

**ANALISIS STATIC SITE GENERATOR PADA WEB RESPONSIF
PORTAL BERITA**



TUGAS AKHIR

*Disusun dalam rangka memenuhi salah satu persyaratan
Untuk menyelesaikan program Strata-1 Departemen Teknik Informatika*

Fakultas Teknik Universitas Hasanuddin

Makassar

Disusun Oleh :

A.ARDIANSYAH YUSUF

D421 15 510

DEPARTEMEN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

MAKASSAR

2021

LEMBAR PENGESAHAN SKRIPSI
ANALISIS STATIC SITE GENERATOR PADA WEB RESPONSIF
PORTAL BERITA

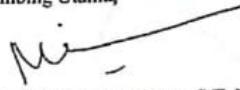
Disusun dan diajukan oleh

A. ARDIANSYAH YUSUF
D421 15 510

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka
Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika Fakultas
Teknik Universitas Hasanuddin pada tanggal 22 Desember 2021
dan dinyatakan telah memenuhi syarat kelulusan

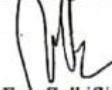
Menyetujui,

Pembimbing Utama,



Dr. Eng. Muhammad Niswar, S.T., M.IT.
Nip. 19730922 199903 1 001

Pembimbing Pendamping



Dr. Eng. Zulkifli Tahir, S.T., M.Sc.
Nip. 19840403 201012 1 004

Ketua Program Studi,

Dr. Amil Ahmad Ilham, S.T., M.IT
Nip. 19731010 199802 1 001

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama : A. ARDIANSYAH YUSUF

NIM : D421 15 510

Program Studi : TEKNIK INFORMATIKA

Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

ANALISIS STATIC SITE GENERATOR PADA WEB RESPONSIF PORTAL BERITA

Adalah karya tulis saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 13 Januari 2022

Yang Menyatakan



A. ARDIANSYAH YUSUF

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena berkat rahmat dan karunia-Nya sehingga tugas akhir yang berjudul “ANALISIS STATIC SITE GENERATOR PADA WEB RESPONSIF PORTAL BERITA” ini dapat diselesaikan sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Penulis menyadari bahwa dalam penyusunan dan penulisan laporan tugas akhir ini tidak lepas dari bantuan, bimbingan serta dukungan dari berbagai pihak, dari masa perkuliahan sampai dengan masa penyusunan tugas akhir. Oleh karena itu, penulis dengan senang hati menyampaikan terima kasih kepada:

1. Tuhan Yang Maha Esa atas semua berkat, karunia, kekuatan, serta pertolongan- Nya yang tiada batas, yang diberikan kepada penulis disetiap langkah untuk menyelesaikan tugas akhir ini;
2. Kedua Orang tua penulis, Bapak M. Yusuf Razak,S.H dan Ibu A. Besse Rosdiana Gau,S.H yang selalu memberikan dukungan, doa, dan semangat serta selalu sabar dalam mendidik penulis sejak kecil;
3. Bapak Dr. Eng. Muhammad Niswar,S.T., M.IT., selaku pembimbing I dan Bapak Dr. Eng. Zulkifli Tahir, S.T., M.Sc., selaku pembimbing II yang selalu menyediakan waktu, tenaga, pikiran dan perhatian yang luar biasa untuk mengarahkan penulis dalam penyusunan tugas akhir;
4. Bapak Dr. Amil Ahmad Ilham, ST., M.IT., selaku Ketua Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin atas bimbingannya selama masa perkuliahan penulis;

5. Ibu Anugrayani Bustamin.,S.T.,MT., selaku dosen pembimbing akademik yang telah memberikan bimbingan selama masa perkuliahan penulis;
6. Alfina Sulfiana yang selalu memberikan dukungan, semangat, serta bantuan kepada penulis selama proses penulisan draft tugas akhir;
7. Fiqar Aprialim dan Rahmat Reynaldi yang telah senantiasa mengajarkan serta memberikan arahan kepada penulis dalam menyusun tugas akhir;
8. Teman-teman Hypervisor FT UH atas dukungan dan semangat yang diberikan selama ini;
9. Saudara dan saudari Lovely atas dukungan yang diberikan selama ini;
10. Segenap Staf Departemen Teknik Informatika yang telah membantu penulis;
11. Orang-orang berpengaruh lainnya yang tidak sempat disebutkan oleh penulis.

Akhir kata, penulis berharap semoga Allah SWT. berkenan membalas segala kebaikan dari semua pihak yang telah banyak membantu. Semoga Tugas Akhir ini dapat memberikan manfaat bagi pengembangan ilmu. Aamiin.

Wassalam

Sinjai, 14 September 2021

Penulis

ABSTRAK

Penerapan situs web portal berita mengharuskan aplikasi web bersifat dinamis, namun seperti yang kita ketahui web dinamis lebih lambat jika dibandingkan dengan web statis. Hal ini disebabkan karena situs web diintegrasikan dengan *database* dan penggunaan *script PHP, JavaScript*, ataupun *script* lainnya sedangkan web statis konten informasi hanya diproses langsung dari HTML tanpa adanya *script* lain. Salah satu trend saat ini adalah membangun website dengan Static Site Generator (SSG) yang memberikan kemampuan pada aplikasi web dalam menghasilkan konten informasi secara dinamis ke dalam tampilan halaman web yang bersifat statis. *Framework* pendukung Static Site Generator yang menjadi perbincangan hangat saat ini adalah *framework Gatsby.js* dan *framework Next.js*. Proses analisis penelitian ini, akan membandingkan web portal berita yang menggunakan *framework Gatsby.js* dan *framework Next.js* sebagai *framework* pendukung SSG. Analisis yang dilakukan menggunakan indikator *performance* dari google lighthouse dan proses browser dari web dev-tools google chrome. Hasil penelitian menunjukkan bahwa next.js memiliki nilai waktu yang sedikit lebih baik pada pengujian *performance* tanpa cache, nilai waktu berbeda 369,44 ms. Namun gatsby.js memiliki nilai yang jauh lebih baik pada pengujian *performance* dengan cache, nilai waktu berbeda 908,62 ms. Untuk hasil proses browser menunjukkan bahwa web portal berita yang menggunakan *framework gatsby.js* memberikan nilai waktu proses browser yang lebih cepat dibandingkan dengan *framework next.js*.

