

**SKRIPSI**

**PENGUKURAN APLIKASI USANG DI HTTPARCHIVE**



**Vinson Tandra**

**NPM: 2016730042**

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

**«tahun»**



# DAFTAR ISI

DAFTAR ISI	iii
DAFTAR GAMBAR	v
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	2
1.3 Tujuan . . . . .	2
1.4 Batasan Masalah . . . . .	2
1.5 Metodologi . . . . .	2
1.6 Sistematika Pembahasan . . . . .	2
<b>2 LANDASAN TEORI</b>	<b>3</b>
2.1 BigQuery[1, 2] . . . . .	3
2.1.1 Cloud Storage System . . . . .	3
2.1.2 SQL (Structured Query Language) [3] . . . . .	3
2.2 HTTP Archive [4] . . . . .	5
2.3 Web Almanac [5] . . . . .	9
2.4 OSEMN Framework . . . . .	11
2.4.1 Obtain Data . . . . .	11
2.4.2 Scrub Data . . . . .	11
2.4.3 Explore Data . . . . .	12
2.4.4 Model Data . . . . .	12
2.4.5 Interpreting Data . . . . .	12
2.5 Pengukuran Aplikasi Usang Pada Beberapa Website Populer Di Indonesia[6] . . .	12
2.5.1 Research Method . . . . .	12
2.5.2 Hasil Keseluruhan . . . . .	13
<b>3 ANALISIS</b>	<b>15</b>
3.1 Analisis Masalah . . . . .	15
3.2 . . . . .	15
DAFTAR REFERENSI	17
A KODE PROGRAM	19
B HASIL EKSPERIMEN	21



## DAFTAR GAMBAR

2.1	Jumlah baris pada tabel lighthouse di mobile . . . . .	5
2.2	Jumlah baris pada tabel pages di desktop . . . . .	6
2.3	Jumlah baris pada tabel pages di mobile . . . . .	6
2.4	Jumlah baris pada tabel technologies di desktop . . . . .	8
2.5	Jumlah baris pada tabel technologies di mobile . . . . .	9
2.6	<i>Algorithm to compare current version versus supported versions</i> . . . . .	13
B.1	Hasil 1 . . . . .	21
B.2	Hasil 2 . . . . .	21
B.3	Hasil 3 . . . . .	21
B.4	Hasil 4 . . . . .	21



# BAB 1

## PENDAHULUAN

Pada bab ini dijelaskan latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, dan sistematika pembahasan.

### 1.1 Latar Belakang

Di masa teknologi saat ini, banyak perusahaan yang menggunakan website sebagai tempat untuk mencari informasi. Terdapat banyak website yang menggunakan aplikasi yang sudah usang. HTTP Archive memiliki dataset penggunaan aplikasi suatu website yang dapat dilihat atau dianalisis menggunakan teknologi BigQuery. HTTP Archive memiliki dataset penggunaan aplikasi suatu website dari desktop dan mobile pada bulan Januari tahun 2016 sampai sekarang. Berdasarkan sumber pada web almanac<sup>1</sup>, dapat diambil kesimpulan bahwa website dibuka menggunakan browser di desktop dan mobile. Dataset yang digunakan berada pada label *technologies* merupakan dataset desktop dan mobile dengan nama tabel 2020\_08\_01. Dataset pada desktop memiliki 61.203.638 baris dan pada mobile memiliki 67.452.994 baris.

HTTP Archive<sup>2</sup> adalah sebuah proyek yang bersifat *open source* untuk melihat bagaimana website dibuat. Di dalam HTTP Archive terdapat data-data historis yang disediakan untuk menunjukkan bagaimana website terus berkembang dan proyek ini sering digunakan untuk penelitian. Didalam HTTP Archive terdapat dataset yang berisi jutaan web setiap bulan dan dapat dianalisis menggunakan teknologi BigQuery. BigQuery [2] adalah salah satu produk dari Google yang berbasis *cloud* dan dapat digunakan untuk menganalisis data tanpa harus memikirkan database. BigQuery dapat menjalankan *query* dalam skala *terabyte* dalam hitungan detik dan *petabyte* dalam hitungan menit.

Berdasarkan [6], dari 1.500 situs teratas menurut peringkat Alexa untuk pengunjung situs di Indonesia dan mengidentifikasi jenisnya aplikasi yang mereka gunakan beserta nomor versinya, lebih dari setengah atau 63% aplikasi yang digunakan berhasil dibandingkan dengan skrip yang telah dibuat dan hasilnya aplikasi tidak lagi didukung oleh pengelolanya.

Beberapa aplikasi sudah menyediakan fitur untuk meng-*update* ke versi yang paling baru tanpa harus menginstal ulang. Dalam kebanyakan kasus, versi aplikasi yang semakin baru sudah memperbaiki banyak kerentanan yang sudah diketahui. Beberapa aplikasi usang tidak memiliki pemberitahuan untuk meng-*update* sehingga pengguna tidak mengetahui jika terdapat *update*. Aplikasi yang baik biasanya memberikan update otomatis dan memberikan pesan yang efektif jika

---

<sup>1</sup><https://almanac.httparchive.org/en/2020/mobile-web>

<sup>2</sup>[https://github.com/HTTPArchive/httparchive.org/blob/main/docs/gettingstarted\\_bigquery.md](https://github.com/HTTPArchive/httparchive.org/blob/main/docs/gettingstarted_bigquery.md)

1 terjadi *update*.

