

SKRIPSI

**PERKAKAS ANTARMUKA BARIS PERINTAH UNTUK
APLIKASI BERBASIS WEB KIRI**



Alfred Aprianto Liaunardi

NPM: 6181801014

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2023**

UNDERGRADUATE THESIS

**COMMAND LINE TOOL FOR KIRI WEB-BASED
APPLICATION**



Alfred Aprianto Liaunardi

NPM: 6181801014

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2023**

LEMBAR PENGESAHAN

**PERKAKAS ANTARMUKA BARIS PERINTAH UNTUK
APLIKASI BERBASIS WEB KIRI**

Alfred Aprianto Liaunardi

NPM: 6181801014

Bandung, 19 Januari 2023

Menyetujui,

Pembimbing

**Digitally signed
by Pascal
Alfadian Nugroho**

Pascal Alfadian, Nugroho, M.Comp.

Ketua Tim Penguji

**Digitally signed
by Vania Natali**

Vania Natali, M.T.

Anggota Tim Penguji

**Digitally signed
by Lionov**

Lionov, Ph.D.

Mengetahui,

Ketua Program Studi

**Digitally signed
by Mariskha Tri
Adithia**

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

PERKAKAS ANTARMUKA BARIS PERINTAH UNTUK APLIKASI BERBASIS WEB KIRI

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 19 Januari 2023



Alfred Aprianto Liaunardi
NPM: 6181801014

ABSTRAK

Project KIRI (atau KIRI saja) merupakan sebuah perangkat berbasis *web* yang dapat digunakan untuk membantu memudahkan penggunaannya untuk menggunakan angkutan kota (angkot). KIRI merealisasikan hal ini dengan cara menunjukkan kepada penggunaannya langkah-langkah yang harus dilakukan untuk pergi dari suatu lokasi ke lokasi lainnya, angkot-angkot mana yang harus dinaiki dalam rute tersebut, serta di mana pengguna harus menaiki atau turun dari angkot-angkot yang bersangkutan. Selain itu, KIRI juga dapat menunjukkan estimasi waktu dari rute-rute yang ditemukan.

Saat skripsi ini dibuat, KIRI hanya dapat diakses langsung melalui halaman *web*-nya, yang mengkategorikan KIRI sebagai aplikasi berbasis antarmuka pengguna grafis (GUI). GUI ini hanya merupakan satu dari dua jenis antarmuka perangkat lunak yang paling umum, dengan jenis satu laginya adalah antarmuka baris perintah (CLI). Secara singkat, bedanya adalah bahwa CLI memiliki tampilan yang lebih sederhana (tidak mengandung bantuan elemen-elemen grafis selain karakter/teksnya), dan memerlukan lebih sedikit sumber daya sistem untuk digunakan. Selain aplikasi berbasis *web*-nya, KIRI juga memiliki antarmuka pemrograman aplikasi (API) yang dapat digunakan untuk pengembangan perangkat lunak, akan tetapi, API KIRI hanya menerima masukan berupa permintaan GET, dan keluarannya berupa objek JSON, sehingga mungkin penggunaannya sulit dimengerti oleh pengguna-pengguna awam.

Dalam skripsi ini akan dibuat sebuah perangkat lunak berupa perangkat CLI yang dapat menjalankan fungsi-fungsi API KIRI. Perangkat ini akan dibuat dalam bahasa C untuk alasan keringanan perangkat, dengan menggunakan bantuan berbagai macam fungsi serta *library* eksternal, seperti *getopt*, *cJSON*, *cURL*, dan *CMake*, serta mengimplementasikan fitur-fitur API KIRI serta fitur-fitur dasar perangkat *command line*, seperti mode bantuan, dan *man page* (untuk Linux). Perangkat ini nantinya akan diuji coba dengan cara menguji satu-satu fungsinya, dan juga dengan menguji kasus-kasus umum untuk integrasinya dengan perangkat-perangkat lainnya yang sudah ada.

Hasil dari skripsi ini merupakan sebuah perangkat CLI yang dapat mengutilisasikan fungsi-fungsi API KIRI, serta memiliki fitur-fitur dasar perangkat CLI, yang meliputi sebuah halaman manual (berupa opsi *--help*) dan *man page*), pengeluaran pesan *error* yang sesuai dengan *error* yang terjadi, serta kemampuan integrasi keluaran perangkat dengan perangkat-perangkat CLI lainnya, seperti *findstr/grep*.