Kata Kunci : static site generator, *gatsby.js*, *next.js*, *performance*, web portal berita, javascript.

ABSTRACT

The implementation of a news portal website requires that the web application be dynamic, but as we know dynamic web is slower than static web. This is because the website is integrated with the database and the use of PHP scripts, JavaScript, or other scripts, while the static web information content is only processed directly from HTML without any other scripts. One of the current trends is to build a website with a Static Site Generator (SSG) which gives web applications the ability to generate information content dynamically into a static web page display. The Static Site Generator supporting frameworks that are currently being discussed are the Gatsby.js framework and the Next.js framework. The analysis process of this research will compare news web portals that use the Gatsby.js framework and the Next.js framework as a framework for supporting SSG. The analysis is carried out using performance indicators from Google Lighthouse and the browser process from web dev-tools Google Chrome. The results show that next.js has a slightly better time value on performance testing without cache, the different time value is 369.44 ms. But gatsby.js has a much better value on performance testing with cache, the different time value is 908.62 ms. The browser process results show that the news web portal that uses the gatsby.js framework provides a faster browser processing time than the next.js framework.

Keywords: Static site generator, gatsby.js, next.js, performance, web portal news, javascript.

DAFTAR ISI

KATA PENGANTAR	ii
ABSTRAK	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian.....	2
1.4 Manfaat Penelitian.....	3
1.5 Batasan Masalah.....	3
1.6 Sistematika Penulisan.....	3
BAB II LANDASAN TEORI	5
2.1 Portal Berita.....	5
2.2 Web Statis dan Web Dinamis.....	5
2.2.1 Website Statis.....	5
2.2.2 Web Dinamis.....	6
2.2.3 <i>Page Load Time</i> Web Statis dan Web Dinamis.....	7

2.3	Static Site Generator (SSG).....	8
2.3.1	Gatsby.js.....	10
2.3.2	Next.js	12
2.4	Bahasa Pemrograman	15
2.4.1	JavaScript	15
2.4.2	Node.js	15
2.4.3	React.js	16
2.5	Basis Data.....	16
2.5.1	DBMS (Database Management System)	17
2.5.2	MySQL.....	18
2.5.3	phpMyAdmin	19
2.5.4	<i>JavaScript Object Notation (JSON)</i>	19
2.6	Github	20
BAB III METODOLOGI PENELITIAN.....		22
3.1	Tahapan penelitian.....	22
3.2	Waktu dan Lokasi Penelitian.....	23
3.3	Instrumen Penelitian dan Instalasi Kebutuhan Simulasi	23
3.3.1	Perangkat Keras	23
3.3.2	Perangkat Lunak.....	24
3.4	Tahap Persiapan.....	24

3.5	Gambaran Umum Sistem.....	25
3.5.1	<i>Gatsby.js</i>	25
3.5.2	<i>Next.js</i>	26
3.6	Hasil Pembuatan Sistem.....	28
3.7	Skenario Pengujian.....	32
3.6.1	Performance	32
3.6.2	Proses Browser.....	34
BAB IV HASIL DAN PEMBAHASAN		36
4.1.	Pengujian Performance.....	36
4.1.1	<i>Clear Storage Enable</i> (tanpa cache)	36
4.1.2	<i>Clear Storage Disable</i> (dengan cache)	46
4.2	Pembahasan Performance.....	56
4.3	Proses Browser	59
4.3.1	Pengujian Tanpa Cache	59
4.3.2	Hasil Pengujian Dengan Cache.....	63
4.4	Pembahasan Proses Browser.	66
BAB V PENUTUP.....		68
5.1.	Kesimpulan.....	68
5.2.	Saran	68

DAFTAR PUSTAKA	70
----------------------	----

DAFTAR GAMBAR

Gambar 2. 1 Alur Kerja Web Statis	6
Gambar 2. 2 Alur Kerja Web Dinamis.....	7
Gambar 2. 3 Alur kerja Sistem <i>Gatsby.js</i>	12
Gambar 2. 4 Alur kerja Sistem <i>Next.js</i>	14
Gambar 3. 1 Tahapan penelitian	22
Gambar 3. 2 Alur Kerja Sistem <i>Gatsby.js</i>	25
Gambar 3. 3 Localhost Browser <i>Gatsby.js</i>	26
Gambar 3. 4 Alur Kerja Sistem <i>Next.js</i>	27
Gambar 3. 5 Localhost Browser <i>Next.js</i>	27
Gambar 3. 6 Tampilan Sistem <i>Gatsby.js</i>	28
Gambar 3. 7 Tampilan Sistem <i>Next.js</i>	29
Gambar 3. 8 Tampilan Admin	29
Gambar 3. 9 Tampilan Database.....	30
Gambar 3. 10 Tampilan Data Yang Ditarik.....	30
Gambar 3. 11 Tampilan <i>Content</i>	31
Gambar 3. 12 Tampilan <i>Functionality</i>	31
Gambar 3. 13 Tampilan <i>Navigation</i>	32
Gambar 3. 14 Tampilan <i>Image</i>	32
Gambar 3. 15 Network Panel Chrome <i>Gatsby.js</i>	35
Gambar 3. 16 Network Panel Chrome <i>Next.js</i>	35
Gambar 4. 1 Alur Kerja Sistem <i>framework Gatsby.js</i>	58
Gambar 4. 2 Alur Kerja Sistem <i>framework Next.js</i>	58

