SKRIPSI

PEMBUATAN TWITTER BOT UNTUK MENCARI JALUR TRANSPORTASI PUBLIK



KEVIN THEODORUS YONATHAN

NPM: 2011730037

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2014

UNDERGRADUATE THESIS

FOLLOWING THE MAJORITY: A NEW ALGORITHM FOR COMPUTING A MEDIAN TRAJECTORY



KEVIN THEODORUS YONATHAN

NPM: 2011730037

DEPARTMENT OF INFORMATICS FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES PARAHYANGAN CATHOLIC UNIVERSITY 2014

LEMBAR PENGESAHAN

PEMBUATAN TWITTER BOT UNTUK MENCARI JALUR TRANSPORTASI PUBLIK

KEVIN THEODORUS YONATHAN

NPM: 2011730037

Bandung, 4 Juli 2014 Menyetujui,

Pembimbing Tunggal

Pascal Alfadian, M.Com.

Ketua Tim Penguji

Anggota Tim Penguji

Thomas Anung Basuki, Ph.D. Dr. rer. nat. Cecilia Esti Nugraheni

Mengetahui,

Ketua Program Studi

Thomas Anung Basuki, Ph.D.

PERNYATAAN

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

PEMBUATAN $TWITTER\ BOT$ UNTUK MENCARI JALUR TRANSPORTASI PUBLIK

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 4 Juli 2014

Meterai

Kevin Theodorus Yonathan NPM: 2011730037

ABSTRAK

. . .

Kata-kata kunci: Lintasan, Homotopy, Fréchet, Lintasan Median, Penyangga

ABSTRACT

. . .

Keywords: Trajectory, Homotopy, Fréchet, Median Trajectory, Buffer



KATA PENGANTAR

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Bandung, Juli 2014

Penulis

DAFTAR ISI

K	ATA	PENGA	ANT.	\mathbf{AR}																	$\mathbf{x}\mathbf{v}$
\mathbf{D}_{I}	Daftar Isi xvii																				
\mathbf{D}_{A}	AFTA	R GAN	MBA	\mathbf{R}																X	viii
\mathbf{D}_{I}	AFTA	R TAB	BEL																	-	xix
1	PEN 1.1	NDAHU Latar			g .		•														1 1
	1.2 1.3	Rumus Batasa	ısan	Mas	$\overset{ ext{o}}{ ext{alah}}$				 												2 2
	1.4 1.5	Tujua: Metod	n.						 												4
2	DAS	SAR TI																			5
	2.1 2.2	Twitte		PΙ																	5 6
		$2.2.1 \\ 2.2.2$		arch Auth																	6 9
		2.2.3	Str	ream ıblic	ming	g AF	PΙ		 												13 13
		2.2.5	Us	er St	rean	ns .			 											 •	15
		2.2.6 $2.2.7$		RI A																	16 18
\mathbf{D}_{A}	AFTA	R REF	FERI	ENSI																	21

DAFTAR GAMBAR

2.1	flow application-only authentication	11
2.2	Ilustrasi sign in	12

DAFTAR TABEL

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan zaman, perkembangan internet di Indonesia sudah semakin maju. Banyak orang sudah menggunakan fasilitas internet untuk berbagai macam kebutuhan. Contoh dari penggunaan internet adalah untuk mencari informasi, email, bermain jejaring sosial online, Internet Banking, online shop, dan lain lain. Menurut Kemkoinfo (8 mei 2014) pengguna internet di Indonesia capai 82 Juta orang, delapan puluh persen diantaranya adalah remaja. Hal ini menunjukkan bahwa internet sudah tidak asing lagi untuk masyarakat di Indonesia ini. Sebagai informasi tambahan bahwa pengguna internet di Indonesia 95 persennya digunakan untuk social media atau jejaring social online.

Twitter adalah salah satu layanan jejaring sosial online yang memungkinkan pengguna memposting pesan berbasis teks hingga 140 karakter. Pengguna Twitter menyebutnya sebagai tweet. Tweet ini akan meneruskan pesan singkat yang ditujukan ke semua follower suatu akun¹. Follow adalah salah satu istilah dalam Twitter yang bertujuan untuk mengikuti aktivitas tweet suatu akun. Sedangkan cara seseorang untuk dapat memberi rujukan kepada akun Twitter yang lainnya adalah dengan cara reply atau lebih dikenal dengan nama mention². Sebagai contoh, diketahui akun bernama @kviniink mem-follow @infobdg untuk mengetahui perkembangan apa saja yang tejadi di kota Bandung. Lalu akun @kviniink ingin bertanya tentang info mall yang ramai di Bandung, maka akun @kviniink membuat mention tweet yang berisikan "@infobdg Halo saya ingin bertanya apa saja mall yang sedang ramai di Bandung yah?".

Transportasi publik sudah banyak digunakan oleh kebanyakan orang di dunia, bukan hanya di Indonesia saja transportasi publik ini sudah banyak digunakan di luar negeri. Menurut data, angkutan umum di Kota Bandung pada tahun 2013 sudah lebih dari 12000 unit kendaraan. Keuntungan memakai transportasi publik sudah banyak dirasakan di seluruh dunia yaitu untuk mengatasi kemacetan dan mengurangi pemanasan global. Seiring dengan perkembangan teknologi, menaiki transportasi publik menjadi semakin mudah. Seiring dengan perkembangan teknologi, menaiki transportasi publik menjadi semakin mudah. Dengan adanya KIRI di Indonesia terutama di daerah Bandung, masyarakat dapat naik transportasi publik tanpa harus mengetahui terlebih dahulu kendaraan yang harus dinaikinya. Dengan adanya KIRI, pengguna hanya perlu tahu tempat asal dan tempat tujuan untuk dapat menaiki transportasi publik di Bandung ini.