2 Pada skripsi ini, akan dibuat sebuah penelitian untuk mengetahui seberapa besar penggunaan  
3 aplikasi usang pada seluruh website yang ada di dunia. Data dapat diambil dari HTTP Archive  
4 dengan melakukan *query* pada BigQuery. Pada penelitian ini akan dilakukan perhitungan pada  
5 jumlah aplikasi yang sudah diberi versi dan belum diberi versi. Versi aplikasi yang dipakai setiap  
6 website juga akan dibandingkan dengan versi aplikasi yang masih didukung berdasarkan *official*  
7 *website*-nya. Kemudian hasil tersebut akan ditampilkan dalam bentuk bar chart.

## 8 1.2 Rumusan Masalah

9 Berikut ini adalah rumusan masalah dari penelitian ini:

- 10 1. Bagaimana cara membaca data dari HTTP Archive?
- 11 2. Bagaimana mengimplementasi proyek [6] dengan menggunakan data yang lebih besar?
- 12 3. Berapa banyak website pada web almanac yang menggunakan aplikasi yang masih didukung?

## 13 1.3 Tujuan

14 Berikut ini adalah tujuan dari penelitian ini:

- 15 1. Membaca data dari HTTP Archive.
- 16 2. Mengimplementasi proyek [6] dengan menggunakan data yang lebih besar.
- 17 3. Mencari jumlah website pada web almanac yang menggunakan aplikasi yang masih didukung.

## 18 1.4 Batasan Masalah

19 Penelitian ini dibuat dengan batasan - batasan berikut:

## 20 1.5 Metodologi

21 Bagian-bagian pekerjaan skripsi ini adalah sebagai berikut:

- 22 1. Mempelajari teori HTTP Archive.
- 23 2. Mempelajari teori BigQuery.
- 24 3. Mempelajari bagaimana suatu website dikatakan usang.
- 25 4. Menganalisis beberapa website yang dikatakan usang.
- 26 5. Menulis dokumen skripsi.

## 27 1.6 Sistematika Pembahasan



## BAB 2

### LANDASAN TEORI

#### 2.1 BigQuery[1, 2]

Google memiliki salah satu produk yaitu BigQuery yang berbasis *cloud* dan dapat digunakan untuk menganalisis data tanpa harus memikirkan database. BigQuery memaksimalkan fleksibilitas dengan memisahkan memisahkan mesin komputasi yang menganalisa data. BigQuery dapat digunakan sebagai tempat penyimpanan dan data tersebut dapat dianalisis. Google meluncurkan BigQuery secara publik pada tahun 2012. Saat ini BigQuery sudah berkembang menjadi penyedia penyimpanan terstruktur berbasis *cloud* yang dikelola dan dihosting.

##### 2.1.1 Cloud Storage System

Selain sebagai tempat untuk menjalankan *query* dari data, saat ini BigQuery juga merupakan tempat penyimpanan data terstruktur di *cloud*. Data akan direplikasi ke beberapa lokasi yang berbeda secara geografis untuk meningkatkan ketersediaan dan ketahanan. Jika pusat data di Google pada suatu lokasi ditutup, data tetap dapat diakses tanpa terjadi gangguan. Data juga akan direplikasi dalam sebuah kluster agar tidak terjadi kehilangan data jika terjadi kegagalan perangkat keras.

##### 2.1.2 SQL (Structured Query Language) [3]

SQL adalah bahasa pemrograman menghasilkan, memanipulasi, dan mengambil informasi dari database relasional. Mengambil informasi dari database relasional harus menggunakan *query*. *Query* merupakan sintaks atau perintah yang digunakan untuk mengambil dan menghasilkan data dari database. Terdapat beberapa komponen atau klausa dari *query* yang digunakan mengambil dan menghasilkan data dari database dapat dilihat pada tabel 2.1

Clause Name	Purpose
Select	Menentukan kolom dari suatu tabel yang akan ditampilkan dalam <i>query result</i>
From	Mengidentifikasi tabel yang ingin diambil datanya
Where	Membatasi jumlah baris dalam <i>query result</i>
Group by	Mengelompokkan baris berdasarkan nilai kolom yang sama
Having	Membatasi jumlah baris dalam <i>query result</i> menggunakan data yang dikelompokkan
Order by	Mengurutkan <i>query result</i> berdasarkan satu atau lebih kolom

Tabel 2.1: *Query Clauses*

Didalam *query* juga terdapat beberapa fungsi agregat untuk melakukan operasi tertentu yang dapat dilihat pada tabel 2.2

Aggregate Function	Purpose
Max()	Mengembalikan nilai maksimal dari atribut sebuah tabel
Min()	Mengembalikan nilai minimum dari atribut sebuah tabel
Avg()	Mengembalikan nilai rata-rata dari atribut sebuah tabel
Sum()	Mengembalikan jumlah nilai dari atribut sebuah tabel
Count()	Mengembalikan jumlah baris dari atribut sebuah tabel

Tabel 2.2: *Aggregate Function*

### Querying Multiple Tables

Karena database relasional di-*design* dibentuk dengan mengamanatkan bahwa setiap entitas dibuat kedalam tabel yang terpisah, sehingga dibutuhkan mekanisme untuk menghubungkan beberapa tabel dalam *query* yang sama. Mekanisme ini disebut dengan join. Terdapat beberapa jenis join sebagai berikut:

- LEFT OUTER JOIN

Kata kunci kiri menunjukkan bahwa tabel di sisi kiri klausa from bertanggung jawab untuk menentukan jumlah baris dalam kumpulan hasil, sedangkan tabel di sisi kanan digunakan untuk memberikan nilai kolom setiap kali ditemukan kecocokan.