DAFTAR TABEL

Tabel 4. 1 Hasil Pengujian Gatsby.js dan Next.js tanpa cache.	36
Tabel 4. 2 Hasil pengujian <i>First Contentful Paint</i> (FCP)	37
Tabel 4. 3 Hasil pengujian <i>Speed Index</i> (SI).....	39
Tabel 4. 4 Hasil pengujian <i>Largest Contentful Paint</i> (LCP).....	40
Tabel 4. 5 Hasil pengujian <i>Time To Interactive</i> (TTI).....	41
Tabel 4. 6 Hasil pengujian <i>Total Blocking Time</i> (TBT).....	43
Tabel 4. 7 Hasil pengujian <i>Cumulative Layout Shift</i> (CLS).....	45
Tabel 4. 8 Hasil Pengujian Gatsby.js dan Next.js kondisi Clear Storage disable.	47
Tabel 4. 9 Hasil pengujian <i>First Contentful Paint</i> (FCP)	47
Tabel 4. 10 Hasil pengujian <i>Speed Index</i> (SI).....	49
Tabel 4. 11 Hasil pengujian <i>Largest Contentful Paint</i> (LCP).....	50
Tabel 4. 12 Hasil pengujian <i>Time To Interactive</i> (TTI).....	51
Tabel 4. 13 Hasil pengujian <i>Total Blocking Time</i> (TBT).....	53
Tabel 4. 14 Hasil pengujian <i>Cumulative Layout Shift</i> (CLS).....	54
Tabel 4. 15 Hasil pengujian tanpa cache.....	59
Tabel 4. 16 Hasil <i>DOMContentLoaded</i> Gatsby.js dan Next.js	60
Tabel 4. 17 Hasil <i>Load</i> Gatsby.js dan Next.js.....	61
Tabel 4. 18 Hasil Pengujian Dengan Cache.....	63
Tabel 4. 19 Hasil <i>DOMContentLoaded</i> Gatsby.js dan Next.js.....	63
Tabel 4. 20 Hasil <i>Load</i> Gatsby.js dan Next.js.....	65

DAFTAR GRAFIK

Grafik 4. 1 Hasil Pengujian <i>First Contentful Paint</i> (FCP).....	38
Grafik 4. 2 Hasil Pengujian <i>Speed Index</i> (SI)	39
Grafik 4. 3 Hasil Pengujian <i>Largest Contentful Paint</i> (LCP).....	41
Grafik 4. 4 Hasil Pengujian <i>Time To Interactive</i> (TTI)	42
Grafik 4. 5 Hasil Pengujian <i>Total Blocking Time</i> (TBT).....	44
Grafik 4. 6 Hasil Pengujian <i>Cumulative Layout Shift</i> (CLS).....	46
Grafik 4. 7 Hasil Pengujian <i>First Contentful Paint</i> (FCP).....	48
Grafik 4. 8 Hasil Pengujian <i>Speed Index</i> (SI)	49
Grafik 4. 9 Hasil Pengujian <i>Largest Contentful Paint</i> (LCP).....	51
Grafik 4. 10 Hasil Pengujian <i>Time To Interactive</i> (TTI)	52
Grafik 4. 11 Hasil Pengujian <i>Total Blocking Time</i> (TBT).....	54
Grafik 4. 12 Hasil Pengujian <i>Cumulative Layout Shift</i> (CLS).....	55
Grafik 4. 13 Hasil Pengujian <i>DOMContentLoaded</i>	61
Grafik 4. 14 Hasil Pengujian <i>Load</i>	62
Grafik 4. 15 Hasil Pengujian <i>DOMContentLoaded</i>	64
Grafik 4. 16 Hasil Pengujian <i>Load</i>	66

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penerapan situs web portal berita mengharuskan aplikasi web bersifat dinamis sehingga memungkinkan *rendering* pada tampilan halamannya sesuai dengan konten informasi berita yang ada. Dalam pemrosesan tersebut, situs web diintegrasikan dengan *database* dan penggunaan *script PHP, JavaScript*, ataupun *script* lainnya sehingga dapat melakukan proses *rendering* untuk menampilkan konten informasi yang sesuai.

Proses ini tidak dapat dicapai pada sistem web yang bersifat statis dikarenakan konten informasi yang hanya diproses langsung dari HTML tanpa adanya *script* lain yang digunakan dalam menampilkan konten informasi tertentu. Walaupun demikian, terdapat keuntungan dalam penggunaan web statis, yaitu performa yang lebih cepat.

Static Site Generator (SSG) memberikan kemampuan pada aplikasi web dalam menghasilkan konten informasi secara dinamis ke dalam tampilan halaman web yang bersifat statis. Dengan penerapan SSG, sistem diharapkan tetap dapat memberikan keuntungan dalam penggunaan situs web yang bersifat statis, yaitu memiliki kinerja yang cepat dan sekaligus dapat menampilkan konten informasi pada situs.

Pada dewasa ini pembuatan website mulai berkembang dimana para web developer sudah memiliki banyak pilihan dalam membangun sebuah website.

Salah satu trend saat ini adalah membangun website dengan *static site generator* atau yang dikenal dengan sebutan SSG. Cukup banyak pilihan *Framework* untuk membuat website SSG, ada hugo, nuxtjs, nextjs, vuepress, gatsbyjs, gridsome, jigsaw dan masih banyak lagi. Orang biasanya memilih berdasarkan bahasa atau *Framework* apa yang selama ini digunakan. Namun saat ini *framework* yang menjadi perbincangan hangat adalah *framework Gatsby.js* dan *framework Next.js*.

Pada penelitian ini, peneliti membuat dua buah sistem berupa website portal berita responsif yang diintegrasikan dengan Static Site Generator (SSG) menggunakan *Framework Gatsby.js* dan *Framework Next.js*, kemudian peneliti meneliti perbandingan performa antara *Framework Gatsby.js* dengan *Framework Next.js* yang merupakan *Framework* pendukung *Static Site Generator* (SSG).