KIRI API adalah aplikasi pihak ketiga yang memungkinkan programmer mendapatkan data ten-

¹Dusty Reagan, Twitter Application Development For Dummies, Wiley, 2010, page 7

²Dusty Reagan, Twitter Application Development For Dummies, Wiley, 2010, page 9

2 Bab 1. Pendahuluan

tang info jalur transportasi publik. Twitter API adalah aplikasi pihak ketiga yang memungkinkan programmer melakukan manipulasi dan pengolahan data di Twitter. Dengan memanfaatkan KIRI API dan Twitter API peneliti akan membuat program yang dapat membalas tweet untuk mencari jalur transportasi publik. Program yang dibuat akan bersifat real time sehingga jika seseorang melakukan mention kepada bot pencari jalur maka bot akan menangkapnya dan membalas mention tersebut berupa jalur yang harus ditempuh. Contoh dari jalannya program adalah ketika akun bernama @kviniink melakukan mention kepada @kiriupdate untuk bertanya jalur transportasi publik "@kiriupdate #find bip to ip". Maka Twitter bot @kiriupdate akan mendengarkan mention dari akun @kviniink lalu mention tersebut akan diolah oleh server dan akan di-reply dengan tiga buah tweet "@kviniink istana plaza to bandung indah plaza", "@kviniink Walk about 135 meter from your starting point to Jalan Aceh.", "@kviniink Take angkot Ciroyom - Antapani at Jalan Aceh, and alight at Jalan Pajajaran about 3.6 kilometer later.", "@kviniink Walk about 93 meter from Jalan Pajajaran to your destination.". Karena keterbatasan 140 karakter maka tweet akan dipecah sesuai dengan instruksi yang dikirimkan dari KIRI API.

Oleh karena itu, dalam penelitian ini akan dibangun sebuah perangkat lunak yang dapat memudahkan pengguna dalam mencari jalur transportasi publik. Sebuah applikasi yang menggabungkan jejaring sosial online Twitter dengan KIRI API. Jadi pengguna bisa melakukan tweet kepada Twitter bot yang dibuat dengan format tertentu yang berisikan tempat asal dan tempat yang akan dituju. Lalu pengguna akan menerima balasan tweet berupa rute jalan yang harus ditempuhnya.

1.2 Rumusan Masalah

Mengacu kepada deskripsi yang diberikan, maka rumusan masalah pada penelitian ini adalah:

- Bagaimana membuat Twitter bot untuk mencari jalur transportasi publik?
- Bagaimana membuat Twitter bot untuk dapat merespon secara real time?
- Bagaimana memformat petunjuk rute perjalanan dalam keterbatasan tweet 140 karakter?

1.3 Batasan Masalah

Pada pembuatan perangkat lunak ini, masalah-masalah yang ada akan dibatasi menjadi:

- Input hanya mencakup Kota Bandung saja.
- Input yang diinputkan harus benar, memiliki asal dan tujuan yang jelas di Kota Bandung.
- Hasil yang dikeluarkan berupa tweet jalur transportasi publik.
- Media transportasi publik yang digunakan adalah angkutan umum.
- Pencarian jalur memanfaatkan KIRI API.

1.3. Batasan Masalah 3

Project Kiri (KIRI)

@kiriupdate

@kiriupdate #find unpar ciumbuleuit to bandung indah plaza

9:08 - 5 Mei 14

@pascalalfadian unpar ciumbuleuit to bandung indah plaza

9:20 - 5 Mei 14

@pascalalfadian 1. Walk about 507 meter from your starting point to Jalan Gunung Kareumbi.

9:20 - 5 Mei 14

@pascalalfadian 2. Take angkot Ciumbuleuit - St. Hall (lurus) at Jalan Gunung





4 Bab 1. Pendahuluan

1.4 Tujuan

Tujuan dari penelitian ini adalah:

- Membuat aplikasi Twitter bot untuk mencari jalur transportasi publik.
- Membuat aplikasi Twitter yang bekerja secara real time.
- Membuat algoritma untuk memecah instruksi dari KIRI API dan mengubahnya ke dalam bentuk tweet.

1.5 Metode Penelitian

Pada perangkat lunak yang dibuat ini digunakan beberapa metode dalam penyelesaian masalah yang menjadi topik pada penelitian ini, antara lain:

- 1. Melakukan studi literatur, antara lain:
 - KIRI API,
 - REST API Twitter (https://dev.twitter.com/docs/api/1.1),
 - Streaming API Twitter (https://dev.twitter.com/docs/api/streaming).
- 2. Mempelajari pembuatan server dalam bahasa Java.
- 3. Membuat TwitterBot sederhana
- 4. Melakukan analisis terhadap teori-teori yang sudah dipelajari, guna membangun perangkat lunak yang dimaksud.
- 5. Melakukan pengujian terhadap system yang sudah dibangun

٠

BAB 2

DASAR TEORI

Sebelum bisa membuat Twitter bot untuk mencari jalur transportasi publik, berikut diberikan beberapa definisi yang berkaitan dengan pembuatan Twitter bot. Bab ini akan menjelaskan Twitter, Twitter API, KIRI, KIRI API, dan Twitter4j.

2.1 Twitter

Twitter adalah layanan yang memungkinkan pengguna untuk mengirim pesan menggunakan 140 karakter atau kurang. Pesan tersebut dapat diadaptasikan melalui teks, aplikasi *mobile*, atau web. (referensi dari buku Sams teach yourself the twitter api) Berikut ini adalah daftar istilah umum pada Twitter:

Twitter adalah salah satu layanan jejaring sosial online yang memungkinkan pengguna memposting pesan berbasis teks hingga 140 karakter (referensi dari buku The Twitter Book).