- RIGHT OUTER JOIN

Kata kunci kiri menunjukkan bahwa tabel di sisi kanan klausa from bertanggung jawab untuk menentukan jumlah baris dalam kumpulan hasil, sedangkan tabel di sisi kiri digunakan untuk memberikan nilai kolom setiap kali ditemukan kecocokan.

- FULL OUTER JOIN

Full outer join merupakan gabungan dari left outer join dan right outer join.

- CROSS JOIN

Cross join menggabungkan beberapa tabel dengan cara mengkali silangkan tabel tersebut tanpa menentukan kondisi apapun.

- INNER JOIN

Inner join menghubungkan dua atau lebih tabel dengan hubungan antara dua kolom.

### Subquery

*Subquery* merupakan query yang terkandung dalam *query* lain. Sebuah *subquery* selalu diapit dalam tanda kurung, dan biasanya dieksekusi terlebih dahulu sebelum *query* yang memuatnya. Tabel yang dikembalikan oleh *subquery* menentukan bagaimana tabel tersebut dapat digunakan dan operator mana yang dapat digunakan oleh *query* yang memuatnya untuk berinteraksi dengan tabel yang dikembalikan oleh *subquery*. Ketika query yang memuat telah selesai dieksekusi, tabel yang dikembalikan oleh subquery akan dibuang, membuat subquery bertindak seperti tabel sementara dengan cakupan pernyataan.

## 2.2 HTTP Archive [4]

HTTP Archive adalah sebuah *open-source project* yang melihat bagaimana website dibuat. HTTP Archive menyediakan data-data historis untuk melihat bagaimana website berkembang. HTTP Archive pertama sekali dimulai pada tahun 2010 oleh Steve Souders dan di-maintain oleh Pat Meenan, Rick Viscomi, Paul Calvano, and Barry Pollard. HTTP Archive memiliki keterbatasan seperti HTTP Archive hanya melihat halaman utama. Misalnya sebagian besar *website* terdiri dari banyak halaman web terpisah. Karena batasan ini sehingga ada kemungkinan bahwa suatu halaman yang dianalisis tidak mewakili sebuah situs website. Di dalam HTTP Archive terdapat dataset yang dapat diambil menggunakan teknologi BigQuery, dataset tersebut adalah sebagai berikut:

1. almanac

2. blink\_features

3. core\_web\_vitals

4. latest

5. lighthouse

Dataset pada lighthouse berisi tabel-tabel dari bulan Juni tahun 2017 sampai dengan sekarang yang terdiri dari website pada mobile. Dataset bulan Agustus tahun 2020 baris pada mobile memiliki 6.290.147 baris 2.1 yang dapat dianalisis. Masing-masing terdiri dari URL dan report. *URL (Uniform Resource Locator)* merupakan nama-nama domain dan *report*

Row	url	report
1	https://votesearch.utah.gov/	{ "userAgent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/84.0.4143.7 Mobile Safari/537.36", "environment": { "networkUserAgent": "Mozilla/5.0 (Linux; Android 10; SM-N960U) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/84.0.4147.105 Safari/537.36", "benchmarkIndex": 506, "lighthouseVersion": "6.1.1", "fetchTime": "2020-08-10T10:36:03.335Z", "requestedUrl": "https://votesearch.utah.gov/", "finalUrl": "https://votesearch.utah.gov/", "loadingAsExpected": true, "secondUrl": "https://votesearch.utah.gov/" }, "audits": { "is-on-https": { "id": "is-on-https", "title": "Does not use HTTPS", "description": "Modern web browsers have built-in warnings for non-HTTPS content, and Google Chrome will display a \"Not Secure\" warning in the address bar. HTTPS is the secure version of HTTP. The \"S\" stands for \"secure\". It encrypts the connection between the browser and the website, so that no one can intercept or tamper with the information that's being sent between them. This is important for the security of online transactions and other sensitive data. For more information, see: https://developer.chrome.com/docs/101/https/start-using-https/\" } } }
2	https://otricolore.ru/	{ "userAgent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/84.0.4143.7 Mobile Safari/537.36", "environment": { "networkUserAgent": "Mozilla/5.0 (Linux; Android 10; SM-N960U) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/84.0.4147.125 Safari/537.36", "benchmarkIndex": 4561, "lighthouseVersion": "6.2.0", "fetchTime": "2020-08-10T10:36:03.335Z", "requestedUrl": "https://otricolore.ru/", "finalUrl": "https://otricolore.ru/", "loadingAsExpected": true, "secondUrl": "https://otricolore.ru/" }, "audits": { "is-on-https": { "id": "is-on-https", "title": "Does not use HTTPS", "description": "Modern web browsers have built-in warnings for non-HTTPS content, and Google Chrome will display a \"Not Secure\" warning in the address bar. HTTPS is the secure version of HTTP. The \"S\" stands for \"secure\". It encrypts the connection between the browser and the website, so that no one can intercept or tamper with the information that's being sent between them. This is important for the security of online transactions and other sensitive data. For more information, see: https://developer.chrome.com/docs/101/https/start-using-https/\" } } }

Gambar 2.1: Jumlah baris pada tabel lighthouse di mobile

6. pages

Dataset pada pages berisi tabel-tabel dari bulan Januari tahun 2016 sampai dengan sekarang yang terdiri dari website pada desktop dan mobile. Dataset bulan Agustus tahun 2020 baris pada desktop memiliki 5.593.642 baris 2.2 dan pada mobile memiliki 6.347.640 baris 2.3 yang

dapat dianalisis. Masing-masing terdiri dari URL dan payload. *URL (Uniform Resource Locator)* merupakan nama-nama domain dan *payload*