1.2 Rumusan Masalah

1. Bagaimana performa *Gatsby.js* sebagai *Framework Static Site Generators*?
2. Bagaimana performa *Next.js* sebagai *Framework Static Site Generators*?
3. Bagaimana perbandingan performa *Gatsby.js* dan *Next.js*?

1.3 Tujuan Penelitian

1. Untuk mengetahui performa *Gatsby.js* sebagai *Framework Static Site Generators*.
2. Untuk mengetahui performa *Next.js* sebagai *Framework Static Site Generators*.
3. Untuk mengetahui perbandingan performa *Gatsby.js* dan *Next.js*.

1.4 Manfaat Penelitian

1. Bagi peneliti, dengan melakukan analisis performa *Gatsby.js* dan *Next.js* sebagai *framework static site generators* kita dapat mengetahui mana *framework* yang lebih baik digunakan dalam pembuatan website portal berita yang terintegrasi *Static Site Generators (SSG)*.
2. Bagi institusi pendidikan, dapat digunakan sebagai referensi dalam pengembangan Website.

1.5 Batasan Masalah

1. Pembuatan web responsif portal berita menggunakan framework *Gatsby.js* dan *Next.js*.
2. *Software* yang digunakan untuk menguji performa website adalah Google Lighthouse dan web-dev tools.
3. Perangkat yang digunakan dalam membuat sistem adalah Laptop Asus dengan processor Intel Core i5 dan memori RAM 4GB

1.6 Sistematika Penulisan

Untuk memberikan gambaran singkat mengenai isi tulisan secara keseluruhan, maka diuraikan beberapa tahapan dari penulisan secara sistematis, yaitu:

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini akan dijelaskan mengenai teori-teori yang menunjang percobaan yang dilakukan.

BAB III METODOLOGI PENELITIAN

Bab ini berisi analisis kebutuhan sistem, perancangan sistem, dan skenario pengujian.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi hasil penelitian dan pembahasan penjabaran dari penelitian yang dilakukan.

BAB V PENUTUP

Bab ini berisi kesimpulan dari hasil penelitian dan saran yang diperlukan untuk pengembangan pada penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Portal Berita

Web portal adalah situs atau link web yang digunakan untuk tujuan tertentu dan mengarahkan pembaca supaya melihat, membaca, dan berinteraksi dengan informasi pada situs atau link website tersebut. Secara teknis, portal merupakan penyedia layanan informasi khusus untuk menarik minat pembaca agar masuk pada halaman website yang dicari, sesuai yang anda butuhkan. serta dapat diakses melalui berbagai perangkat seperti desktop, mobile, dan tablet. Tentunya di dalam web portal sendiri memiliki maksud dan tujuan khusus di dalamnya.

Berita secara umum adalah suatu kejadian atau peristiwa yang timbul atau sebagai bahan menarik publik atau media massa dalam bentuk penyampain atau laporan.

Portal Berita bisa di simpulkan suatu media massa berbentuk web yang didalamnya menyampaikan kejadian atau informasi secara digital dengan maksud dan tujuan sebagai media komunikasi dan informasi bahkan sebagai referensi sesuai kebutuhan pembaca yang ada dalam penyedia portal berita tersebut. Web portal berita memiliki kriteria *content, functionality, navigation, image, interactivity*.

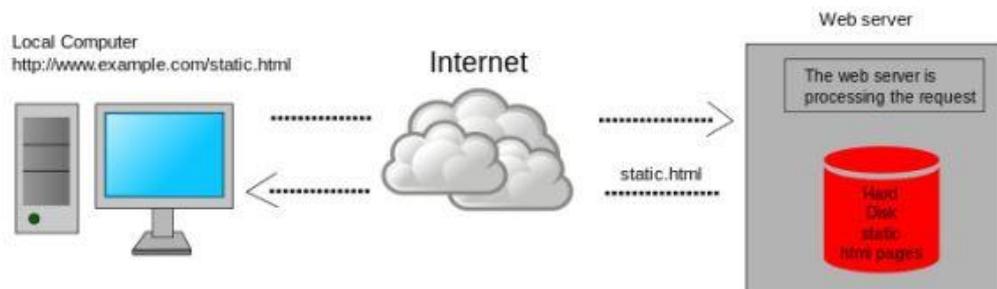
2.2 Web Statis dan Web Dinamis.

2.2.1 Website Statis

Cara kerja web statis sendiri cukup sederhana, yaitu client (web browser) mengirimkan sebuah request (*HTTP Request*) kepada server, untuk kemudian server akan mengirimkan kembali hasilnya kepada browser (*HTTP Response*) berupa halaman statis (*HTML*).

Dengan kata lain, web statis tidak membutuhkan sebuah database. Komunikasi keduanya hanya terjadi antara client (yaitu web browser) dan server (yaitu web server). Karena data tersebut telah disimpan di server tanpa

menggunakan database. Data tersebut akan dirender menggunakan metode (HTTP Response) ketika user melakukan sebuah *request* (HTTP Request) ke server tersebut. Web statis cenderung memiliki tampilan yang tetap, tidak dapat berubah secara berkala. Hanya orang dengan akses tertentu yang dapat merubah apa isi halaman tersebut.