• Tweet

Posting pada Twitter disebut sebagai tweet. Tweet ini akan meneruskan pesan singkat yang ditujukan ke semua follower suatu akun¹. Contohnya adalah seorang akun @kviniink ingin menuliskan bahwa hari ini cuaca cerah, maka @kviniink akan men-tweet 'Hari ini cerah yah..' Tweet juga bisa menyertakan link ke video, foto, atau media lain di internet selain teks biasa. URL link teks termasuk ke dalam 140 batas karakter, namun URL tersebut akan menghabisnya tempat/space dari keterbatasan karakter tweet. Oleh karena itu URL akan dibuat versi singkatnya, contoh saat pengguna memasukkan link 'http://www.chacha.com/gallery/7253/15-movies-that-make-guys-cry', maka akan dibuat menjadi 'bit.ly/1uRi8vV'.

• Follow

Follow adalah satu istilah dalam Twitter yang bertujuan untuk mengikuti aktivitas tweet suatu akun. Following adalah ketika sebuah akun mengikuti akun orang lain, dan Follower adalah ketika sebuah akun melakukan aksi follow kepada akun anda.

• Reply

Reply adalah cara seseorang untuk dapat memberi rujukan kepada akun Twitter yang lainnya atau lebih dikenal dengan nama $mention^2$. Sebagai contoh, diketahui akun bernama @kviniink

¹Dusty Reagan, Twitter Application Development For Dummies, Wiley, 2010, page 7

²Dusty Reagan, Twitter Application Development For Dummies, Wiley, 2010, page 9

mem-follow @infobdg untuk mengetahui perkembangan apa saja yang tejadi di kota Bandung. Lalu akun @kviniink ingin bertanya tentang info mall yang ramai di Bandung, maka akun @kviniink membuat mention tweet yang berisikan "@infobdg Halo saya ingin bertanya apa saja mall yang sedang ramai di Bandung yah?".

• Retweet

Retweet ini merupakan salah satu yang paling penting dari Twitter(referensi the twitter book halaman 47). Retweet ini berguna ketika pengguna menemukan tweet menarik dan berbagi tweet tersebut dengan follower akun tersebut (follower). Retweet ini juga secara tidak langsung mengatakan bahwa "saya menghormati anda dan pesan yang anda buat".

• Hashtag

Sebuah fitur yang diciptakan oleh Twitter untuk membantu pencarian kata kunci dan penandaan suatu diskusi.

• Direct Message(DM)

Direct message digunakan untuk mengirim pesan yang bersifat private antara dua orang. Orang yang mengirim direct message ini hanya bisa untuk orang yang mengikuti akun tersebut.

• Timeline

Timeline adalah sekumpulan tweet-tweet dari semua orang yang anda follow lalu akan ditampilkan di halaman utama.

2.2 Twitter API

Twitter API adalah aplikasi pihak ketiga yang memungkinkan programmer melakukan manipulasi dan pengolahan data di Twitter. Twitter API tidak seperti API pada umumnya karena Twitter memaparkan hampir semuanya termasuk setup account dan informasi kustumisasi[1]. Ini adalah salah satu bentuk pendekatan dari Twitter yang berfokus pada jaringan dan memungkinkan developer memiliki hak untuk berpikir 'out of the box' untuk membuat aplikasi yang mereka inginkan. Tetapi tetap akan terjadi keterbatasan yang dimiliki Twitter API, yaitu:

- Hanya bisa men-update 1000 per harinya, baik melalui handphone, website, API, dan sebagainya.
- Total pesan hanya bisa sebanyak 250 per harinya, pada setiap dan semua perangkat.
- 150 permintaan API per jam.
- OAuth diijinkan 350 permintaan per jam.

2.2.1 Search API

Twitter Search API memungkinkan query memiliki perilaku yang sama terhadap indeks dari tweet baru ataupun tweet populer. Twitter Search API ini bukan fitur yang tersedia pada Twitter itu 2.2. Twitter API

sendiri. API ini difokuskan kepada relefansi, bukan terhadap kelengkapan data. Ini berarti bahwa beberapa Tweet dan pengguna akan hilang dari hasil pencarian.

Bagaimana cara membuat sebuah query Cara berbaik dalam membuat sebuah query dan melakukan percobaan yang valid dan mengembalikan tweet yang sesuai dapat dilakukan pada twitter.com/search. URL yang ditampilkan pada browser akan berisi sintaks query yang sesuai agar dapat digunakan kempali pada API endpoint. Berikut adalah contohnya:

- 1. Melakukan pencarian untuk tweet yang direferensikan kepada akun @twitterapi. Pertama kita harus melakukan pencarian pada twitter.com/search.
- 2. Lakukan pengecekan dan salin URL yang ditampilkan. Sebagai contoh didapatkan URL seperti ini, https://twitter.com/search?q=%40twitterapi
- 3. Ganti "https://twitter.com/search" dengan "https://api.twitter.com/1.1/search/tweets.json" dan akan didapatkan "https://api.twitter.com/1.1/search/tweets.json?q=%40twitterapi"
- 4. Eksekusi URL tersebut untuk melakukan pencarian di dalam API.

API v1.1 mewajibkan bahwa request harus diotentifikasi. Perlu diingat juga bahwa hasil pencarian yang dilakukan di twitter.com dapat menghasilkan hasil yang sudah sangat lama, sedangkan Search API hanya melayani tweet dari seminggu terakhir.