Row	url	payload
1	http://laikrodistau.lt/	{ "startedDateTime": "2020-08-05T15:17:45.404+00:00", "title": "Run 1, First View http://laikrodistau.lt/", "id": "page_1_0_1", "pageTimings": { "onLoad": 11907, "onContentLoaded": -1, "startRender": 6900 }, "cpu.BlinkGC.LazyScriptEvaluation": 0, "cpu.V8.GC_MC_CLEAR_UPDATE_REFS": 0, "cpu.V8.GC_MC_MARK_EMBEDDER_TRACING_CLOSURE": 0, "cpu.V8.GC_SCAVENGER_COMPLETE_SWEEP_ARRAY_BUFFERS": 0, "cpu.V8.GC_MC_MARK_ROOTS": 0, "cpu.BlinkGC.IncrementalMarking": 0, "on_checked": 1, "cpu.V8.GC_MC_MARK_ROOTS": 0, "cpu.BlinkGC.IncrementalMarking": 0 }
2	http://lasvegasair.net/	{ "startedDateTime": "2020-08-05T15:18:44.005+00:00", "title": "Run 1, First View http://lasvegasair.net/", "id": "page_1_0_1", "pageTimings": { "onLoad": 10295, "onContentLoaded": -1, "startRender": 1100 }, "cpu.BlinkGC.LazyScriptEvaluation": 0, "cpu.V8.GC_MC_CLEAR_UPDATE_REFS": 0, "cpu.V8.GC_MC_MARK_EMBEDDER_TRACING_CLOSURE": 0, "cpu.V8.GC_SCAVENGER_COMPLETE_SWEEP_ARRAY_BUFFERS": 0, "cpu.V8.GC_MC_MARK_ROOTS": 0, "cpu.BlinkGC.IncrementalMarking": 0, "on_checked": 1, "cpu.V8.GC_MC_MARK_ROOTS": 0, "cpu.BlinkGC.IncrementalMarking": 0 }

Gambar 2.2: Jumlah baris pada tabel pages di desktop

Row	url	payload
1	https://www.business-takeover.com/	{ "startedDateTime": "2020-08-14T21:06:05.421+00:00", "title": "Run 1, First View https://www.business-takeover.com/", "id": "page_1_0_1", "pageTimings": { "onLoad": 43774, "onContentLoaded": -1, "startRender": 10000 }, "cpu.BlinkGC.LazyScriptEvaluation": 0, "cpu.V8.GC_MC_CLEAR_UPDATE_REFS": 0, "cpu.V8.GC_MC_MARK_EMBEDDER_TRACING_CLOSURE": 0, "cpu.V8.GC_SCAVENGER_COMPLETE_SWEEP_ARRAY_BUFFERS": 0, "cpu.V8.GC_MC_MARK_ROOTS": 0, "cpu.BlinkGC.IncrementalMarking": 0, "on_checked": 1, "cpu.V8.GC_MC_MARK_ROOTS": 0, "cpu.BlinkGC.IncrementalMarking": 0 }
2	https://www.geraldbrodylaw.com/	{ "startedDateTime": "2020-08-13T02:49:03.809+00:00", "title": "Run 1, First View https://www.geraldbrodylaw.com/", "id": "page_1_0_1", "pageTimings": { "onLoad": 23982, "onContentLoaded": -1, "startRender": 7900 }, "cpu.BlinkGC.LazyScriptEvaluation": 0, "cpu.V8.GC_MC_CLEAR_UPDATE_REFS": 0, "cpu.V8.GC_MC_MARK_EMBEDDER_TRACING_CLOSURE": 0, "cpu.V8.GC_SCAVENGER_COMPLETE_SWEEP_ARRAY_BUFFERS": 0, "cpu.V8.GC_MC_MARK_ROOTS": 0, "cpu.BlinkGC.IncrementalMarking": 0, "on_checked": 1, "cpu.V8.GC_MC_MARK_ROOTS": 0, "cpu.BlinkGC.IncrementalMarking": 0 }

Gambar 2.3: Jumlah baris pada tabel pages di mobile

## 7. requests

Dataset pada request berisi tabel-tabel dari bulan Januari tahun 2016 sampai dengan sekarang yang terdiri dari website pada desktop dan mobile. Dataset bulan Agustus tahun 2020 baris pada desktop memiliki 535.841.778 baris dan pada mobile memiliki 579.752.745 baris yang dapat dianalisis. Masing-masing terdiri dari URL dan payload. *URL (Uniform Resource Locator)* merupakan nama-nama domain dan *payload*.

## 8. response\_bodies

Dataset pada response\_bodies berisi tabel-tabel dari bulan Januari tahun 2016 sampai dengan sekarang yang terdiri dari website pada desktop dan mobile. Dataset bulan Agustus tahun

2020 baris pada desktop memiliki 215.621.667 baris dan pada mobile memiliki 270.249.686 baris yang dapat dianalisis. Masing-masing terdiri dari page, URL, body, truncated, dan requestId.