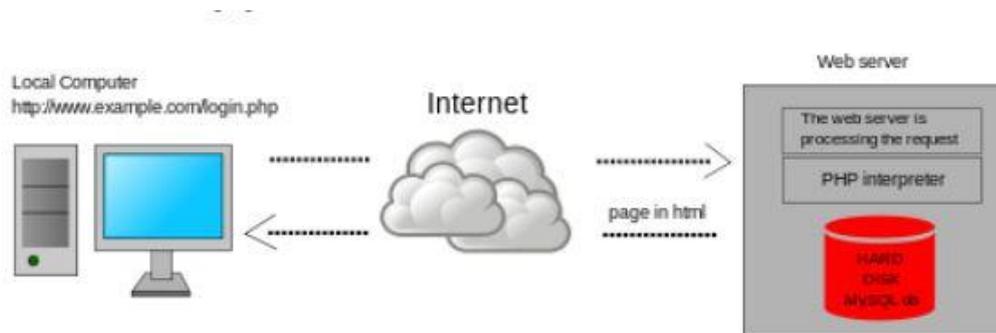
Gambar 2. 1 Alur Kerja Web Statis

Perbedaan web statis dan *Static Site Generator* adalah web statis ketika ingin melakukan *update* pada situs harus dilakukan secara manual yaitu seperti menulis ulang HTML berkali-kali yang tentunya merupakan hal yang sangat merepotkan. Namun, dengan SSG para developer dan pengelola situs dapat melakukan update situsnya secara otomatis sehingga alur kerja pengembangan situs menjadi lebih efektif dan menghasilkan tampilan yang lebih menarik.

2.2.2 Web Dinamis

Cara kerja web dinamis yaitu, client akan mengirimkan request (HTTP Request) tersebut ke server (web server), kemudian server akan meneruskannya ke sisi database, ketika hasil yang direquest oleh client (web browser) tersebut match (cocok), maka database akan memberikan hasil *request* ke sisi server, dan server akan meneruskannya ke sisi client.

Web dinamis membutuhkan database untuk menyimpan informasi / data-data dari pengguna. Sehingga data-data tersebut dapat dirubah-ubah sesuai kemauan client dan user yang memiliki hak akses di dalamnya.



Gambar 2. 2 Alur Kerja Web Dinamis

Perbedaan antara website dinamis pada umumnya dengan *Static Site Generator* adalah Website pada umumnya, user ketika merequest konten maka akan di proses oleh severnya atau ada logika-logika *backend* dan *fetch* data dari database, perbedaannya dengan SSG adalah semua hal tersebut dilakukan sebelum dideploy atau sebelum dikonsumsi oleh *user*. Semua logika-logika yang dilakukan di *backend* atau melakukan *request* ke database semuanya dilakukan sebelum dideploy atau sebelum dijalankan websitenya. Nantinya setelah disiapkan, atau setelah outputnya sudah jadi menjadi beberapa website HTML biasa, *user* hanya melakukan *request* ke HTML tersebut dan tidak lagi melakukan hal-hal yang dilakukan sebelumnya atau melakukan *request* lagi ke backendnya.

2.2.3 Page Load Time Web Statis dan Web Dinamis

Ada banyak alasan bagus mengapa waktu untuk membuka halaman yang lebih cepat itu penting. 40% pengguna meninggalkan situs web jika halaman dimuat lebih dari 3 detik. Jika sebuah situs akan dihosting secara dinamis dan statis dengan infrastruktur serupa dan konfigurasi server HTTP, maka dalam kasus terbaik waktu page load time web dinamis dapat mendekati web statis, namun tidak akan pernah lebih cepat daripada website statis. (Hillar Petersen)

Web statis mengalahkan web dinamis dalam hal kecepatan sehingga kesimpulan yang dapat diambil adalah web statis merupakan cara cepat untuk menyediakan halaman kepada user. (Levi Nunnink)

2.3 Static Site Generator (SSG)

Website berfungsi untuk menampilkan berbagai konten kepada penggunanya (*user*). Pertumbuhan teknologi yang kian cepat, membuat industri *web development* terus berkembang untuk membuat website yang lebih efisien dari sisi proses *maintaining* dan *updating*.

Oleh karena itu, Static Site Generator (SSG) hadir sebagai cara yang efisien dalam pengelolaan website. Dengan SSG yang dikenal juga dengan Jamstack ini, memudahkan para developer sehingga tidak perlu lagi menulis banyak HTML untuk memperbarui konten di website.

Static Site Generator (SSG) adalah tool yang akan membantu mengubah website yang dibangun menjadi file-file statis HTML, CSS dan Javascript untuk dikonsumsi oleh pengguna nantinya. Pada dasarnya SSG adalah istilah umum untuk sebuah alat atau software yang bisa menghasilkan halaman statis. Adapun kelebihan dari SSG, sebagai berikut.

1. Lebih Cepat

Dengan menggunakan halaman yang statis, akan mempercepat proses *Loading* dibandingkan dengan halaman dinamis yang memerlukan pembacaan script dan database.

2. Murah Biaya

Untuk mempublish halaman statis tidak memerlukan hosting yang harga mahal, karena tidak memerlukan hosting yang dilengkapi dengan database. Bahkan banyak yang menyediakan hosting gratis untuk SSG ini. Diantaranya

yang populer adalah Github, GitLab, atau Bitbucket, kemudian kita bisa menghubungkan *repository* GitHub kita dengan *platform* seperti Netlify, GitHub Pages, atau Vercel.

3. Keamanan Terjaga

Karena tidak terlalu banyak kode yang berhubungan dengan data atau backend, otomatis kemungkinan adanya celah keamanan akan semakin sempit..

Sedangkan kelemahan dari SSG, yaitu:

Bagi orang awam yang belum terbiasa dengan terminal, atau command line/command prompt, pasti akan kesulitan. Apalagi jika kita malas membaca dokumentasi yang disediakan. Untuk menambah *resource* external seperti misalnya menambahkan meta tag, plugin, atau modul yang dibutuhkan dilakukan secara manual. Dan lagi-lagi kita harus terbiasa dengan terminal/command prompt. Tentu sangat berbeda dengan web dinamis seperti CMS yang sangat mudah.

Static Site Generator menawarkan pengelolaan konten situs yang sederhana, sehingga para pengguna tidak terlalu membutuhkan database karena website statis tidak memerlukan update secara berkala. Beberapa contoh bahasa pemrograman yang mendukung *Static Site Generator* seperti Javascript beserta *Framework* dapat menggunakan *Next.js* dan *Gatsby.js*.

