksi yang dibutuhkan untuk mencapai target kampanye.

4. Algoritma

Algoritma yang diimplementasikan pada program *crowdfunding* sebagai berikut:

a) Algoritma Pencarian

Kelompok kami mengimplementasikan *Sequential Search* untuk mencari kampanye berdasarkan judul, yang cocok untuk dataset dengan ukuran kecil hingga menengah. *Binary Search* digunakan untuk pencarian berdasarkan ID dengan asumsi data telah terurut, memberikan efisiensi yang lebih baik.

b) Algoritma Pengurutan

Selection Sort diimplementasikan untuk mengurutkan kampanye berdasarkan *progress* pencapaian target. *Insertion Sort* digunakan untuk mengurutkan data donasi berdasarkan jumlah kontribusi.

5. Validasi Input

Aplikasi dilengkapi dengan sistem validasi yang komprehensif. Validasi *email* memastikan format yang benar dan *domain* yang valid seperti *gmail.com*, *yahoo.com*, atau *outlook.com*. *Username* harus memiliki panjang 4 - 25 karakter dan bersifat unik dalam sistem. *Password* harus minimal 8 karakter dengan kombinasi huruf besar, huruf kecil, angka, dan simbol untuk keamanan yang optimal. Peran pengguna dibatasi hanya pada "*admin*" atau "*donatur*" untuk menjaga konsistensi sistem.

BAB 3

PEMBAGIAN TUGAS KELOMPOK

1. Sistem Kampanye dan Integrasi (Muhammad Ammar Ridho)

Tanggung Jawab dan Tugas:

- a) Koordinasi keseluruhan proyek dan pembagian tugas
- b) Implementasi algoritma pencarian (*Sequential Search*, *Binary Search*)
- c) Pengembangan algoritma pengurutan (*Selection Sort* dan *Insertion Sort*)
- d) Pengembangan fitur pembuatan dan pengelolaan kampanye
- e) Integrasi semua fungsi aplikasi menjadi satu kesatuan
- f) *Debugging* dan *troubleshooting* masalah lintas modul

2. Sistem Autentikasi dan Donasi (Muhammad Al Abrar)

Tanggung Jawab dan Tugas:

- a) Desain dan implementasi struktur data untuk kampanye
- b) Pengembangan fungsi validasi *input* (*email*, *username*, *password*)
- c) Implementasi sistem pendaftaran dan *login* pengguna
- d) Integrasi sistem *role-based access*
- e) Implementasi sistem donasi dan validasinya
- f) *Testing* fitur donasi

3. Perancangan Design dan User Interface Terminal (Annisa Putri Nur Sai'd)

Tanggung Jawab dan Tugas:

- a) Integrasi proses donasi dengan update kampanye
- b) Pengembangan tampilan detail kampanye dengan *progress bar*
- c) Pengembangan fitur dan analisis donasi
- d) Implementasi menu navigasi dan *flow* aplikasi
- e) Desain *user interface* dan *user experience* aplikasi

BAB 4

TANTANGAN DAN SOLUSI

1. Tantangan Koordinasi Kelompok

a) Masalah

Setiap anggota mengembangkan bagian yang berbeda, sehingga integrasi kode menjadi rumit dan sering terjadi konflik dalam penggabungan fungsi.

b) Solusi

Kelompok kami membagi pekerjaan menjadi bagian-bagian kecil yang terpisah agar mudah dikerjakan. Kami juga sepakat untuk memberikan nama yang sama untuk hal-hal yang serupa dalam program, serta memastikan hasil kerja masing-masing anggota bisa digabungkan dengan baik.

2. Tantangan Validasi *Input*

a) Masalah

Implementasi validasi input yang komprehensif, terutama untuk *email* dan *password*, cukup kompleks karena harus memeriksa berbagai kondisi sekaligus. Kelompok kami mengalami kesulitan dalam membuat validasi yang efektif dan mudah dipahami.

b) Solusi

Kelompok kami memecah proses validasi menjadi fungsi-fungsi terpisah yang masing-masing fokus pada satu jenis validasi. Pendekatan ini membuat kode lebih mudah untuk *didebug*, dipelihara, dan dipahami oleh semua anggota tim.

3. Tantangan *User Interface Terminal*

a) Masalah

Interface terminal yang sederhana membuat aplikasi terlihat kurang menarik dan kadang membingungkan pengguna, terutama dalam navigasi antar menu.

b) Solusi

Kelompok kami menambahkan elemen visual seperti *progress bar* menggunakan karakter *Unicode*, *header* dan *separator* yang jelas untuk setiap menu, pesan konfirmasi dan *error* yang informatif, serta struktur menu yang konsisten di seluruh aplikasi.

