

# 5 Tahap Membuat Dashboard Admin Untuk Kemudahan Programmer Dengan **ReactJS** dan **TailwindCSS**

**(Studi Kasus : Data Koleksi Museum)**

**PENULIS :**

Roni Andarsyah

Ariq Rafi Kusumah



# **5 Tahap Membuat Dashboard Admin Untuk Kemudahan Programmer Dengan ReactJS dan TailwindCSS**

(Studi Kasus: Data Koleksi Museum)

**Roni Andarsyah  
Ariq Rafi Kusumah**



**PT. Penerbit Buku Pedia  
2023**

# 5 Tahap Membuat Dashboard Admin Untuk Kemudahan Programmer Dengan ReactJS dan TailwindCSS

(Studi Kasus: Data Koleksi Museum)

***Penulis:***

Roni Andarsyah  
Ariq Rafi Kusumah

***ISBN:***

***Editor:***

Rolly Maulana Awangga

***Penyunting:***

Rolly Maulana Awangga

***Desain sampul dan Tata letak:***

Ariq Rafi Kusumah

***Font:***

Calibri

***Penerbit:***

PT. Penerbit Buku Pedia

***Redaksi:***

Athena Residence Blok. E No. 1, Desa Ciwaruga,  
Kec. Parongpong, Kab. Bandung Barat 40559  
Tel. 628-775-2000-300  
Email : penerbit@bukupedia.co.id

***Distributor:***

Informatics Research Center  
Jl. Sariasih No. 54  
Bandung 40151  
Email : irc@ulbi.ac.id

Cetakan Pertama, 2023

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan  
dengan cara apa pun tanpa ijin tertulis dari penerbit

# PRAKATA

**D**engan memanjatkan Puji Syukur kehadiran Allah, kami menyampaikan tuntunan praktis 5 Tahap Menjadi Dashboard Admin Untuk Kemudahan Programmer Dengan ReactJS dan TailwindCSS ini kepada para pembaca yang terhormat. Karena hanya dengan karuniaNya kami dapat menyelesaikan buku ini. Buku ini membahas langkah demi langkah bahasa pemrograman Dashboard, dengan disertai contoh-contoh untuk memberikan petunjuk dan kemudahan bagi pembaca yang mempelajari Bahasa ReactJS dan TailwindCSS. Dengan dibuatnya buku ini, penulis berharap bahwa pembuatan program aplikasi dengan bahasa ReactJS dan TailwindCSS tidaklah menjadi sesuatu yang sulit.

Tidak semua instruksi yang ada di Dashboard ini dibahas dalam buku, hanya instruksi yang erat hubungannya dengan pembuatan program aplikasi yang penulis bahas. Instruksi-instruksi lain dapat dilihat dari beberapa buku acuan yang penulis gunakan dalam penyusunan buku ini.

Dengan segala kerendahan hati, penulis menyadari bahwa buku ini masih jauh dari sempurna, karena keterbatasan kemampuan pengalaman penulis. Karena itu penulis ucapkan banyak Terima kasih atas saran dan kritik yang positif untuk perbaikan serta penyempurnaan buku ini yang sangat penulis harapkan.

Buku ini disajikan dalam setiap bab yang berisi langkah-langkah untuk pembaca pemula. Setiap bab dibuka dengan definisi dan kerangka kerja. Pada bagian selanjutnya diterangkan cara membuat kode program beserta penjelasan setiap baris kode program tersebut. Pada bagian akhir disertakan kode program untuk latihan dan evaluasi pembaca. Setiap chapter atau bab yang disajikan dalam buku ini disertakan juga kode program yang bisa diakses melalui link: <https://github.com/bukped/5-Tahap-Membuat-Dashboard-Admin-Untuk-Kemudahan-Programmer-Dengan-ReactJS-dan-TailwindCS>