#### 9. sample\_data

#### 10. sample\_data\_2020

#### 11. scratchspace

#### 12. summary\_pages

Dataset pada summary\_pages berisi tabel-tabel dari bulan November tahun 2010 sampai dengan sekarang yang terdiri dari website pada desktop dan mobile. Dataset bulan Agustus tahun 2020 baris pada desktop memiliki 5.593.642 baris dan pada mobile memiliki 6.347.919 baris yang dapat dianalisis. Masing-masing terdiri dari pageid, createDate, archive, label, crawlid, wptid, wptrun, url, urlShort, urlhash, cdn, startedDateTime, TTFB, renderStart, onContentLoaded, onLoad, fullyLoad, visualComplete, PageSpeed, SpeedIndex, rank, reqTotal, reqHTML, reqJS, reqCSS, reqImg, reqGif, reqJpg, reqPng, reqFont, reqFlash, reqJson, reqOther, bytesTotal, bytesHTML, bytesJS, bytesCSS, bytesImg, bytesGif, bytesJpg, bytesPng, bytesFont, bytesFlash, bytesJson, bytesOther, bytesHtmlDoc, numDomains, maxDomainReqs, numRedirects, numErrors, numGlibs, numHttps, numCompressed, numDomElements, maxageNull, maxage0, maxage1, maxage30, maxage365, maxageMore, gzipTotal, gzipSavings, \_connections, \_adult\_site, avg\_dom\_depth, document\_height, document\_width, localStorage\_size, sessionStorage\_size, num\_iframes, num\_scripts, doctype, meta\_viewport, reqAudio, reqVideo, reqText, reqXml, reqWebp, reqSvg, bytesAudio, bytesVideo, bytesText, bytesXml, bytesWebp, bytesSvg, num\_scripts\_async, num\_scripts\_sync, usertiming.

#### 13. summary\_requests

Dataset pada response\_requests berisi tabel-tabel dari bulan November tahun 2010 sampai dengan sekarang yang terdiri dari website pada desktop. Dataset bulan Agustus tahun 2020 baris pada desktop memiliki 215.621.667 baris dan pada mobile memiliki 1.234.599 baris yang dapat dianalisis. Masing-masing terdiri dari requestid, pageid, startedDateTime, time, method, url, urlShort, redirectUrl, firstReq, firstHtml, reqHttpVersion, reqHeaderSize, reqBodySize, reqCookieLen, reqOtherHeader, status, respHttpVersion, respHeaderSize, respBodySize, respSize, respCookieLen, expAge, mimeType, respOtherHeader, req\_accept, req\_accept\_charset, req\_accept\_encoding, req\_accept\_language, req\_connection, req\_host, req\_if\_modified\_since, req\_if\_none\_match, req\_referer, req\_user\_agent, resp\_accept\_ranges, resp\_age, resp\_cache\_control, resp\_connection, resp\_content\_encoding, resp\_content\_language, resp\_content\_length, resp\_content\_location, resp\_content\_type, resp\_date, resp\_etag, resp\_expires, resp\_keep\_alive, resp\_last\_modified, resp\_location, resp\_pragma, resp\_server, resp\_transfer\_encoding, resp\_vary, resp\_via, resp\_x\_powered\_by.

#### 14. technologies

Dataset pada technologies berisi tabel-tabel dari bulan Januari tahun 2016 sampai dengan sekarang yang terdiri dari website pada desktop dan mobile. Dataset bulan Agustus tahun

2020 baris pada desktop memiliki 61.203.638 baris dapat dilihat pada gambar 2.4 dan pada mobile memiliki 67.452.994 baris 2.5 yang dapat dianalisis. Masing-masing terdiri dari 4 kolom yaitu *URL*, *category*, *app*, *info*. Pada kolom *URL* (*Uniform Resource Locator*) merupakan nama-nama domain, *category* merupakan jenis aplikasi yang digunakan pada website tersebut, *app* merupakan aplikasi yang digunakan website tersebut, *info* merupakan informasi tambahan dari aplikasi.

Row	url	category	app	info
1	https://www.3-king.com/	Analytics	Google Analytics	
2	https://www.fleabites.net/	Miscellaneous	Twitter Emoji (Twemoji)	
3	http://www.elcarnicero.cl/	Widgets	OWL Carousel	
4	https://thankyou.ws/	Analytics	Google Analytics	
5	https://rogerwaters.com/	Reverse proxies	Nginx	
6	http://www.palaciodelaslampadas.com.br/	JavaScript libraries	jQuery	2.1.1
7	https://copenhagencamping.dk/	CMS	WordPress	
8	https://eachat.ma/	Ecommerce	WooCommerce	4.3.0
9	https://advokat-bondarchuk.ru/	Blogs	WordPress	
10	https://passport.rsl.ru/	JavaScript libraries	jQuery	1.7.1

Rows per page: 100 ▾ 1 - 100 of 61203638 First page |< <

Gambar 2.4: Jumlah baris pada tabel technologies di desktop

Row	url	category	app	info
1	http://www.carobd.fr/	UI frameworks	Bootstrap	4.1.3
2	http://www.minikabebe.com/	Font scripts	Font Awesome	
3	https://sibirskisamojedcom.wordpress.com/	Blogs	WordPress	.
4	https://www.peaudeale.com/	Analytics	Google Analytics	
5	https://www.bestcours.com/	JavaScript libraries	jQuery	1.11.1
6	https://www.chirurgo-stefanoenrico.it/	UI frameworks	Bootstrap	
7	https://retrocores.com/	JavaScript libraries	jQuery	1.12.4
8	https://pakmule.com/	Web servers	Apache	
9	https://edilsonalves.com.br/	JavaScript libraries	jQuery	1.12.4
10	https://mobilierdasie.com/	Ecommerce	Google Analytics Enhanced eCommerce	

Rows per page: 100 1 - 100 of 67452994 First page

Gambar 2.5: Jumlah baris pada tabel technologies di mobile

15. urls

16. wappalyzer

## 2.3 Web Almanac [5]

Web Almanac adalah sebuah proyek yang dikelola oleh HTTP Archive. Misi web almanac adalah menggabungkan statistik mentah dan tren HTTP Archive dengan keahlian komunitas web. Semua metrik yang disediakan oleh web almanac dapat direproduksi secara publik menggunakan dataset di BigQuery. Kueri dapat ditelusuri dengan menggunakan semua bab di repositori GitHub web almanac yang dapat dilihat pada <sup>1</sup>:

### 1. Accessibility

Aksesibilitas web adalah tentang pencapaian fitur dan informasi serta memberikan akses lengkap ke semua aspek antarmuka bagi orang yang tidak memiliki akses. Sebuah produk digital atau situs web tidak lengkap jika tidak dapat digunakan oleh semua orang.