2.3.1 Gatsby.js

Gatsby.js merupakan salah satu *Framework* pendukung untuk pembuatan website *Static Site Generator*. Gatsby adalah kerangka kerja sumber terbuka dan gratis berbasis React yang membantu pengembang membangun situs web dan aplikasi yang sangat cepat. Ini menawarkan kinerja, skalabilitas, dan keamanan bawaan. Salah satu fitur terbesar *Gatsby JS* adalah mempunyai banyak *plugin* yang bisa membantu mempermudah kita dalam membuat suatu *project*. Adapun *plugin* yang ada dalam *Gatsby.js*, yaitu:

a) SEO

SEO (*Search Engine Optimization*) atau optimisasi mesin pencarian sangat penting untuk memungkinkan search engine menilai apakah website kita layak untuk ditampilkan di posisi teratas hasil pencarian atau tidak. Maka dari itu ketika kita membuat sebuah project Gatsby.js secara otomatis sudah ter-install plugin “gatsby-plugin-react-helmet” dan “react-helmet”. Penggunaannya tinggal kita sesuaikan dengan kebutuhan.

b) *Image Optimization & Lazy Load*

Gambar yang besar dan tidak optimal akan sangat mempengaruhi performa, bahkan memperlambat website kita. Dengan plugin yang sudah disediakan Gatsby JS kita bisa mengoptimalkan gambar yang ada di website. Plugin “gatsby-image”, “gatsby-transformer-sharp” dan “gatsby-plugin-sharp” adalah solusi yang ditawarkan untuk mengoptimalkan gambar pada website.

c) JSX

Sintaksis ini di kenal dengan sebutan JSX, dan sintaksis ini adalah sebuah sintaksis ekstensi untuk *JavaScript*. Kami sarankan menggunakannya dengan React untuk mendeskripsikan bagaimana antarmuka pengguna seharusnya terlihat. JSX mungkin akan mengingatkan Anda dengan sebuah bahasa *templat*, bedanya adalah JSX telah dilengkapi dengan kekuatan penuh dari JavaScript.

JSX akan menghasilkan “elemen” React. Kita akan mulai mengeksplor bagaimana *me-render* mereka ke dalam DOM di bagian berikutnya. Di bawah ini, Anda akan menemukan dasar-dasar JSX yang Anda butuhkan untuk memulai.

d) ESLint & Prettier

ESLint dan Prettier sangat bermanfaat untuk membuat code kita lebih konsisten dan rapi. ESLint adalah tool yang membantu untuk mengecek setiap code yang kita buat apakah sudah sesuai aturan atau belum. Dan Prettier adalah code formatter yang dapat membuat code kita terlihat rapih dan indah.

e) Husky & Lint Staged

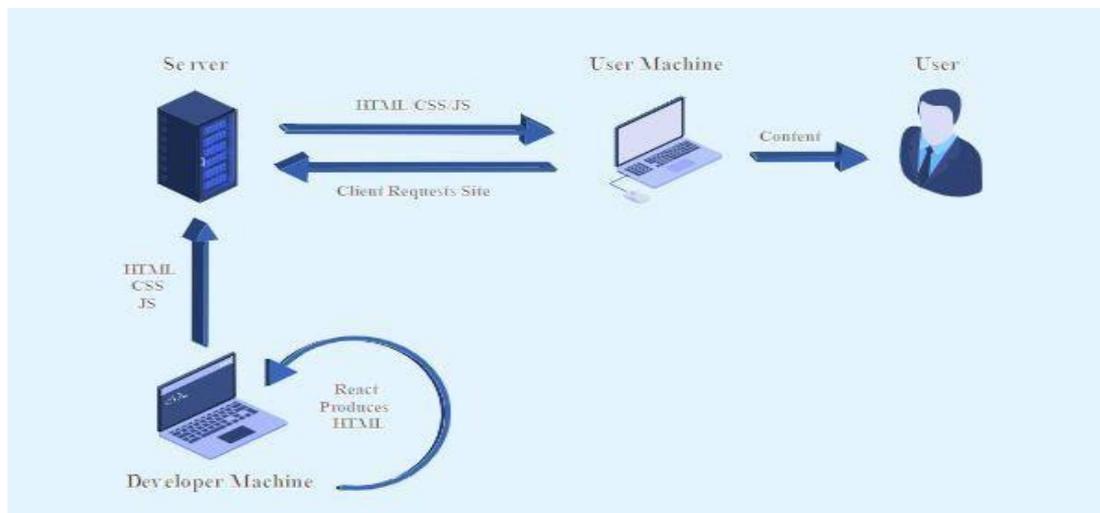
Peran Husky dan Lint Staged ini bermanfaat untuk mengecek ulang apakah sudah sesuai aturan atau belum. File yang kita commit yang belum mengikuti aturan dapat dicegah oleh Husky. Sedangkan Lint Staged bermanfaat untuk menjalankan Linter pada file yang kita commit.

f) MDX

Sebagian besar dari kita mungkin menggunakan markdown untuk membuat konten pada Gatsby JS. Tetapi markdown hanya sebatas konten yang

berbasis text saja, dan itu membuat kita menjadi terbatas. Maka ada MDX, superset dari markdown yang memungkinkan kita bisa melakukan import dan render component ke dalam markdown.

Adapun alur kerja sistem *Gatsby.js* dapat dilihat pada gambar berikut:



Gambar 2.3 Alur kerja Sistem *Gatsby.js*

2.3.2 Next.js

Next.js adalah *Framework* web berbasis React yang bisa digunakan untuk membangun aplikasi web yang sepenuhnya dirender di sisi server (SSR), SSG (*Static Site Generator*), bahkan aplikasi SSG/SSR hybrid. *Next.js* saat ini telah digunakan oleh beberapa perusahaan besar seperti Nike, Netflix, dan Playstation. *Next.js* dibuat karena masalah pre-render secara statis di beberapa halaman yang membuat SEO (Search Engine Optimization) dikarenakan file JavaScript harus di *Load* dan menentukan komponen apa yang harus ditampilkan.