# DAFTAR ISI

<b>KATA PENGANTAR</b>	<b>i</b>
<b>PRAKATA</b>	<b>i</b>
<b>DAFTAR ISI</b>	<b>ii</b>
<b>DAFTAR GAMBAR</b>	<b>iv</b>
<b>DAFTAR TABEL</b>	<b>vi</b>
<b>BAB 1 PENGENALAN DATA KOLEKSI MUSEUM</b>	<b>1</b>
A.    PENDAHULUAN	1
B.    IDENTIFIKASI MASALAH DATA KOLEKSI MUSEUM	2
C.    TUJUAN DAN MANFAAT DATA KOLEKSI MUSEUM	2
D.    RUANG LINGKUP DATA KOLEKSI MUSEUM	3
<b>BAB 2 TEORI PENGENALAN DASHBOARD</b>	<b>4</b>
A.    DESIGN SPRINT	4
B.    REACT JS	5
C.    TAILWIND CSS	7
D.    DATA SPASIAL OPENSTREETMAP	8
E.    DASHBOARD	8
<b>BAB 3 IMPLEMENTASI 5 TAHAP DESIGN SPRINT</b>	<b>10</b>
A.    STUDI LITERATUR DAN PEMAHAMAN	10
B.    ALAT DAN BAHAN YANG DIGUNAKAN	10
C.    SPESIFIKASI DAN PENJELASAN RANCANG BANGUN ALAT	12
D.    PROSEDUR PENELITIAN	15
1.    Sprint	16
<b>BAB 4 IMPLEMENTASI DATA SPASIAL OPENSTREETMAP</b>	<b>26</b>
A.    OPENSTREETMAP	26
1.    Sumber Data	26

2. Pengumpulan Data	26
B. PENGGUNAAN OPENSTREETMAP	30
<b>BAB 5 ANALISIS DAN PERANCANGAN SISTEM</b>	<b>32</b>
A. PERANCANGAN DAN FINISHING SYSTEM	32
1. Analisis Sistem yang akan dibangun	32
B. PERANCANGAN USER INTERFACE SYSTEM	36
1. Rancangan User Interface Login	36
2. Rancangan User Interface Dashboard	39
3. Rancangan User Interface Tabel	45
4. Rancangan User Interface Crud	49
5. Rancangan User Interface OpenStreetMap	80
C. PERANCANGAN ARSITEKTUR PERANGKAT LUNAK DAN PERANGKAT KERAS SISTEM	84
1. Arsitektur Perangkat Lunak	84
2. Perangkat Keras Sistem	84
D. PEMETAAN STRUKTUR DIAGRAM USER / AKTOR SISTEM	84
<b>BAB 6 PENUTUP</b>	<b>86</b>
A. KESIMPULAN	86
<b>DAFTAR PUSTAKA</b>	<b>87</b>
<b>GLOSARIUM</b>	<b>89</b>
<b>KREDIT GAMBAR</b>	<b>93</b>
<b>INDEKS</b>	<b>94</b>
<b>TENTANG PENULIS</b>	<b>96</b>

# DAFTAR GAMBAR

Gambar 1 OpenStreetMap PETA	8
Gambar 2 Node JS	10
Gambar 3 Visual Studio Code	11
Gambar 4 React JS	11
Gambar 5 Tailwind CSS	12
Gambar 6 Instalasi Visual Studio Code	13
Gambar 7 Flowchart Tahapan Penelitian	15
Gambar 8 Pemahaman (JiraSoftware)	16
Gambar 9 Sketsa Admin Login	18
Gambar 10 Sketsa Tampilan Dashboard	18
Gambar 11 Sketsa Tampilan Tabel	19
Gambar 12 Sketsa Tampilan Crud	20
Gambar 13 Pembentukan Login	21
Gambar 14 Pembentukan Dashboard	21
Gambar 15 Pembentukan Tabel	22
Gambar 16 Pembentukan Form	22
Gambar 17 Data POST sistem	23
Gambar 18 Data GET sistem	24
Gambar 19 Data JSON api	24
Gambar 20 Data Delete sistem	25
Gambar 21 Basis Data (Back-end)	27
Gambar 22 Data Polygon	28
Gambar 23 Code Polygon	29
Gambar 24 Gambar Situs Openstreetmap	30
Gambar 25 Navigasi Peta	31
Gambar 26 Analisis Prosedur yang akan dibangun (Flowchart)	32
Gambar 27 Flowchart Algoritma yang diterapkan pada Alat	33
Gambar 28 Activity Diagram Tabel	34
Gambar 29 Activity Diagram Form Batuan, Fosil dan Sumber daya geologi	35
Gambar 30 Rancangan User Interface Login	36
Gambar 31 Rancangan User Interface Dashboard	39
Gambar 32 Rancangan User Interface Tabel	45
Gambar 33 Rancangan User Interface Crud	49