Finds tweets Operator containing both "watching" and "now". This is the default operator. watching now "happy hour" containing the exact phrase "happy hour". love OR hate containing either "love" or "hate" (or both). beer -root containing "beer" but not "root". ontaining the hashtag "haiku". #haiku sent from person "alexiskold". from:alexiskold to:techcrunch sent to person "techcrunch". @mashable referencing person "mashable". superhero since:2010-12-27 containing "superhero" and sent since date "2010-12-27" (year-month-day). containing "ftw" and sent before the date "2010-12-27". ftw until:2010-12-27 containing "movie", but not "scary", and with a positive attitude. movie -scary:) containing "flight" and with a negative attitude. flight :(traffic? containing "traffic" and asking a question. hilarious filter:links containing "hilarious" and linking to URL. news source:twitterfeed containing "news" and entered via TwitterFeed

Dipastikan bahwa melakukan pengkodean URL terhadap query terlebih dahulu sebelum melakukan request. Tabel berikut memberikan contoh mapping dari search query ke query pengkodean URL.

Search query URL encoded query #haiku #poetry %23haiku+%23poetry

"happy hour":) %22happy%20hour%22%20%3A%29

Additional parameters Terdapat parameter tambahan yang dipergunakan untuk hasil pencarian yang lebih baik. Berikut adalah penjelasan dari parameter tambahan tersebut :

- Result Type. Seperti hasil yang terdapat pada twitter.com/search, parameter result_type memungkinkan hasil pencarian akan berdasarkan tweet yang paling baru atau tweet yang paling poluler atau bahkan gabungan dari keduanya.
- Geolocatization. Pencarian tempat tidak tersedia pada API, tetapi ada beberapa cara yang tepat untuk membatasi query dengan cara menggunakan parameter geocode lalu mementukan "latitude, longitude, radius". Contohnya adalah "37.781157,-122.398720,1mi". Ketika pencarian lokasim pencarian API pertama akan mencoba menemukan tweet yang memiliki latitude yang sudah dimasukan kedalam query geocode, jika tidak berhasil maka API akan mencoba menemukan tweet yang dibuat oleh pengguna yang lokasi profilenya terdapat pada latitude tersebut. Artinya adalah hasil pencarian mungkin menerima tweet yang tidak mencakup informasi latidute atau longitude.
- Language. Bahasa dapat dijadikan parameter untuk mencari tweet yang sesuai dengan bahasa tersebut.
- *Iterating in a result set*. Parameter seperti count, until, since_id, max_id memungkinkan untuk mengkontrol bagaimana iterasi melalui hasil pencarian.

Rate limits User pada saat ini diwakilkan oleh access tokens yang dapat membuat 180 request per 15 menit. Tetapi kita bisa membuat 450 request per 15 menit dengan cara menggunakan application-onlu auth atas nama sendiri tanpa konteks pengguna.

Contoh Pencarian Ketika anda mengikuti suatu acara yang sedang berlangsung, anda tertarik untuk mencarinya dengan melihat tweet yang paling baru dan menggunakan hastag dari acara tersebut, maka langkah-langkah yang dilakukan adalah:

- Anda ingin mencari tweet yang paling baru dengan menggunakan hastag #superbowl
- Maka search URL akan seperti ini: https://api.twitter.com/1.1/search/tweets.json?q=%23superbowl&result

Ketika anda intin mengetahui tweet yang datang dari suatu lokasi dengan bahasa yang spesifik, maka langkah-langkah yang dilakukan adalah:

- Anda ingin mencari tweet yang paling baru dalam Bahasa Portugal, yang lokasinya dekat MaracanÃč soccer stadium yang terletak di Rio de Janeiro.
- Maka search URL akan seperti ini: https://api.twitter.com/1.1/search/tweets.json?q=&geocode=-22.912214,-43.230182,1km&lang=pt&result type=recent

Ketika anda ingin mencari tweet yang sedang poluler dari spesifik user dan tweet tersebut terdapat sebuah hashtag tertentu:

- Anda ingin mencari tweet yang poluler yang berasal dari @kviniink yang terdapat hashtag #nasa.
- Maka search URL akan seperti ini: https://api.twitter.com/1.1/search/tweets.json?q=from%3Akviniink

2.2. TWITTER API

2.2.2 OAuth

Dengan semakin berkembangnya website, semakin banyak situs yang bergantung pada layanan distribusi dan *cloud computing*. Contohnya adalah menggunakan jejaring sosial dengan menggunakan account media sosial lainnya seperti Google untuk mencari teman-teman yang sudah tersimpan pada kontak Google. Atau bisa juga menggunakan pihak ketiga yang memanfaatkan API dari beberapa layanan.

OAuth menyediakan suatu metode bagi pengguna untuk memberi akses pihak ketiga untuk resources (sumber daya) mereka tanpa berbagi password mereka. Cara ini juga memberikan cara untuk memberikan akses yang terbatas (dalam satu lingkup atau durasi) Sebagai contoh, seorang pengguna web dapat memberikan layanan percetakan (client) untuk mengakses foto pribadinya yang disimpan di layanan berbagi foto (server) tanpa harus memberikan username dan passwordnya. Ia akan mengotentikasi langsung dengan layanan berbagi foto tersebut yang mengeluarkan layanan percetakan.

Dalam model otentikasi client-server tradisional, klian menggunakan kredensial untuk mengakses recources hosted oleh server. Di dalam model OAuth, klien (bukan pemilik resource, tetapi bertindak atas namanya) meminta aksesn ke resource yang dikenalkan oleh pemilik resource namun diselenggarakan oleh server.