Kata-kata kunci: Navigasi, angkot, *Project KIRI*, *command line*, C

ABSTRACT

Project KIRI (or just KIRI) is a web-based tool which could be used to assist its users in utilizing *angkutan kota* (or *angkot*). KIRI does this by showing to its users the steps needed to go from one location to the other, which *angkots* would have to be taken within said route, as well as where they would have to board or get off of these *angkots*. Aside from that, KIRI also has the ability to show the estimated durations of the available routes.

At the time when this thesis was made, KIRI was only directly accessible from its webpage, which would categorize it as a graphical user interface (GUI) based application. GUI is one of the two most common types of interfaces, with the other being command line interface (CLI). In short, the differences are that CLI has significantly simpler appearances (which do not contain any graphical elements beyond its characters/texts), and requires less system resources to use. Aside from its web-based application, KIRI also has an application programming interface (API) which could be used for development purposes, but not only does it accept inputs only in the form of GET requests, but its outputs are only in the form of JSON objects, making it potentially hard to understand for the average users.

In this undergraduate thesis, a command line tool will be made, in which the tool would be able to run KIRI's API functions. Just like its general type, this tool will be run purely through the command line, and will not have any additional graphical images whatsoever in its interface. This newly-made tool will be made in C language, using various libraries such as getopt, cJSON, cURL, and CMake, along with implementing the KIRI API features, as well as the general features of a command line tool, such as a help mode, and a man page (for Linux). This tool will later be tested by testing each of its functions, as well as the general cases of its integration with other, existing command line tools.

The resulting program of this thesis is a command line tool which utilizes KIRI API's functions, and has the conventional features of a command line tool, including the inclusion of a manual page (--help option and *man* page), outputting the corresponding error messages for various errors, along with an output integration ability with other, existing command line tools, such as `findstr/grep`.

Keywords: Navigation, *angkot*, Project KIRI, command line, C

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan karunia-Nya, yang memungkinkan diselesaikannya penulisan skripsi yang berjudul “Perkakas *Command Line* KIRI” ini. Selama penulisan skripsi ini tentunya penulis menemui berbagai macam kesulitan serta menghadapi berbagai macam halangan, yang puji syukur dapat diselesaikan dengan baik dengan bantuan beberapa pihak. Pada kesempatan ini, penulis ingin mengucapkan terima kasih sebesar-besarnya kepada pihak-pihak tersebut, yaitu:

- Bapak Pascal Alfadian Nugroho, M.Comp. selaku dosen pembimbing yang telah membimbing serta membantu memberikan arahan kepada penulis dalam seluruh proses pembuatan skripsi ini.
- Ibu Vania Natali, M.T. serta Bapak Lionov, Ph. D. selaku dosen-dosen penguji yang telah memeriksa hasil skripsi ini serta memberikan kritik dan saran yang membangun.
- Teman-teman penulis—Daniel, Yalvi, Rama, JJ, serta teman-teman lainnya, yang telah memberikan dorongan, semangat, serta saran tambahan selama penulisan skripsi ini.
- Keluarga penulis, yang telah memberikan bantuan dan dukungan dalam seluruh aspek lainnya yang tidak berhubungan langsung dengan penulisan skripsi ini.
- Para staf tata usaha, baik dalam tata usaha FTIS maupun BAA, yang telah membantu penyelesaian semua urusan non-akademik dalam proses pembuatan skripsi ini.

Terakhir, penulis menyadari bahwa penulisan skripsi ini jauh dari sempurna, karena berbagai macam keterbatasan yang ada. Oleh karena itu, pada kesempatan ini penulis juga ingin mengucapkan permohonan maaf atas kekurangan-kekurangan yang ada. Walaupun begitu, penulis berharap skripsi ini dapat bermanfaat bagi seluruh pihak yang membacanya.