### 2. Caching

Caching adalah teknik yang memungkinkan penggunaan kembali konten yang diunduh sebelumnya. Caching melibatkan sesuatu seperti server atau web browser untuk menyimpan konten dan menandainya agar dapat digunakan kembali.

### 3. Capabilities

Capabilities memberikan *overview* tentang berbagai API web modern. Hal ini penting untuk

<sup>1</sup><https://github.com/HTTPArchive/almanac.httparchive.org/tree/main/sql/2020>

menjaga web tetap relevan sebagai platform.

#### 4. CMS

Istilah CMS mengacu pada sistem yang memungkinkan individu dan organisasi untuk membuat, mengelola, dan mempublikasikan konten. CMS pada konton web adalah sistem yang bertujuan untuk membuat, mengelola, dan menerbitkan konten untuk dikonsumsi dan dialami melalui internet.

#### 5. Compression

Menggunakan HTTP Compression membuat pemuatan situs lebih cepat dan menjamin pengalaman penggunaan yang lebih baik. Penggunaan compression yang efektif dapat mengurangi berat halaman dan meningkatkan kinerja web.

#### 6. CSS

CSS adalah bahasa yang digunakan untuk membuat tampilan dan format pada web dan media lainnya.

#### 7. Ecommerce

Ecommerce platform adalah perangkat lunak atau layanan yang memungkinkan untuk membuat dan mengoperasikan sebuah toko online.

#### 8. Fonts

Fonts adalah bagian penting dalam sebuah situs web dan tipografi adalah seni menyajikan teks tersebut dengan cara yang menarik dan efektif secara visual. Dalam pembuatan tipografi yang baik dibutuhkan pemilihan font yang sesuai. Dalam hal ini akan ditunjukkan bagaimana font web digunakan dan bagaimanafont tersebut dioptimalkan.

#### 9. HTTP

HTTP adalah protokol lapisan aplikasi yang dirancang untuk mentransfer informasi antara perangkat jaringan dan berjalan di atas lapisan lain dari tumpukan protokol jaringan. Dalam web almanac akan mengulas bagaimana status penerapan HTTP/2 atau HTTP versi dua pada saat ini.

#### 10. Jamstack

Jamstack adalah konsep arsitektur yang relatif baru yang dirancang untuk membuat web lebih cepat, lebih aman, dan lebih mudah untuk diskalakan. Dalam web almanac akan memperkirakan dan menganalisis pertumbuhan situs Jamstack, kinerja kerangka kerja Jamstack populer, serta analisis pengalaman pengguna nyata menggunakan metrik Core Web Vitals.

#### 11. Javascript

JavaScript adalah bahasa pemograman yang digunakan untuk menentukan perilaku.

#### 12. Markup

HTML adalah dasar dari sebuah website yang akan ditampilkan ke-*user*. Dalam web almanac mengacu pada kumpulan halaman *mobile*.

#### 13. Media

Pada web alamanac, media digunakan untuk menganalisa bagaimana menggunakan gambar dan video di web.

#### 14. Mobile-web

Saat ini, mobile-web sudah menjadi cara utama banyak orang untuk mengakses website. Dalam mobile-web akan terlihat tren saat ini pada mobile-web.



## 15. Page-weight

Page-weight adalah salah satu metrik sederhana yang tersedia. Memuat sebuah halaman akan memberikan gambaran tentang ukuran dari *resource* yang diambil atau di-*request*.

## 16. Performance

Dalam web almanac, akan melihat data kinerja di dunia nyata yang disediakan oleh Laporan Pengalaman Pengguna Chrome (CrUX) melalui lensa perkembangan baru tersebut serta menganalisis beberapa metrik relevan lainnya.

## 17. Privacy

Web almanac memberikan gambaran umum tentang keadaan privasi saat ini di web. Hal ini bertujuan untuk meningkatkan akuntabilitas pemroses data dan transparansi mereka terhadap pengguna. Dalam hal ini, kami membahas prevalensi pelacakan online dengan berbagai teknik dan tingkat adopsi spanduk persetujuan cookie dan kebijakan privasi oleh situs web.

## 18. PWA

Dalam web almanac, kita akan melihat setiap komponen yang membuat PWA seperti apa adanya, dari perspektif berbasis data.

## 19. Resource-hints

## 20. Security

Dalam web almanac, akan dilakukan menganalisis penerapan berbagai fitur keamanan secara mendalam dan dalam skala besar, kami mengumpulkan wawasan tentang berbagai cara pemilik situs web menerapkan mekanisme keamanan ini, didorong oleh insentif untuk melindungi penggunaannya.

## 21. SEO

Dalam web almanac, untuk mengidentifikasi dan menilai elemen dan konfigurasi utama yang berperan dalam pengoptimalan pencarian organik situs web.

## 22. Third-parties

Web almanac meninjau prevalensi konten pihak ketiga dan bagaimana hal ini telah berubah sejak 2019.

## 2.4 OSEMN Framework

OSEMN merupakan data science framework yang memberikan langkah-langkah pengerjaan proyek.<sup>2</sup>

### 2.4.1 Obtain Data

Obtain data berarti mengumpulkan data dari berbagai sumber. Langkah ini adalah langkah pertama. Mengumpulkan data sangat penting karena dalam melakukan sebuah proyek harus memiliki data. Data dapat didapat dengan meng-query dari database.