Agar klien dapat mengakses resource, pertama-tama ia harus mendapatkan izin dari si pemilik resource. Izin ini dinyatakan dalam bentuk token dan mencocokan *shared-secret*. tujuan dari token ini adalah untuk membuat pemilik resource untuk berbagi kepercayaan kepada klien. Berbeda dengan kepercayaan pemilik resource. Token dapat dikeluarkan dalam ruang lingkup terbatas, durasi yang terbatas, dan akan dicabut secara independen. Reverensi(http://hueniverse.com/oauth/guide/intro/)

Twitter OAuth yang diberikan memiliki fitur

• secure

Pengguna tidak harus berbagi password mereka dengan aplikasi pihak ketiga untuk meningkatkan keamanan akun.

• Standard

Contoh code yang diberikan sudah kompatibel dengan implementasi Twitter OAuth.

 $API\ v1.1$'s Authentication Model Terdapat 2 bentuk dalam otentifikasi model baru, dan keduanya masih memanfaatkan OAuth $1.0\mathrm{A}$

Application-user authentication Application-user authentication adalah bentuk paling umum dari otentikasi resource dalam pelaksanaan OAuth 1.0A Twitter sampai saat ini. Permintaan anda menandatangani baik untuk mengidentifikasi identitas aplikasi anda yang akan menyertakan izin untuk diberikan kepada pengguna. Hal ini bertujuan untuk dapat membuat panggilan API atas nama anda yang diwakili oleh akses token.

 $Application-only\ authentication\ Application-only\ authentication\ adalah\ bentuk\ dari\ otentifikasi$ dimana aplikasi anda membuat $API\ request$ atas nama aplikasi itu sendiri tanpa adanya konteks dari pengguna. Pemanggilan API masih terbatas dalam setiap $API\ method$.

Application-only authentication

Twitter menawarkan aplikasi yang mampu mengeluarkan permintaan otentifikasi atas nama aplikasi itu sendiri. Dengan menggunakan Application-only authentication anda tidak mempunyai konteks dari otentikafikasi pengguna dan ini berarti setiap request API untuk endpoint akan membutuhkan konteks user, seperti memposting tweet tidak akan bekerja. Aplikasi yang akan di dapat adalah:

- Melihat timeline
- Mengakses following dan follower dari suatu akun
- Mencari dalam tweet
- mengambil informasi dari akun manapun

Tetapi application-only authentication tidak bisa melakukan :

- Posting tweet
- Melakukan koneksi dengan Streaming endpoint
- Mencari akun seseorang
- menggunakan geo endpoint manapun
- mengakses DM

Auth Flow Langkah-langkah dari application-only auth terdiri dari beberapa langkah yaitu: Sebuah aplkasi dikodekan berdasarkan consumer key dan secret ke dalam satu set khusus yang dikodekan secara kredensial. Aplikasi membuat request ke POST OAuth2/token endpoint untuk merubah kredensial tersebut untuk token bearer. Ketika mengakses REST API, aplikasi menggunakan token bearer untuk otentifikasi. Kerena tidak ada kebutuhan duntuk menandatangani request, pendekatan ini lebih sederhana dari model standar OAuth 1.0a

Tentang application-only auth Token adalah password. Perlu diingat bahwa consumer key dan secret, bearer token credential, dan the bearer token itu sendiri memberikan akses untuk membuat permintaan atas nama aplikasi itu sendiri. Point-point ini harus dianggap sensitif layaknya password dan tidak boleh dibagikan atau didistribusikan kepada pihak yang tidak dipercaya atau tidak berkepentingan

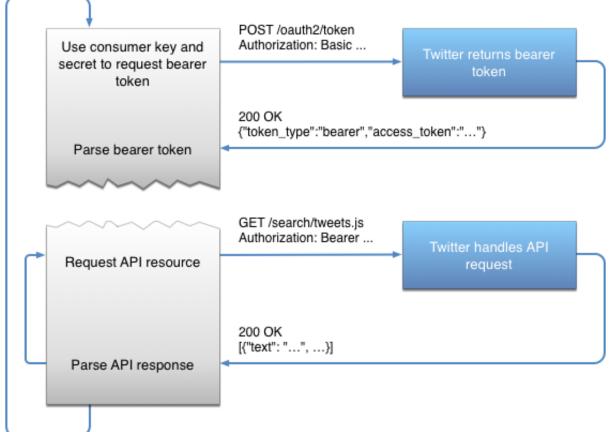
SSL benar-benar dibutuhkan karena ini adalah cara otentifikasi yang aman. Oleh karena itu semua request (baik untuk mendapatkan atau menggunakan token) harus menggunakan endpoint HTTPS, yang juga merupakan syarat untuk menggunakan API v1.1.

Tidak ada konteks pengguna. Ketika mengeluarkan permintaan menggunakan application-only auth, tidak ada konsep 'current-user'. Karena itu endpoint seperti POST status / update tidak akan berfungsi dengan application-only auth.

Rate limiting. Request yang dibuat atas nama pengguna tidak akan menguras ketersediaan rate limit dan request tidak akan menguras batas penggunaan limit dalam user-based auth.

11 2.2. Twitter API

application-only authentication.png



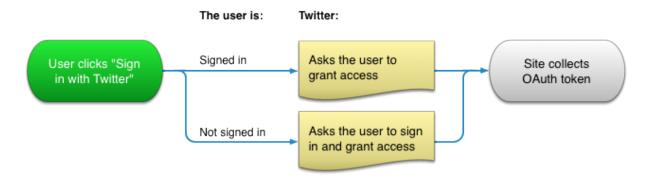
Gambar 2.1: flow application-only authentication

3-legged authorization

Cara kerja dari 3-legged authorization adalah dengan memberikan aplikasi yang anda buat untuk mengambil access token dengan cara melakukan redirect user dengan Twitter dan memberikan mereka sebuah otorisasi dari aplikasi yang anda buat. Cara kerja ini hampir identik dengan cara kerja yang dijelaskan pada implementasi Sign in dengan Twitter, hanya saja terdapat dua pengecualian yaitu:

- GET oauth endpoint digunakan sebagai pengganti GET oauth
- User akan selalu diminta untuk mengotorisasi akses ke aplikasi anda, bahkan jika akses sebelumnya telah diberikan

Beginilah ilustrasi interaksi sign in dengan menggunakan following flowchart



Gambar 2.2: Ilustrasi sign in

PIN-based authorization cara kerja dari PIN-based authorization ini ditujukan untuk aplikasi yang tidak bisa mengakses atau menanamkan web browser untuk mengarahkan user kepada authorization endpoint. Contohnya adalah aplikasi yang bersifat command-line, embedded systems, game konsol, dan beberapa jenis aplikasi mobile.