Bandung, Januari 2023

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xxi
DAFTAR KODE PROGRAM	xxiii
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan	3
1.4 Batasan Masalah	3
1.5 Metodologi	4
1.6 Sistematika Pembahasan	4
2 LANDASAN TEORI	5
2.1 KIRI	5
2.1.1 Tampilan	5
2.1.2 API	6
2.2 Pengantar <i>Command Line Interface</i>	11
2.2.1 <i>Command Line Interface</i> dan <i>Graphical User Interface</i>	12
2.2.2 <i>Command Line</i> di Linux	13
2.2.3 <i>Command Line</i> di Windows	15
2.3 Fungsi dan <i>Library</i> Bahasa C	18
2.3.1 <i>getopt</i>	18
2.3.2 <i>libcurl</i>	21
2.3.3 <i>cJSON</i>	23
2.3.4 <i>CMake</i>	27
3 ANALISIS	33
3.1 Analisis Aplikasi Sejenis	33
3.1.1 <i>Chrome Web Store Item Property CLI</i>	33
3.1.2 <i>iTunes Search API</i>	35
3.1.3 <i>Uber CLI</i>	36
3.1.4 <i>Google Maps Direction CLI</i>	39
3.2 Analisis API KIRI	40
3.2.1 <i>Search Place</i>	40
3.2.2 <i>Routing</i>	40
3.3 Analisis Perangkat yang Dibuat	41
3.3.1 Analisis Fitur Perangkat	41
3.3.2 Analisis Kebutuhan Perangkat	44

4	PERANCANGAN	49
4.1	Rancangan Alur Kerja Perkakas	49
4.1.1	Mencari lokasi menggunakan kata kunci pencarian	49
4.1.2	Mencari rute dengan angkot menggunakan <i>latitude</i> dan <i>longitude</i> lokasi	50
4.1.3	Mencari rute dengan angkot menggunakan kata kunci pencarian lokasi	50
4.2	Rancangan Implementasi Perkakas	51
4.2.1	Cara Kerja Perkakas	52
4.2.2	Tipe Data Tambahan	53
4.2.3	Variabel Global	53
4.2.4	<code>print_help()</code>	56
4.2.5	<code>replace_space()</code>	56
4.2.6	<code>build_url_searchplace()</code>	56
4.2.7	<code>build_url_findroute()</code>	57
4.2.8	<code>reset_url()</code>	57
4.2.9	<code>execute_curl()</code>	57
4.2.10	<code>print_curl_error()</code>	57
4.2.11	<code>write_memalloc()</code>	57
4.2.12	<code>write_searchplace()</code>	57
4.2.13	<code>write_findroute()</code>	57
4.2.14	<code>write_searchplace_noreturns()</code>	58
4.2.15	Fungsi utama (<code>main()</code>)	58
5	IMPLEMENTASI DAN PENGUJIAN	67
5.1	Implementasi Kode	67
5.1.1	<code>print_help()</code>	67
5.1.2	<code>replace_space()</code>	67
5.1.3	<code>build_url_searchplace()</code>	67
5.1.4	<code>build_url_findroute()</code>	67
5.1.5	<code>reset_url()</code>	67
5.1.6	<code>execute_curl()</code>	68
5.1.7	<code>print_curl_error()</code>	68
5.1.8	<code>write_memalloc()</code>	68
5.1.9	<code>write_searchplace()</code>	68
5.1.10	<code>write_findroute()</code>	69
5.1.11	<code>write_searchplace_noreturns()</code>	69
5.1.12	Fungsi utama (<code>main</code>)	69
5.1.13	CMakeLists	69
5.1.14	Halaman manual (<i>man page</i>)	69
5.2	Pengujian	70
5.2.1	Lingkungan Perangkat Keras	70
5.2.2	Lingkungan Perangkat Lunak	70
5.2.3	Pembangunan dan Instalasi	70
5.2.4	Pengujian	72
6	KESIMPULAN	87
6.1	Kesimpulan	87
6.2	Saran	87
	DAFTAR REFERENSI	89
	A KODE SUMBER PERKAKAS <i>Command Line</i> KIRI	91