Gambar 34 Rancangan User Interface Openstreetmap	80
Gambar 35 Arsitektur Perangkat Lunak	84
Gambar 36 Pemetaan Struktur Diagram User / Aktor Sistem	85



# DAFTAR TABEL

Tabel 1 Pertanyaan Bercabang (Diverage)	17
Tabel 2 Perangkat Keras Sistem	84

# BAB 1

## PENGENALAN DATA KOLEKSI MUSEUM

---

### A. PENDAHULUAN

Perkembangan era digitalisasi membuat masyarakat kita yaitu masyarakat Indonesia merasakan dampak dari berbagai aspek dari adanya kemajuan teknologi. Kemajuan teknologi yang akan dibahas pada topik ini merupakan perkembangan teknologi dibidang internet.

Dengan adanya pernyataan tersebut, maka tak sedikit pula perusahaan di Indonesia yang berlomba-lomba dalam berinovasi di bidang aplikasi, salah satunya adalah Data Koleksi Museum. Data Koleksi Museum merupakan pengelolaan atau penyimpanan barang sejarah yang bergerak dalam pengelolaan, penelitian, pengembangan, konservasi, peragaan, dan penyebaran informasi koleksi geologi (BADAN GEOLOGI - KEMENTERIAN ENERGI , n.d.).

Berbagai koleksi geologi dan pertambangan terlengkap di Indonesia dapat dijumpai di sana. Museum yang semula merupakan laboratorium, memiliki ragam koleksi batuan, mineral, meteorit, fosil dan artefak yang telah dikumpulkan sejak tahun 1850-an. Gedung tersebut pada awalnya bernama Geologisch Laboratorium yang kemudian juga disebut Geologisch Museum. Museum Geologi pertama kali diresmikan pada tanggal 16 Mei 1929 Masehi. Awalnya museum ini berfungsi sebagai tempat penyimpanan koleksi hasil penyelidikan geologi yang dilakukan oleh para ahli yang dikomandoi oleh pemerintah Belanda pada zaman dahulu. Dalam museum ini tersimpan dan dikelola materi-materi geologi yang berlimpah, seperti fosil, batuan, dan mineral (SY, A, & Riana, 2011).

Data Koleksi Museum adalah museum geologi satu-satunya yang dimiliki Indonesia. Koleksi mereka yang berupa benda-benda yang merupakan material geologi adalah aset yang harus dipelihara dan dijaga agar

kebermanfaatannya dapat dirasakan untuk waktu yang lama hingga ke generasi-generasi selanjutnya. Berbagai cara bisa dilakukan oleh pihak museum agar koleksinya awet salah satunya adalah dengan melakukan konservasi. Penelitian ini penting untuk dilakukan karena penulis belum menemukan penelitian yang membahas tentang konservasi preventif koleksi Data Koleksi Museum (Indriani, Siti Khadijah, CMS, & Khoerunnisa, 2022).

Penelitian ini bertujuan untuk mendeskripsikan tindakan preservasi preventif untuk melindungi koleksi yang dilakukan di Museum Geologi. Penelitian ini menggunakan pendekatan kualitatif berdasarkan data hasil dari wawancara, observasi, dan studi pustaka. Hasil dari penelitian ini menyimpulkan bahwa telah dilakukan tindakan preservasi preventif seperti melalui pemantauan kebersihan koleksi, pengawasan pengaturan suhu, penanganan koleksi, serta perlindungan koleksi. Meski begitu dibutuhkan preservasi yang lebih intensif sehingga dapat mengurangi potensi adanya faktor penyebab kerusakan koleksi di Museum Geologi (Tiaranisa, Siti Khadijah, CMS, & Khoerunnisa, 2022).