Implementasi

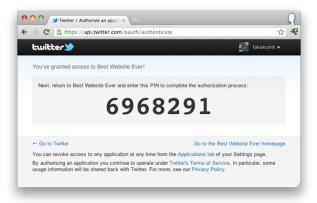
Implementasi PIN-based ini memiliki cara kerja yang sama seperti implementasi Sign in dengan Twitter dan 3-legged authorization, perbedaannya terletak pada nilai dari oauth_callback yang harus di set menjadi oob saat proses pemanggilan POST oauth atau request_token.

Setelah applikasi anda telah mendapatkan GET oauth/authenticate atau GET oauth/authorize URL, tampilkan URL kepada user agar mereka dapat menggunakan web browser untuk mengakses Twitter.

Ketika *callback oob* diminta dan user pengunjungi Twitter, user tidak akan dipindahkan secara otomatis ke aplikasi setelah menyetujui akses. Sebaliknya, mereka akan melihat PIN code, dengan instruksi untuk kembali ke aplikasi dan memasukkan nilai dari PIN code tersebut.

Aplikasi anda harus memungkinkan user untuk memasukkan $PIN\ code$ ini untuk menyelesaikan flow tersebut. Nilai dari $PIN\ code$ harus lolos sebagai oauth_verifier untuk $POST\ oauth/a-ccess\ token\ request$. semua request akan berjalan normal kedepannya.

2.2. Twitter API



2.2.3 Streamming API

Streamming API adalah contoh real-time API. API ini ditujukan bagi para pengembang dengan kebutuhan data yang intensif. Contohnya jika mencari cara untuk membangun sebuah data produk data-mining atau tertarik dalam analisis penelitian. Streaming API memungkinkan melacak kata kunci yang ditentukan dalam jumlah besar dan melakukan suatu aksi (seperti tweet) secara langsung atau real-time.

Twitter menawarkan beberapa endpoint streaming, disesuaikan dengan kasus yang terjadi.

• Public stream

Steaming data publik yang mengalir melalui Twitter. Dipergunakan untuk mengikuti sebuah akun atau topik tertentu. Selain itu juga public stream digunakan untuk data mining.

• User Stream

Single-user streams, mengandung hampir semua data yang berhubungan dengan satu user tertentu.

• Site Stream

Versi dari multi-user stream. Site stream harus terhubung dengan server yang terkoneksi dengan twitter atas nama banyak pengguna.

2.2.4 Public Streams

Stream ini menawarkan sampel data publik yang mengalir melalui Twitter. Ketika aplikasi membuat sambungan ke streaming endpoint, aplikasi akan menyampaikan umpan Tweet tanpa perlu khawatir akan keterbatasan rate limit.

Endpoints

- POST statuses / filter
- GET statuses / sample
- GET statuses / firehose

POST statuses/filter POST filter ini mengembalikan status publik yang sesuai dengan satu atau lebih predikat yang telah di filter. Multiple parameter memungkinkan klien untuk menggunakan koneksi tunggal untuk ke Streaming API. Antara GET dan POST request keduanya didukung tetapi GET request yang memiliki parameter yang terlalu banyak mungkin akan ditolak karena URL yang terlalu panjang. Gunakanlah POST request untuk menghindari URL yang panjang. Track, follow, dan lokasi harus dipertimbangkan untuk dapat digabungkan dengan operator OR. track=foo&follow=1234 ini mengembalikan tweet yang memiliki kata "foo" atau dibuat oleh user 1234. Hal ini memungkinkan akses hingga 400 kata kunci, 5000 follow users, dan 250.1-360 derajat kotak lokasi.

Response formats	JSON
Requires authentication?	Ya (hanya user context)
Rate limited?	Ya

Resource Information

Parameter Perlu diperhatikan bahwa salah satu parameter (follow, location, atau track) harus diisi secara spesifik.

follow Tanda koma memisahkan list user ID, hal ini menunjukkan pengguna untuk kembali ke status

track Kata pencarian untuk track. Fase kata kunci dipisahkan oleh tanda koma.

locations Menentukan lokasi yang dilacak.

delimited Menentukan apakah pesan harus dibatasi limitnya.

stall_warnings Menentukan apakah pesan warning harus dikirim atau tidak.

GET statuses/sample Mengembalikan random sampel dari semua status public. Tweet akan dikembalikan dengan cara seperti biasa, jadi jika terdapat dua client yang terhubung dengan endpoint ini, maka mereka akan melihat Tweet yang sama.

Response formats	JSON
Requires authentication?	Ya (hanya user context)
Rate limited?	Ya

Resource Information

Parameter

GET statuses/firehose Mengembalikan semua status public. Beberapa aplikasi membutuhan akses ini. Teknik ini diolah secara kreatif dengan cara menggabungkan sumber informasi yang ada dengan berbagai sumber lainnya maka dapat memuaskan pengguna.