DAFTAR GAMBAR

1.1	Unit angkot	1
1.2	Tampilan halaman web KIRI	2
2.1	Tampilan halaman web KIRI sebelum masukan diketikkan	5
2.2	Tampilan halaman web KIRI sesudah masukan diproses	6
2.3	Halaman <i>web API Keys</i> KIRI	7
2.4	Penggunaan API KIRI untuk layanan pencarian lokasi	8
2.5	Penggunaan API KIRI untuk layanan pencarian rute	10
2.6	Penggunaan API KIRI untuk layanan <i>smart direction</i>	11
2.7	Contoh CLI	12
2.8	Dua jenis antarmuka perangkat lunak	12
2.9	Baris <i>shell prompt</i> terminal di sistem operasi Linux.	13
2.10	Contoh keluaran <i>man page</i>	15
2.11	Tampang kedua antarmuka <i>command line</i> bawaan di sistem operasi Windows.	16
2.12	Tampilan aplikasi <i>cmake-gui</i>	29
2.13	Tampilan aplikasi <i>ccmake</i>	30
3.1	Contoh penggunaan perkakas Chrome <i>Web Store Item Property CLI</i>	34
3.2	Opsi bantuan penggunaan pada perkakas Chrome <i>Web Store Item Property CLI</i>	35
3.3	Opsi bantuan penggunaan pada perkakas <i>iTunes Search API</i>	36
3.4	Contoh penggunaan perkakas <i>iTunes Search API</i>	37
3.5	Contoh penggunaan perkakas Uber CLI (<i>time</i>)	37
3.6	Contoh penggunaan perkakas Uber CLI (<i>price</i>)	38
3.7	Contoh penggunaan perkakas Google <i>Maps Direction CLI</i>	39
3.8	Diagram <i>use case</i> perkakas yang dibangun	45
4.1	<i>Activity diagram</i> fitur pencarian lokasi menggunakan kata kunci lokasi	50
4.2	<i>Sequence diagram</i> fitur pencarian lokasi menggunakan kata kunci lokasi	51
4.3	<i>Activity diagram</i> fitur pencarian rute angkot menggunakan koordinat lokasi	52
4.4	<i>Sequence diagram</i> fitur pencarian rute angkot menggunakan koordinat lokasi	53
4.5	<i>Activity diagram</i> fitur pencarian rute angkot menggunakan kata kunci lokasi	54
4.6	<i>Sequence diagram</i> fitur pencarian rute angkot menggunakan kata kunci lokasi	55

DAFTAR KODE PROGRAM

2.1	Contoh sederhana penggunaan <code>getopt</code>	19
2.2	Contoh sederhana penggunaan <code>getopt_long</code>	20
2.3	Loop sederhana dari penggunaan <i>multi handle curl</i>	22
2.4	Kumpulan implementasi penggunaan <i>multi socket handle curl</i>	23
2.5	Struktur data <code>cJSON</code>	25
2.6	Kode utama operasional <code>CMake</code>	28
2.7	Contoh kode pembangunan <code>CMake</code> lebih dari satu mode	31
A.1	<code>print_help()</code>	91
A.2	<code>replace_space()</code>	91
A.3	<code>build_url_searchplace()</code>	92
A.4	<code>build_url_findroute()</code>	93
A.5	<code>reset_url()</code>	93
A.6	<code>execute_curl()</code>	94
A.7	<code>print_curl_error()</code>	94
A.8	<code>write_malloc()</code>	94
A.9	<code>write_searchplace()</code>	94
A.10	<code>write_findroute()</code>	95
A.11	<code>write_searchplace_noreturns()</code>	96
A.12	Fungsi utama perkakas (<code>main</code>)	97
A.13	<code>CMakeLists.txt</code>	100
A.14	<code>kiritool.1</code> (<i>Source Code man page</i>)	101
A.15	Keseluruhan kode perkakas (<code>main.c</code>)	102
B.1	Bantuan penggunaan perkakas	113
B.2	<code>man page</code> Perkakas <i>Command Line KIRI</i>	113

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Zaman sekarang ada banyak masalah-masalah terkait transportasi yang sudah mulai mempengaruhi hampir semua orang, seperti pemanasan global akibat gas emisi kendaraan, kemacetan di mana-mana, dan bagi pemilik kendaraan bermotor, semakin meningkatnya harga BBM (bahan bakar minyak). Upaya dari pemerintah untuk menyelesaikan atau meringankan masalah-masalah ini adalah dengan mendorong masyarakat untuk menggunakan sarana transportasi publik. Di Indonesia ada beberapa jenis transportasi umum, sama seperti di negara-negara lain, seperti kereta api, bus, taksi, tetapi satu jenis transportasi umum belum tentu ada di negara-negara lain adalah angkutan kota, atau sering kali disingkat menjadi “angkot”.