BAB 5

TANTANGAN DAN SOLUSI

1. Kesimpulan

Tugas besar ini berhasil memenuhi seluruh tujuan yang ditetapkan, termasuk penerapan struktur data dan algoritma dasar dalam pengembangan aplikasi *crowdfunding* menggunakan bahasa *Go*. Aplikasi mampu mengelola data pengguna, kampanye, dan donasi secara terintegrasi, serta menyediakan fitur pencarian, pengurutan, dan laporan yang relevan. Melalui proyek ini, kelompok kami memperoleh pemahaman praktis mengenai desain program, validasi *input*, efisiensi algoritma, dan pentingnya pengalaman pengguna bahkan dalam aplikasi berbasis *terminal*. Proyek ini juga memperkuat keterampilan kolaborasi, perencanaan teknis, serta penerapan konsep pemrograman yang telah dipelajari.

2. Rekomendasi

Berdasarkan metode dan teknik yang telah diterapkan, kami berharap dapat mengembangkan lebih lanjut aplikasi *crowdfunding* dengan rekomendasi metode dan fitur sebagai berikut:

a) Penyimpanan Data Permanen

Aplikasi dapat ditingkatkan dengan menambahkan fitur penyimpanan data ke *file*. Data pengguna, kampanye, dan donasi dapat disimpan dalam format *CSV* atau *JSON* sehingga tidak hilang saat aplikasi ditutup. Implementasinya dapat menggunakan *package* yang tersedia di *Go* untuk *file handling*.

b) Peningkatan Kapasitas Data

Mengganti *array* dengan ukuran tetap menjadi *slice* yang dapat berkembang secara dinamis. Ini akan memungkinkan aplikasi menampung lebih banyak data tanpa batasan yang kaku dan memberikan fleksibilitas yang lebih baik.

c) Fitur Edit dan Hapus

Menambahkan fungsi untuk mengedit informasi kampanye dan menghapus kampanye yang tidak aktif. Ini akan membuat aplikasi lebih praktis untuk digunakan dalam situasi nyata di mana perubahan informasi sering diperlukan.

d) Peningkatan *Interface*

Meskipun tetap berbasis terminal, *interface* dapat diperbaiki dengan menambahkan warna pada teks, membuat tampilan yang lebih rapi dengan *formatting* yang konsisten, menambahkan konfirmasi sebelum melakukan aksi penting, dan menyediakan opsi navigasi yang lebih baik.

e) Fitur Keamanan Tambahan

Menambahkan enkripsi sederhana untuk *password* yang tersedia di *Go*. Hal ini akan meningkatkan keamanan sistem secara signifikan.

f) Sistem Notifikasi

Implementasi sistem notifikasi sederhana yang memberitahu admin tentang acara penting seperti pencapaian target kampanye atau donasi baru.

BAB 6

TAUTAN GITHUB

Repository kode program dapat diakses melalui tautan berikut:

https://github.com/alabrar12/tugas_besar_alpro_kelompok_RendangKeju.git

REFERENSI

- Atlassian. (2025) *Git and GitHub Best Practices for Team Collaboration*. Tersedia di: <https://www.atlassian.com/git> (Diakses: 26 Mei 2025).
- Documentation Go Official. (2025) *Package fmt*. Tersedia di: <https://golang.org/pkg/fmt/> (Diakses: 26 Mei 2025).
- Documentation Go Official. (2025) *Package math*. Tersedia di: <https://golang.org/pkg/math/> (Diakses: 26 Mei 2025).
- GeeksforGeeks. (2025) *Searching Algorithms*. Tersedia di: <https://www.geeksforgeeks.org/searching-algorithms/> (Diakses: 26 Mei 2025).
- GeeksforGeeks. (2025) *Sorting Algorithms*. Tersedia di: <https://www.geeksforgeeks.org/sorting-algorithms/> (Diakses: 26 Mei 2025).
- Kitabisa. (n.d.) *Dashboard Pengguna*. Tersedia di: <https://kitabisa.com/user> (Diakses: 26 Mei 2025).
- Ramadhan, K.A., Rahadi, N.W. dan Supriyono, A.R. (2020) 'Sistem informasi crowdfunding untuk kegiatan mahasiswa (Studi kasus: Star Laundry)', *Jurnal Politeknik Negeri Cilacap*, Politeknik Negeri Cilacap.
- Stack Overflow Community. (2025) *Go Programming Questions and Answers*. Tersedia di: <https://stackoverflow.com/questions/tagged/go> (Diakses: 26 Mei 2025).
- Syadzali, C. (2020) *Data mining menggunakan algoritma K-Nearest Neighbor untuk analisa perilaku pelanggan pada sistem layanan penggalangan dana online*. Tesis Magister. Universitas Diponegoro. Tersedia di: <https://eprints2.undip.ac.id/id/eprint/5859/> (Diakses: 26 Mei 2025).
- Tutorial Point. (2025) *Go Programming Tutorial*. Tersedia di: <https://www.tutorialspoint.com/go/> (Diakses: 26 Mei 2025).