Data Koleksi Museum sendiri terbentuk karena adanya urgensi tentang permasalahan data yang ada di Indonesia, permasalahan yang diangkat adalah banyak sistem yang masih merasa kesulitan dalam melakukan berbagai macam penginputan dan penyimpanan.

## **B. IDENTIFIKASI MASALAH DATA KOLEKSI MUSEUM**

Adapun identifikasi masalah pada penelitian ini sebagai berikut,

- a) Penginputan yang bersumber dari Data Koleksi Museum Rawan terhadap human error
- b) Karyawan masih menggunakan penginputan Microsoft Excell
- c) Titik lokasi atau tempat benda bersejarah masih tidak akurat

## **C. TUJUAN DAN MANFAAT DATA KOLEKSI MUSEUM**

Adapun tujuan pada penelitian ini sebagai berikut,

- a) Meminimalisir terhadap human error penginputan data
- b) Dapat Memudahkan karyawan dalam penginputan data penyimpanan dan pengerjaan lebih cepat dengan Metode Design Sprint
- c) Dapat memperjelas titik akurat lokasi penemuan dengan Metode Data Spasial OpenStreetMap

Adapun manfaat pada penelitian ini sebagai berikut,

- a) Merkomendasikan dengan performa yang lebih baik dalam data penginputan.
- b) Merekomendasikan performa lokasi dengan jelas dari Metode Data Spasial OpenStreetMap

#### **D. RUANG LINGKUP DATA KOLEKSI MUSEUM**

Adapun ruang lingkup pada penelitian ini sebagai berikut,

- a) Penginputan yang bersumber dari Data Koleksi Museum
- b) Bahasa pemrograman yang digunakan menggunakan React.js
- c) Framework yang digunakan UI ( User Interface ) TailwindCSS

# BAB 2

## TEORI PENGENALAN DASHBOARD

---

### A. DESIGN SPRINT

Design Sprint merupakan metodologi desain yang dikembangkan oleh Google untuk menyelesaikan masalah dengan melibatkan pengguna melalui perancangan, pembuatan prototype, dan pengujian ide dengan cepat. Design Sprint juga dapat diartikan sebagai metode untuk membuat suatu produk yang baru. Pada penelitian ini akan menghasilkan suatu tampilan user interface dari aplikasi dashboard, sehingga metode design sprint terhadap penelitian ini dapat berarti membuat suatu solusi user interface pada kasus aplikasi dashboard. Terdapat lima tahapan Design Sprint (Ramadan, Az-Zahra, & Rokhmawati, 2019):

1. Fase di hari Senin : Menentukan tujuan Tahap awal fase Sprint di hari Senin adalah
  - a) Menentukan tujuan jangka panjang, dengan cara membuat daftar pertanyaan mengenai apa yang akan dicapai dan bagaimana cara mencapai tujuan tersebut
  - b) Memetakan tantangan yang akan dituju, dengan cara menuliskan siapa saja actor dan tujuan yang akan diselesaikan
  - c) Bertanya kepada ahli, dengan cara melakukan metode How Might We, metode ini memungkinkan untuk menggali solusi yang lebih spesifik.
  - d) Menentukan target
2. Fase di hari Selasa : Mencari Solusi Tahap Sprint di hari Selasa yaitu:
  - a) Mencari inspirasi kemudian menggabungkan dan mengembangkannya, dengan cara Lightning Demo. Lightning Demo merupakan metode yang digunakan untuk mengumpulkan berbagai ide dan menyatukan ide tersebut
  - b) Membuat sketsa wireframe produk, dengan melakukan metode Crazy 8 dengan anggota tim, kemudian menjelaskan detail solusi tersebut kepada masing-masing anggota