### 2.4.2 Scrub Data

Pada proses scrubbing data, data yang dikumpulkan tersebut akan dibersihkan atau difilter. Jika menggunakan data yang tidak difilter maka akan mempengaruhi keakuratan hasil akhir. Scrubbing

<sup>2</sup><https://towardsdatascience.com/5-steps-of-a-data-science-project-lifecycle-26c50372b492>

1 data bisa saja merupakan ekstraksi data dan bertukar nilai.

### 2 **2.4.3 Explore Data**

3 Pada explore data, akan dilakukan pengecekan terhadap tipe dari data. Kemudian data-data  
4 tersebut akan dikumpulkan dan dibandingkan sehingga mendapat kesimpulan dari data yang ingin  
5 dicari.

### 6 **2.4.4 Model Data**

7 Model data adalah pembuatan hasil akhir dari data yang diselidiki. Tujuan dari model data adalah  
8 mengelompokkan data untuk memahami logika di balik cluster tersebut.

### 9 **2.4.5 Interpreting Data**

10 Interpreting data mengacu pada penyajian data, penyampaian hasil agar dapat menunjukkan  
11 kesimpulan. Hasil-hasil yang ditunjukkan dapat berupa grafik-garfik agar dapat dijelaskan secara  
12 jelas dan aplikatif.

## 13 **2.5 Pengukuran Aplikasi Usang Pada Beberapa Website Populer** 14 **Di Indonesia[6]**

15 Pada bagian ini akan dijelaskan tentang research method dan hasil keseluruhan dari [6].

### 16 **2.5.1 Research Method**

17 1. Memilih list website yang populer

18 Memilih website paling populer dilakukan dengan mengambil daftar dari website teratas dari  
19 Alexa dengan negara tertentu.

20 2. Mengidentifikasi aplikasi yang dipakai website

21 Untuk setiap website akan dilakukan pengidentifikasian nomor versi yang dipakai. Hal ini  
22 dibantu dengan menggunakan *third party* yaitu Wappalyzer.

23 3. Mengelompokkan berdasarkan nama aplikasi dan ambil versi yang didukung

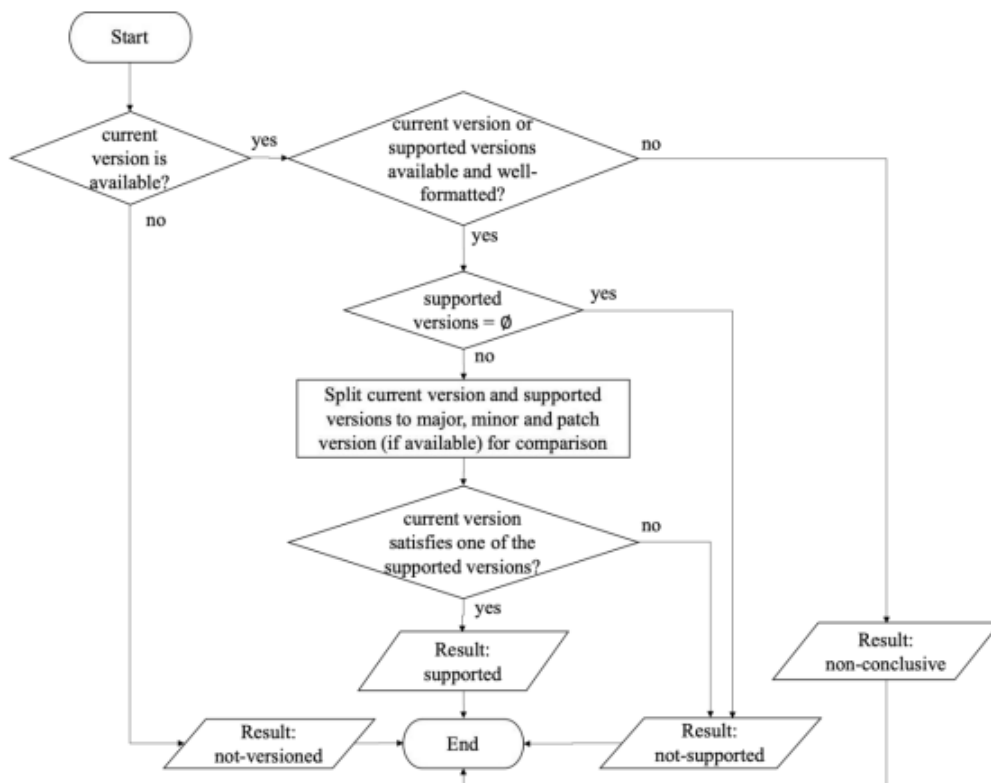
24 Untuk melihat nomor versi yang masih didukung akan dilakukan pencarian di website resmi  
25 dari setiap aplikasi. Terdapat beberapa website yang tidak dapat ditampilkan versinya,  
26 sehingga suatu website dapat didefinisikan didukung jika memenuhi kondisi sebagai berikut:

- 27 • Versi aplikasi yang didukung dapat dilihat secara eksplisit di dalam website.
- 28 • Dokumen untuk versi aplikasi tersebut masih tersedia.
- 29 • Aplikasi secara langsung memberikan pernyataan untuk versi yang masih didukung.