Angkot (dapat dilihat di Gambar 1.1) merupakan sebuah unit transportasi umum yang metode operasinya menyerupai bus, hanya saja penumpang angkot dapat meminta pengemudinya untuk turun di mana saja, selama lokasinya masih berada di dalam rute yang sudah ditentukan untuk unit angkot tersebut. Hal ini membuat rute angkot esensial untuk diketahui, karena pengetahuan mengenai lokasi mana saja yang akan dilewati unit-unit angkot yang dinaiki dapat menghemat waktu perjalanan—apabila lokasi yang ingin dituju dilewati oleh unit angkot tersebut, tetapi bukan merupakan pemberhentian akhirnya, penumpangnya dapat langsung meminta pengemudi angkot untuk berhenti di lokasi tersebut. Sistem ini menimbulkan dua masalah lain. Pertama, untuk pergi dari suatu lokasi ke lokasi lain, pengguna juga harus mengetahui unit-unit angkot mana saja yang harus dinaiki. Kedua, untuk mengefektifkan waktu perjalanan dengan angkot, penggunanya juga harus mengetahui persis lokasi-lokasi mana saja yang dilewati oleh unit angkot yang dinaiki.

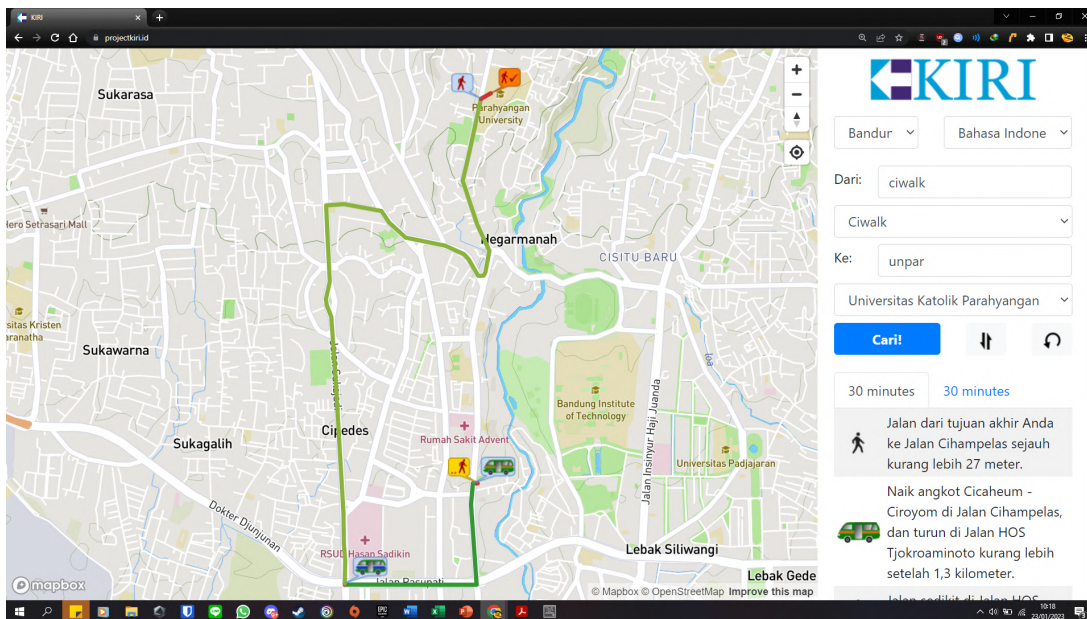


Gambar 1.1: Unit mobil angkutan kota (angkot).

Masalah inilah yang merupakan salah satu tujuan dari Project KIRI. Project KIRI¹ (akan disingkat sebagai KIRI dalam dokumen ini) adalah sebuah perangkat lunak berbasis *web* yang dibuat untuk membantu penggunanya, baik masyarakat maupun turis, dalam menggunakan angkot. Cara

¹<https://projectkiri.id>

KIRI mempermudah penggunaan angkot adalah dengan menunjukkan rute yang akan ditempuh, beserta langkah-langkah yang harus dilakukan oleh pengguna yang ingin berpergian dari satu titik ke titik lain, mulai dari seberapa jauh pengguna harus berjalan untuk menaiki angkot yang bersangkutan, di mana pengguna harus naik atau turun, seberapa jauh lagi pengguna harus berjalan sampai ke titik tujuan, dan seberapa lama estimasi waktu perjalanan yang akan ditempuh—semua di halaman *web* dari KIRI, yang dapat dilihat di Gambar 1.2. Walaupun begitu, dalam kasus-kasus tertentu KIRI memiliki berbagai keterbatasan, misalnya pencarian lokasi tidak akurat, atau rute angkot tidak berhasil ditemukan (walaupun bisa jadi ada rute angkot yang cukup dekat dengan lokasi awal dan tujuan).



Gambar 1.2: Tampilan halaman web KIRI, yang menunjukkan rute dari Cihampelas Walk ke Universitas Katolik Parahyangan.