3. Fase di hari Rabu : Memutuskan Solusi Tahap Sprint di hari Rabu, yaitu:
  - Memutuskan solusi yang akan dijadikan atau dibuat menjadi prototype dengan cara:
    - a) Present Sketch Solutions, Menempelkan sketsa
    - b) Heatmap Voting, Melihat solusi sketsa yang ada dan menandai yang dianggap menarik
    - c) Quick Criticism, Membahas solusi setiap sketsa
    - d) Note and Vote, Setiap orang memilih satu solusi dengan cara menempelkan note atau alat bantu untuk memilih pada sketsa yang ada
    - e) Rumble or All in One, Decision Maker mengambil keputusan dengan cara menilai sketsa mana yang mempunyai note terbanyak
  - Membuat Storyboard dari sketsa
4. Fase di hari Kamis : Membuat Prototype Tahap Sprint di hari Kamis, yaitu:
  - Membuat prototype berdasarkan Storyboard yang telah dibuat, dalam pembuatan prototype ini peneliti menggunakan tools Adobe XD dan Invision
5. Fase di hari Jumat : Melakukan Validasi Tahap Sprint di hari Jumat, yaitu:
  - Melakukan validasi dengan menggunakan metode Stakeholder Review dan melakukan pengujian dengan menggunakan metode usability testing kepada stakeholder dikarenakan aplikasi ini harus sesuai dengan permintaan stakeholder sebelum diluncurkan kepada user.

## B. REACT JS

React.js digunakan sebagai pendukung web framework. Hal ini senada dengan React.js adalah salah satu web framework paling populer di dunia Node.js, Dokumentasinya yang lengkap dan penggunaannya yang cukup mudah, dapat membuat kita mengembangkan berbagai produk seperti aplikasi web ataupun RESTful API dan dapat digunakan menjadi pijakan untuk membangun web framework yang lebih kompleks. Menurut penulis React adalah front-end library yang dikembangkan oleh Facebook. React digunakan untuk menangani lapisan tampilan untuk web dan aplikasi seluler. ReactJS memungkinkan pengguna atau pengembang untuk membuat komponen UI yang dapat digunakan kembali. Saat ini salah satu JavaScript libraries paling populer dan memiliki fondasi keamanan yang kuat dan memiliki komunitas besar saat ini (Fathoni, 2018).

Pemanfaatan teknologi dalam pengembangan perangkat lunak memberi kesempatan kepada peneliti untuk membangun aplikasi dan mengetahui efektifitasnya dalam proses pembuatan content management system (CMS) yang bersifat open source. Selain itu, dengan meningkatnya perkembangan teknologi khususnya pada pengembangan perangkat lunak, memicu penelitian mengenai layanan dan aplikasi untuk Access Open Journal System (AOJS) dengan berbagai fitur layanan manajemen penerbitan jurnal (Raharjo, 2013).

Pengembangan journal directory juga dilakukan oleh Raharjo. Dalam hal ini, Raharjo membangun journal directory yang dibangun dengan Drupal. Drupal adalah Content Management Framework (CMF) yang memiliki banyak modul yang bisa Digunakan untuk mengembangkan banyak jenis di situs web. Customizable, fleksibel, dapat diperluas secara umum dan banyak alasan untuk menggunakan Drupal sebagai mesin untuk membangun sebuah website, Mengembangkan journal directory dan isi artikelnya menggunakan website Drupal bisa dilakukan dengan mudah. Namun, karena journal directory yang dibangun dengan CMF Drupal, maka manajemen penerbitan secara elektronik tidak dimiliki dan hanya menfokuskan pada journal directory (Raharjo, 2013).

Selain itu, penelitian juga dilakukan oleh Borchard, et al. Penelitian ini bertujuan untuk menguji sistem informasi Public Knowledge Project (PKP) Open Journal Systems (OJS) untuk aksesibilitas keseluruhan web dengan hukum Federal Electronic and Information Technology Accessibility and Compliance pada CSU Northridge. Pada penelitiannya, hanya membahas kebutuhan penting untuk desain perangkat lunak yang mudah diakses. Dalam kasus CSUN, salah satu mitra kampus akan dilayani dengan lebih baik dengan meningkatkan aksesibilitas web jurnal akses online penulis (Borchard, Biondo, Kutay, Morck, & Weiss, 2015).