Resource Information

Parameter

2.2. Twitter API 15

delimited Menentukan apakah pesan harus dibatasi limitnya. stall warning Menentukan apakah pesan warning harus dikirim atau tidak.

Response formats	JSON
Requires authentication?	Ya (hanya user context)
Rate limited?	Ya

Menggunakan Streaming API Hubungkan endpoint yang sudah tercantum di atas dengan parameter yang sudah di list pada koneksi kepada streaming endpoint dan juga request parameter streaming API. Proses pengembalian data oleh streaming API dilakukan dengan cara mengikuti petunjuk dalam pengolahan data streaming.

Koneksi Setiap akun hanya dapat membuat satu koneksi yang terbuhung dengan public endpoint dan jika melakukan koneksi ke public stream lebih dari satu kali dengan menggunakan akun
yang sama akan menyebabkan koneksi terlama akan putus. Klien yang membuat koneksi secara
berlebihan baik berhasil ataupun tidak maka IP mereka otomatis akan di banned.

2.2.5 User Streams

User Stream memberikan aliran(stream) data dan event yang spesific untuk pengguna yang sudah diotentifikasi. User Stream tidak dimaksudkan untuk koneksi server ke server, Jika anda perlu membuat koneksi atas nama beberapa user dari mesin yang sama maka lebih baik menggunakan site stream.

Endpoints

• GET user

Resource Information

Parameter

Koneksi Meminimalkan jumlah koneksi suatu aplikasi untuk membuat user stream. Setiap akun Twitter terbatas hanya untuk beberapa koneksi user streams per aplikasi OAuth, terlepas dari IP. Setelah mencapai batasnya maka koneksi tertua / terlama akan diberhentikan. Akun login dari beberapa instansi dari aplikasi AOuth yang sama akan mengalami sikluk koneksi yaitu akan dihubungan dan diputuskan satu sama lain.

Sebuah aplikasi harus dapat mengatasi HTTP 420 error code yang memberitahukan bahwa suatu akun sudah terlalu sering login. Oleh karena itu user yang seperti itu akan secara otomatis di banned dari User Stream untuk tingkat login yang berlebihan. Untuk memulihkan akses streaming user harus menutup aplikasi tambahan yang ada, mungkin berjalan di perangkat atau device yang berbeda.

Perhatikan bahwa setiap aplikasi memiliki alokasinya masing-masing, sehingga login dari applikasi pertama tidak akan mempengaruhi applikasi ke dua begitu juga sebaliknya. Tetapi menjalankan terlalu banyak salinan aplikasi pertama maupun ke dua akan menimbulkan masalah. Perhatikan

count Kumpulan pesan untuk dijadikan bahan materi delimited Menentukan apakah pesan harus dibatasi limitnya.

stall_warning Menentukan apakah pesan warning harus dikirim atau tidak.

Response formats	JSON
Requires authentication?	Ya (hanya user context)
Rate limited?	Ya

juga bahwa jumlah koneksi yang serentak per alamat IP masih terbatas terlepas dari aplikasi yang ada.

2.2.6 KIRI API

KIRI API adalah aplikasi pihak ketiga yang memungkinkan *programmer* mendapatkan data tentang info jalur transportasi publik. KIRI API dapat diakses dengan beberapa cara. Semua request harus berisikan API key, yang dapat diambil melalui KIRI API Management Dashboard. Berikut adalah spesifikasi dari KIRI API

- Routing Web Service
- Search Place Web Service
- Nearest Transports Web Service

Routing Web Service

Routing Web Service adalah salah satu KIRI API yang digunakan untuk mendapatkan langkah perjalanan dari lokasi asal menuju lokasi tujuan.

Berikut ini adalah parameter request yang diperlukan:

Parameter	Valid values	Description		
version	2	Memberitahukan bahwa layanan yang dipakai		
		adalah protokol versi 2		
mode	"findroute"	Mengintruksikan layanan untuk mencari rute		
locale	"en" or "id"	Respon bahasa yang digunakan		
start	lat,lng (both are decimal values)	Titik awal <i>Latitude</i> dan <i>longitude</i>		
finish	lat,lng (both are decimal values)	Titik akhir <i>Latitude</i> dan <i>longitude</i>		
presentation	"mobile" or "desktop"	Menentukan tipe presentasi untuk hasil keluaran.		
		Contoh, jika tipe presentasi "mobile",		
		maka link "tel:" akan ditambahkan di hasil.		
apikey	16-digit hexadecimals	API key yang digunakan		

Listing 2.1: code respond pencarian rute

2.2. Twitter API

delimited Menentukan apakah pesan harus dibatasi limitnya. stall warnings Menentukan apakah pesan warning harus dikirim atau tidak. with Menentukan apakah pesan informasi harus dikembalikan untuk user yang sudah diote: Menentukan apakah harus mengembalikan @replies. replies follow Termasuk tweet public tambahan dari daftar yang disediakan ID pengguna. track Termasuk tweet tambahan yang cocok dengan kata kunci tertentu. locations Termasuk tweet tambahan yang termasuk dalam batasan lokasi tertentu. stringify friend ids Mengirim list teman yang diterdiri dari array of integer dan array of string.

```
10
                           "human readable description, dependant on locale",
11
                           URL for ticket booking or null (future)
12
13
                           "walk" or "none" or others,
14
                           "walk" or vehicle_id or "none",
15
                           ["lat_1,lon_1", "lan_2,lon_2", ... "lat_n,lon_n"],
16
17
                           "human readable description, dependant on locale"
18
                           URL for ticket booking or null (future)
19
^{20}
                 ], $'' traveltime": any text string, null if and only if route is not found
21
22
23
             {
                 "steps": [ ... ],
"traveltime": "..."
^{24}
25
26
27
                 "steps": [ . . . ],
"traveltime": "..."
^{28}
29
30
            } ,
31
32
        ]
```