Next.js mempunyai beberapa keunggulan seperti automatic code splitting

yang digunakan sebagai pemecah code agar lebih cepat saat meLoad halaman, memiliki konsep SSR (Server Side Rendering) yang akan membantu untuk memudahkan dalam menggunakan SEO. *Next.js* juga memiliki direktori pages yang akan sangat memudahkan pada proses routing dan dapat dimodifikasi sesuai dengan kebutuhan (Hendrikus, 2020)

Berikut beberapa fitur utama pada *Next.js* :

1. *Hot Code reloading*

Next.js akan mereload page ketika di deteksi terdapat perubahan yang disimpan.

2. *Automatic routing*

Next.js memiliki direktori pages sehingga untuk routing akan disesuaikan dengan file yang kita simpan di direktori pages namun tentunya kita juga masih dapat memodifikasi sesuai dengan kebutuhan kita.

3. *Single file component*

Dengan adanya *styled-jsx* kita menjadi sangat mudah untuk membuat style langsung di komponen yang kita buat.

4. *Server Rendering*

Kita bisa merender (optional) component react pada server side sebelum mengembalikan page html ke *client* sehingga akan baik untuk SEO tentunya.

5. *Automatic Code Splitting*

Page yang akan di render hanya dengan library yang digunakan pada page tersebut, sehingga akan lebih ringan.

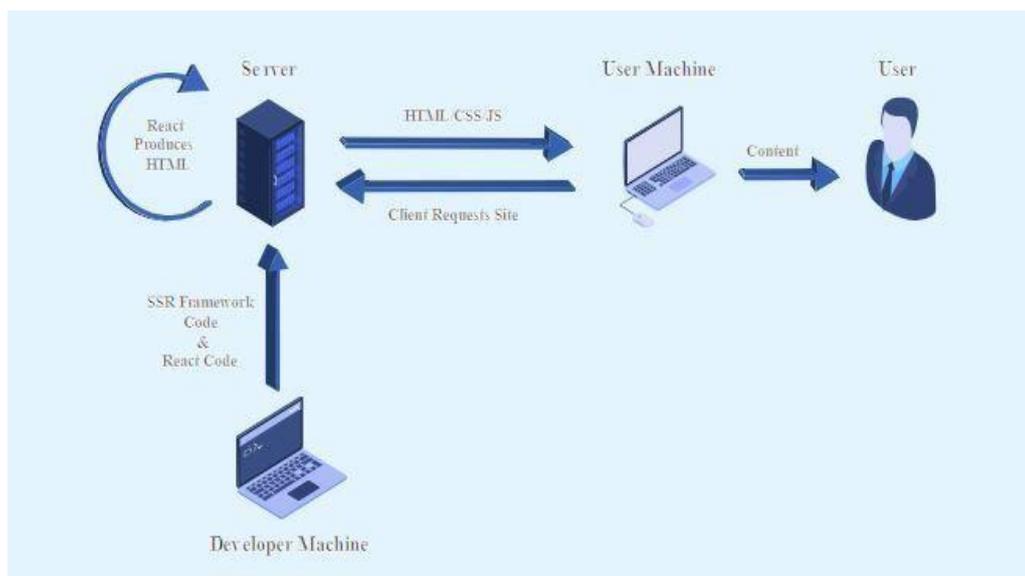
6. *Dynamic component*

Kita dapat mengimport javascript module atau react component secara dinamis.

7. *Static export*

Dengan menggunakan keyword `next export`, kita dapat mengexport static site dari app kita.

Adapun alur kerja sistem *Next.js* dapat dilihat pada gambar berikut:



Gambar 2. 4 Alur kerja Sistem *Next.js*

2.4 Bahasa Pemrograman

Bahasa pemrograman adalah instruksi standar untuk memerintah komputer dan merupakan suatu himpunan dari aturan sintaks dan semantik yang di pakai untuk mendefinisikan program komputer.

2.4.1 JavaScript

JavaScript adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokument HTML yang ditampilkan pada sebuah browser menjadi lebih interaktif, tidak sekedar indah saja. JavaScript memberikan beberapa fungsionalitas ke dalam halaman web, sehingga dapat menjadi sebuah program yang disajikan dengan menggunakan antar muka web.

JavaScript merupakan yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada web browser seperti Google Chrome, Mozilla Firefox, Opera Mini dan sebagainya. JavaScript pertama kali dikembangkan pada pertengahan dekade 90'an. Meskipun memiliki nama yang hampir serupa, JavaScript berbeda dengan bahasa pemrograman Java. Untuk penulisannya, JavaScript dapat disisipkan di dalam dokumen HTML ataupun dijadikan dokumen tersendiri yang kemudian diasosiasikan dengan dokumen lain yang dituju. (Omar dkk, 2018)

2.4.2 Node.js

Node.JS adalah sistem perangkat lunak yang didesain untuk pengembangan aplikasi web. Node.JS dapat juga disebut sebagai *runtime environment*. Aplikasi

ini ditulis dalam campuran Bahasa C++ dan juga JavaScript, mempunyai model event driven(*basis event*) dan asynchronousI/O. Tidak seperti kebanyakan bahasa JavaScript yang dijalankan pada web browser, Node.JS dieksekusi sebagai aplikasi server. Node.JS dapat berjalan di server karena dukungan dari V8 Engine buatan Google dan beberapa modul bawaan yang terintegrasi seperti modul http, modul filesystem, modul security dan beberapa modul penting lainnya. (Rahmat, 2017)

2.4.3 React.js

ReactJs merupakan kerangka kerja *open source* yang menggunakan *library* javascript untuk membuat *user interface* dan React biasa digunakan untuk menangani pengembangan pada aplikasi single-page dan aplikasi mobile. ReactJS memiliki keunggulan dimana kerangka kerja ini memberikan kecepatan, *simplicity*, dan *scalability* (Tung, 2018) React yang dikembangkan oleh facebook untuk memfasilitasi pengembang dalam membuat komponen UI yang lebih interaktif, *stateful*, & *reusable* (Kumar & Singh, 2016).