Pada saat penulisan skripsi ini, KIRI hanya dapat diakses langsung melalui halaman *web*-nya, di mana aplikasi ini memiliki banyak bantuan elemen-elemen grafis, yang menjadikannya sebuah aplikasi berbasis antarmuka pengguna grafis (*graphical user interface/GUI*). GUI ini hanya merupakan satu dari dua jenis antarmuka perangkat lunak, dengan jenis antarmuka satu lagi berupa antarmuka baris perintah (*command line interface/CLI*). *Command line interface* adalah sebuah antarmuka yang berupa sebuah *window* yang memuat teks berupa perintah-perintah,² yang menerima masukan dari pengguna dan menjalankannya [1]. Perintah-perintah ini hanya berupa gabungan dari teks and simbol-simbol berupa karakter. Singkatnya, tipe perangkat lunak ini bukan merupakan tipe yang paling indah untuk dilihat oleh para pengguna, tetapi jika digunakan dengan tepat, maka jenis perangkat lunak ini bisa menyuruh komputer untuk melakukan banyak sekali perintah-perintah dengan sangat cepat dan sangat efektif [2].

Walaupun tipe antarmuka CLI muncul lebih awal dari GUI, sampai sekarang juga masih banyak perangkat-perangkat lunak yang memiliki versi CLI, atau bahkan hanya berbentuk CLI. Ada beberapa alasan CLI masih dipakai di perangkat-perangkat lunak modern, seperti [3]:

- penggunaannya lebih cepat dan sederhana,
- selalu kompatibel di berbagai sistem operasi, bahkan di instalasi-instalasi yang paling mendasar, dan
- kesederhanaannya membuat CLI lebih ideal digunakan untuk tugas-tugas di mana kemudahan konfigurasi lebih penting dari efisiensi dibandingkan GUI.

²<https://ubuntu.com/tutorials/command-line-for-beginners/#3-opening-a-terminal>

Selain aplikasi *web*-nya sendiri, KIRI juga memiliki antarmuka pemrograman aplikasi (*application programming interface/API*) yang dapat digunakan untuk tujuan pengembangan perangkat lunak³. API ini merupakan sebuah antarmuka logikal ke perangkat lunak serta menyembunyikan detail-detail internal implementasinya. Dalam kata lain, API menyediakan sebuah abstraksi untuk sebuah masalah serta mendiktekan bagaimana klien/penggunanya harus berinteraksi dengan komponen perangkat lunak yang menyelesaikan masalah tersebut. Secara esensi, API akan mendefinisikan bagian-bagian yang dapat digunakan ulang, yang memungkinkan potongan-potongan fungsi modular yang dapat langsung diimplementasikan ke perangkat-perangkat lunak lainnya [4]. Khusus untuk kasus ini, API KIRI dapat digunakan dengan mengirimkan permintaan GET, dan nantinya akan mengeluarkan keluaran berupa objek JSON⁴.

Pada skripsi ini akan dibuat sebuah perangkat lunak berupa perkakas *command line* (*command line tool*) yang dapat menjalankan fungsi-fungsi API dari KIRI. Perangkat lunak ini, seperti jenisnya, akan dibuat murni sebagai perkakas yang dijalankan dari *command line* (Terminal, cmd, PowerShell, dll.). Keseluruhan dari perangkat lunak ini akan dibangun dalam bahasa C—bahasa C dipakai karena penggunaan bahasa ini memungkinkan pengaturan manual besar memori sistem yang dipakai oleh perkakas [5] (karena perkakas termasuk perangkat lunak yang ringan, maka memori maksimum yang boleh digunakan dapat diatur sekecil mungkin.) Selain itu, perkakas ini juga akan menggunakan bantuan fungsi bahasa C, yaitu *getopt* untuk penerimaan opsi-opsi masukan, serta *library-library* tambahan, seperti *libcurl* untuk proses transfer data, *cJSON* untuk *mem-parse* keluaran API, dan *CMake* untuk kompatibilitas antar sistem operasi. Bagaimana dan kapan persisnya modul-modul ini akan digunakan di dalam perkakas yang akan dibuat akan dibahas di bagian perancangan. Terakhir, perkakas akan memiliki fitur-fitur perkakas *command line* pada umumnya, seperti halaman/mode bantuan, serta akan memiliki kemampuan integrasi dengan perkakas-perkakas *command line* lainnya, dalam arti bahwa keluaran dari perkakas ini akan bisa digunakan sebagai masukan untuk perkakas-perkakas *command line* lainnya yang sudah ada, seperti *pipeline* (*>*) atau *grep*.