Ketika pencarian route berhasil yaitu dengan memberitahukan bahwa status "ok" seperti pada baris 2, maka server juga harus memberikan hasil dari rute, yang berisikan langkah-langkah yang disimpan di dalam array. Berikut ini adalah keterangan dari array tersebut:

- Index 0 (baris ke) berisikan "walk" atau "none" atau "others". Arti dari "walk" adalah jalan kaki, "none" berarti rute jalan tidak ditemukan, dan "others" berarti menggunakan kendaraan.
- Index ke 1 merupakan detail dari index ke 0 yang memiliki arti:
 - Jika berisikan "walk" berarti index ini pun harus berisikan "walk",
 - jika berisikan "none" maka index ini pun harus berisikan "none",
 - selain itu, maka field ini berisikan id kendaraan yang dapat digunakan untuk menambilkan gambar dari id kendaraan tersebut.
- Index ke 2 berisikan array of string, yang berisikan jalur dalam format "lat,lon". Lat adalah latitude, dan lon adalah longitude yaitu titik awal dan titik akhir.
- Index ke 3 merupakan bentuk yang dapat dibaca oleh manusia lalu akan ditampilkan kepada pengguna. Informasi tersebut dapat berupa:
 - % toicon sebuah ikon penanda yang menunjukkan titik awal atau "from". Biasanya digunakan untuk mode presentasi perangkat bergerak.

— % toicon sebuah ikon penanda yang menunjukkan titik akhir atau "to". Biasanya digunakan untuk mode presentasi perangkat bergerak.

• Index ke 4 berisi URL untuk pemesanan tiket untuk travel jika tersedia. Jika tidak ada maka nilai dari index ini bernilai null.

Search Place Web Service

Routing Web Service berguna untuk menemukan rute perjalanan berdasarkan latitute dan longitude koordinat, yang tidak nyaman bagi pengguna akhir. Layanan Search Place Web Service ini membantu untuk mengubah string teks untuk latitude dan longitude. Untuk permintaan routing, berikut parameter request yang diperlukan berikut penjelasannya:

version	2	Memberitahukan bahwa layanan yang dipakai		
		adalah protokol veris 2		
mode	"searchplace"	mengintruksikan layanan untuk mencari tempat		
region	"cgk" or "bdo" or "sub"	kota yang akan dicari tempatnya		
querystring	text apa saja dengan minimum	query string yang akan dicari menggunakan		
	text satu karakter	layanan ini		
apikey	16-digit hexadecimals	API key yang digunakan		

Berikut format kembalian dari Kiri API:

Listing 2.2: code respond pencarian lokasi

```
1
2
3
4
5
6
7
8
        "searchresult": [
                 "placename": "place name"
                 "location": "lat , lon"
                 "placename": "place name"
                 "location": "lat, lon"
10
11
            },
12
13
14
             "attribution_1", "attribution 2", ...
15
16
17
```

Ketika request find place berhasil, server akan mengembalikan place result, yang merupakan array dari langkah-langkah dan masing-masing berisi tentang deskripsi dalam format pemetaan:

- searchresult berisikan array dari hasil objek:
 - placename nama dari suatu tempat
 - location : latitude dan longitude dari suatu tempat
- attributions berisikan array string dan atribut tambahan yang akan ditampilkan

2.2.7 Nearest Transports Web Service

Nearest Transports Web Service digunakan untuk menemukan rute transportasi terdekat dengan titik yang diberikan.

Berikut parameter request yang diperlukan berikut penjelasanya:

2.2. Twitter API

version	2	Memberitahukan bahwa layanan yang dipakai		
		adalah protokol veris 2		
mode	"nearbytransports"	mengintruksikan layanan untuk mencari rute		
		transportasi terdekat		
start	latitude dan longitude	kota yang akan dicari tempatnya		
	(keduanya menggunakan nilai desimal)			
apikey	16-digit hexadecimals	API key yang digunakan		

Berikut format kembalian dari Kiri API:

Listing 2.3: code respond menemukan lokasi terdekat

```
2
3
4
5
6
7
8
9
10
        "status": "ok" or "error"
        "nearbytransports": [
             [
                  "walk" or "none" or others,
                 "walk" or vehicle_id or "none",
                  text string,
                 decimal value
                  "walk" or "none" or others,
                 "walk" or vehicle \_id or "none",
12
13
                  text string,
14
                 decimal value
15
             ],
\frac{16}{17}
        ]
```

Pencarian akan memberitahukan status berhasil ("ok") atau tidak ("error"), jika sukses maka respon akan mengembalikan array yang berisikan transportasi terdekat yang diurutkan dari yang terdeket ke yang terjauh. Berikut keterangan dari setiap array tersebut:

- Index ke 0 dapat berisi "walk" atau "none" atau "others". Artinya jika isi dari array tersebut "walk" berarti berjalan kaki, "none" jika rute tidak ditemukan dan "others" berarti menggunakan kendaran.
- Index ke 1 merupakan detail dari index 0. Artinya jika index 0 "walk" berarti index 1 harus "walk", "none" berarti index 1 harus "none" dan selain itu menyatakan id kendaraan yang mana bisa dipakai untuk ditampilkan gambarnya.
- Index ke 2 berisi nama kendaraan yang dapat dibaca oleh pengguna.
- Index ke 3 berisi jarak dalam satuan kilometer.

DAFTAR REFERENSI

 $[1] \ \ Twitter \ \ \textit{Twitter Documentation } 2014: \ \ \texttt{https://dev.twitter.com/overview/documentation}.$