Dalam kaidah MVC (Model View Control) react hanya merepresentasikan pada bagian View saja dan ini merupakan bagian terbaik dalam penyederhanaan. React dikembangkan oleh facebook dengan berusaha untuk menyediakan kecepatan, kesederhanaan, dan skalabilitas. Beberapa fitur utama adalah JSX, komponen *stateful*, dan virtual document object model.

2.5 Basis Data

Basis data dapat dianggap sebagai tempat untuk sekumpulan berkas data

yang terkomputerisasi dengan tujuan untuk memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan. Basis data yang membutuhkan media untuk melakukan penyimpanan dan pengelolaan data yang ada yang diolah oleh program sql maupun nosql. Keduanya adalah media yang digunakan untuk melakukan pengolahan basis data yang digunakan sesuai dengan kebutuhan pengembang. (Haidar, 2007)

2.5.1 DBMS (Database Management System)

Pengelolaan database secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak yang khusus yang akan menentukan bagaimana data diorganisasi, disimpan, diubah dan diambil kembali. Ia juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, pemaksaan keakuratan/konsistensi data dan sebagainya.

Database berbeda dengan Database Management System (DBMS). DBMS adalah kumpulan program yang digunakan untuk mendefinisikan, mengatur, dan memproses database; sedangkan database itu sendiri merupakan alat atau tool yang berperan untuk membangun struktur data tersebut. Saat ini kita bisa menemukan banyak program DBMS, seperti MySQL, Oracle, Interbase/Firebird, IBM, DB2 dan sebagainya. Beberapa DBMS ada yang hanya bisa beroperasi di komputer mainframe, beberapa hanya dapat dijalankan di mini komputer dan juga ada yang hanya dapat dijalankan di *Personal Computer* (PC). Maka dari itu, untuk tren sekarang, DBMS yang banyak digunakan adalah DBMS yang memiliki fasilitas lintas- platform atau cross-platform (berjalan di beberapa platform yang berbeda).

Dalam satu DBMS dapat memiliki lebih dari satu database. DBMS juga sering disebut sebagai *server* database. Sebagai contoh sederhana, dalam satu DBMS (*server* database) MySQL, dapat terdiri dari beberapa database (Hesananda, 2017).

2.5.2 MySQL

MySQL adalah DBMS yang *open source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah database server yang gratis dengan lisensi GNU *General Public License* (GPL) sehingga dapat anda pakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada.

Seperti yang sudah disinggung di atas, MySQL masuk ke dalam jenis *lational Database Management System*. Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada MySQL. Contohnya di dalam MySQL sebuah database terdapat satu atau beberapa tabel.

merupakan suatu bahasa yang dipakai di dalam pengambilan data pada *relational database* atau database yang terstruktur. Jadi MySQL adalah database *management system* yang menggunakan bahasa SQL sebagai bahasa penghubung antara perangkat lunak aplikasi dengan database server. (Jubilee Enterprise, 2004)

2.5.3 phpMyAdmin

PhpMyAdmin adalah sebuah aplikasi/perangkat lunak bebas (opensource) yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi database SQL melalui jaringan lokal maupun internet. phpMyAdmin mendukung berbagai operasi SQL, diantaranya (mengelola basis data, tabel-tabel, bidang (*fields*), relasi (*relations*), indeks, pengguna (*users*), perijinan (*permissions*), dan lain-lain) (Standsyah, 2017).

Perbedaan phpMyAdmin dengan MySQL terletak pada fungsi. phpMyAdmin merupakan alat untuk memudahkan dalam mengoperasikan database MySQL, sedangkan MySQL adalah database tempat penyimpanan data. phpMyAdmin sendiri digunakan sebagai alat untuk mengolah/ mengatur data pada MySQL (Standsyah, 2017).

2.5.4 *JavaScript Object Notation (JSON)*

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript.

JSON adalah salah satu bahasa markup yang dapat melakukan pertukaran data dimana JSON ini dibuat berdasarkan JavaScript dan pastinya sintaknya lebih ke JavaScript. Dengan membuat sebuah JSON sama halnya dengan kita membuat sebuah object pada JavaScript itu sendiri. Di dalam membuat JSON pasti kita bakal berkenalan dengan yang namanya array pada JavaScript sehingga

memudahkan bagi para developer/programmer (Sudirman 2016).

2.6 Github

Github adalah software Hosting untuk proyek open source yang menggunakan Tool System revisi kontrol Git. Jadi Git adalah tool untuk melakukan revisi code, sedangkan github adalah webhostingnya. Mudahnya Github adalah Webhosting untuk proyek-proyek software seperti Google code atau sourceforge.net. Dalam bahasa gaul, Github adalah jejaring sosial untuk software developer (Putra, 2014). Yang membedakan Github dari Project hosting yang lain bisa kita lihat di bawah ini:

- 1) Github menggunakan Git sebagai tool utama dalam merevisi code.
- 2) Github menyediakan free hosting untuk proyek open source, namun Github juga menawarkan hosting berbayar untuk proyek perusahaan atau pribadi yang sifatnya privat.
- 3) Github ditargetkan untuk para developer yang bekerja secara tim dan tidak disatu tempat.
- 4) Github mempunyai fitur Social seperti Twitter seperti follow atau favorit (Star).
- 5) Github mendukung semua bahasa pemrograman, anda bebas memakai bahasa dan tool yang biasa anda pakai.

Beberapa fitur Sosial di Github:

- a. Github user profile.
- b. Follow, Kita bisa memfollow profil progammer lain di github.

- c. Star, fungsinya sama dengan Bookmark.
- d. Watch, memonitor repository tertentu. Setiap ada perubahan di Repo, maka kita akan dikirim notifikasi.
- e. Fork, mengcopy keseluruhan code repository ke repo kita sendiri.