1.2 Rumusan Masalah

Rumusan masalah yang akan dibahas dalam skripsi ini adalah sebagai berikut:

1. Bagaimana membangun perkakas *command line* yang dapat mengimplementasikan fitur-fitur API KIRI dalam bahasa C?
2. Bagaimana integrasi perkakas *command line* KIRI dapat dilakukan dengan perkakas-perkakas *command line* lainnya?

1.3 Tujuan

Tujuan dari skripsi ini adalah sebagai berikut:

1. Membangun perkakas *command line* yang dapat mengimplementasikan fitur-fitur API KIRI dalam bahasa C.
2. Melakukan integrasi perkakas *command line* KIRI dengan perkakas-perkakas *command line* lainnya.

1.4 Batasan Masalah

Perkakas yang dibuat hanya akan memperbaiki kesalahan terjemahan (durasi rute yang ditemukan selalu dalam bahasa inggris) yang sudah sejak awal terdapat dalam API KIRI. Kesalahan-kesalahan lain (lokasi tidak terdeteksi, rute tidak berhasil ditemukan, dsb.) akan ditulis di keluaran perkakas apa adanya.

³<https://projectkiri.id>

⁴<https://github.com/projectkiri/Tirtayasa/wiki/KIRI-API-v2>

1.5 Metodologi

Metodologi yang akan diikuti dalam skripsi ini adalah sebagai berikut:

1. Melakukan studi dan eksplorasi terhadap fungsi-fungsi yang dimiliki perangkat lunak KIRI serta cara implementasi API KIRI.
2. Melakukan analisis dan desain perangkat lunak yang akan dibangun.
3. Melakukan studi dan eksplorasi terhadap seluruh kemungkinan *library-library* yang memenuhi spesifikasi dalam pembuatan perangkat lunak, berdasarkan analisis dan desain yang telah dilakukan sebelumnya.
4. Melakukan analisis kebutuhan fitur-fitur perangkat lunak dan melakukan eksplorasi *library* yang dapat digunakan dan memenuhi spesifikasi dalam pembuatan perangkat lunak.
5. Membangun perangkat lunak berdasarkan rancangan yang sudah dibuat, dengan mengimplementasikan seluruh modul dan *library* yang telah ditentukan di tahap sebelumnya dalam bahasa C.
6. Melakukan pengujian fungsional, perbaikan *bug*, serta rekomendasi perbaikan berdasarkan pengujian yang sudah dilakukan, jika diperlukan.
7. Menyelesaikan pembuatan dokumen-dokumen yang berkaitan, seperti dokumen skripsi dan dokumentasi perangkat lunak.

1.6 Sistematika Pembahasan

Setiap bab dalam skripsi ini mengikuti sistematika yang terdiri atas poin-poin sebagai berikut:

1. **Bab 1:** Pendahuluan
Bab ini berisi latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, dan sistematika pembahasan.
2. **Bab 2:** Dasar Teori
Bab ini berisi pembahasan-pembahasan teoretis mengenai aspek-aspek yang akan dirujuk di dalam skripsi ini, seperti *command line*, bahasa C, dan juga KIRI.
3. **Bab 3:** Analisis
Bab ini berisi analisis perangkat-perkakas sejenis, analisis API KIRI, serta analisis fitur-fitur perangkat yang akan dibuat.
4. **Bab 4:** Perancangan
Bab ini berisi pembahasan mengenai rancangan cara kerja tiap-tiap fitur serta fungsi-fungsi dalam kode perangkat yang akan dibuat.
5. **Bab 5:** Implementasi dan Pengujian
Bab ini berisi dua bagian utama, yaitu:
 - Implementasi
Bagian ini meliputi struktur kelas dan penjelasan tiap-tiap fungsi di dalamnya.
 - Pengujian
Bagian ini meliputi cara instalasi, cara menggunakan perangkat, serta pengujian fungsional terhadap fitur-fitur dari perangkat yang telah dibuat.
6. **Bab 6:** Kesimpulan dan Saran
Bab ini berisi kesimpulan hasil pembuatan perangkat lunak dan saran-saran terhadap hasil perangkat lunak yang diberikan selama pengerjaan skripsi.