30 4. Membandingkan versi yang dipakai aplikasi saat ini dengan versi aplikasi yang didukung  
31 dapat dilihat pada gambar 2.6

32 Buka kembali setiap aplikasi kemudian menggunakan Wappalyzer untuk membandingkan  
33 versi aplikasi yang dipakai dengan versi aplikasi yang masih didukung. Klasifikasikan setiap  
34 aplikasi di setiap situs web menjadi salah satu dari berikut ini:

- *Not-versioned* berarti aplikasi yang terdeteksi oleh Wappalyzer tidak memiliki informasi versi sehingga tidak dapat dibandingkan.
- Non-konklusif dapat berarti salah satu dari dua:
  - Dapat mengambil nomor versi yang digunakan dalam aplikasi, tetapi kami tidak dapat menentukan apakah versi tersebut masih didukung atau tidak oleh pengelola.
  - Versi yang didukung untuk aplikasi tertentu tidak diketahui.
- Tidak didukung berarti dapat disimpulkan bahwa aplikasi yang digunakan menggunakan nomor versi yang tidak didukung oleh pengelola.
- Didukung berarti dapat disimpulkan bahwa aplikasi yang digunakan menggunakan nomor versi masih didukung oleh pengelola.



Gambar 2.6: *Algorithm to compare current version versus supported versions*

### 2.5.2 Hasil Keseluruhan

Pada paper[6], dari 1.500 URL yang dideteksi oleh Wappalyzer, hanya 1.439 URL yang berhasil diidentifikasi. Dari 1.500 URL tersebut ditemukan total 12.762 aplikasi yang dapat dilihat pada tabel 2.3

Result	Application count	Percentage
Not-versioned	8,980	70.37
Non-conclusive	1,409	11.04
Unsupported	1,508	11.82
Supported	865	6.78
Total	12,762	100.00

Tabel 2.3: Overall application count for measurement result

## **BAB 3**

## **ANALISIS**

1

2

3 Pada bab ini akan dijelaskan analisis masalah penelitian ini. Analisis meliputi analisis masalah,

### **4 3.1 Analisis Masalah**

### **5 3.2**



## DAFTAR REFERENSI

- [1] Tigani, J. dan Naidu, S. (2014) Google bigquery analytics. *asd*, **2**, 15–21.
- [2] Developer, G. Bigquery. <https://cloud.google.com/bigquery/docs/introduction>.
- [3] Beaulieu, A. (2005) *Learning SQL*, 1st ed edition. O'Reilly Media.
- [4] Souders, S. Http archive. <https://httparchive.org/faq>.
- [5] Pyltsyn, A. Web almanac. <https://almanac.httparchive.org/en/2020/methodology>.
- [6] Nugroho, P. A. dan Steven, H. (2013) Measuring unsupported applications in indonesia popular websites. *JITEKI*, **66**, 595–614.





# LAMPIRAN A

## KODE PROGRAM

Kode A.1: MyCode.c

```
1 // This does not make algorithmic sense,
2 // but it shows off significant programming characters.
3
4 #include<stdio.h>
5
6 void myFunction( int input, float* output ) {
7     switch ( array[i] ) {
8         case 1: // This is silly code
9             if ( a >= 0 || b <= 3 && c != x )
10                 *output += 0.005 + 20050;
11             char = 'g';
12             b = 2^n + ~right_size - leftSize * MAX_SIZE;
13             c = (--aaa + &daa) / (bbb++ - ccc % 2 );
14             strcpy(a,"hello_$@?");
15         }
16         count = ~mask | 0x00FF00AA;
17     }
18 }
19
20 // Fonts for Displaying Program Code in LATEX
21 // Adrian P. Robson, nepsweb.co.uk
22 // 8 October 2012
23 // http://nepsweb.co.uk/docs/progfonts.pdf
```

Kode A.2: MyCode.java

```
1 import java.util.ArrayList;
2 import java.util.Collections;
3 import java.util.HashSet;
4
5 //class for set of vertices close to furthest edge
6 public class MyFurSet {
7     protected int id; //id of the set
8     protected MyEdge FurthestEdge; //the furthest edge
9     protected HashSet<MyVertex> set; //set of vertices close to furthest edge
10    protected ArrayList<ArrayList<Integer>> ordered; //list of all vertices in the set for each trajectory
11    protected ArrayList<Integer> closeID; //store the ID of all vertices
12    protected ArrayList<Double> closeDist; //store the distance of all vertices
13    protected int totaltrj; //total trajectories in the set
14
15    /*
16     * Constructor
17     * @param id : id of the set
18     * @param totaltrj : total number of trajectories in the set
19     * @param FurthestEdge : the furthest edge
20     */
21    public MyFurSet(int id,int totaltrj,MyEdge FurthestEdge) {
22        this.id = id;
23        this.totaltrj = totaltrj;
24        this.FurthestEdge = FurthestEdge;
25        set = new HashSet<MyVertex>();
26        ordered = new ArrayList<ArrayList<Integer>>();
27        for (int i=0;i<totaltrj;i++) ordered.add(new ArrayList<Integer>());
28        closeID = new ArrayList<Integer>(totaltrj);
29        closeDist = new ArrayList<Double>(totaltrj);
30        for (int i = 0;i <totaltrj;i++) {
31            closeID.add(-1);
32            closeDist.add(Double.MAX_VALUE);
33        }
34    }
35
36 }
```



## LAMPIRAN B

### HASIL EKSPERIMEN

Hasil eksperimen berikut dibuat dengan menggunakan TIKZPICTURE (bukan hasil excel yg diubah ke file bitmap). Sangat berguna jika ingin menampilkan tabel (yang kuantitasnya sangat banyak) yang datanya dihasilkan dari program komputer.



Gambar B.1: Hasil 1



Gambar B.2: Hasil 2



Gambar B.3: Hasil 3



Gambar B.4: Hasil 4