Peterson., et al dalam jurnal yang berjudul Open Access and the Author-Pays Problem: Assuring Access for Readers and Authors in a Global Community of Scholars melakukan penelitian pada minat yang kritis terhadap komunikasi ilmiah, produksi penelitian, dan ukuran kualitas yang tepat dan obyektif. Dalam jangka menengah, dengan bertambahnya Pilihan jurnal OA, harus dimungkinkan untuk memiliki dana tersebut untuk mendukung penerbitan akademik dari dalam dunia akademis. Pilihan seperti itu sulit bagi penulis dan peneliti dalam mempresentasikan karya dan dari sisi konten yang dibutuhkan akademisi dan siswa harus menyiapkan dana jutaan dolar kepada penerbit komersial per tahun, pada sisi lain sangat dibutuhkan dalam reinventionsistematis dan etis dari

sistem komunikasi ilmiah. Pada penelitiannya, hanya memuat informasi mengenai solusi dalam meningkatkan motivasi pada peningkatan publikasi dan minat untuk mendukung kualitas akademik serta penulis maupun pembaca (Peterson, 2013).

### **C. TAILWIND CSS**

Tailwind CSS dikembangkan dengan tujuan untuk mempercepat proses prototyping halaman web agar sesuai kebutuhan (custom). Tailwind CSS mencoba mengambil pendekatan lain dalam pengembangan web, yang selama ini didominasi Bootstrap, agar tidak ada lagi istilah “tampilan sejuta umat” karena kemiripan antara web yang satu dengan yang lain (karena sama-sama memakai tema default Bootstrap, minim kustomisasi).

Alasan mengapa menggunakan tailwind pada kasus penelitian kali ini, dikarenakan ingin mengimplementasikan utility classes. Apa itu utility classes, utility classes merupakan sebuah class didalam css yang berasal dari framework yang telah dideklarasikan berupa properti - proterti yang mengatur sebuah layout.

Contohnya adalah untuk pengaturan padding dideklarasikan hanya berisi properti padding saja, mengapa demikian dikarenakan agar pengguna dapat leluasa mengatur tampilan dari website tanpa terpaku oleh aturan - aturan yang dibuat oleh framework yang mendefinisikan lebih dari satu property css didalam 1 class (Fataha, 2022).

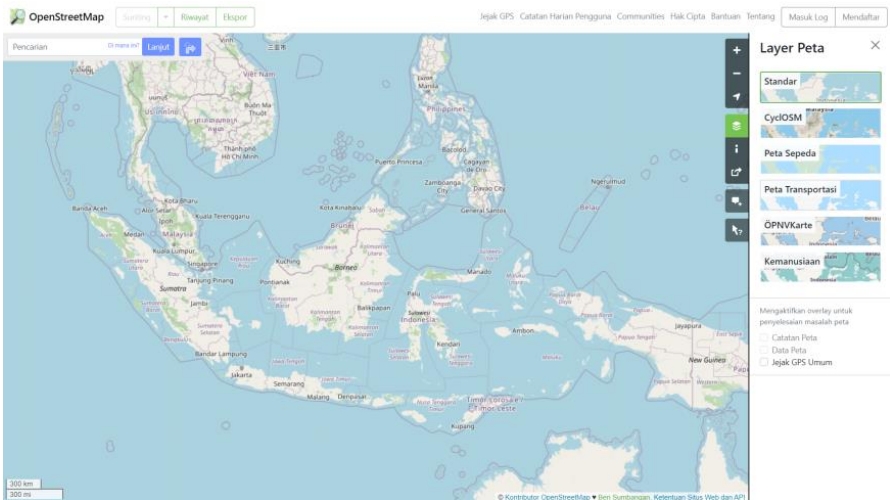
Point Penting Dari Tailwind CSS :

- a) Tailwind CSS menyediakan banyak sekali CSS class yang siap pakai untuk
- b) membangun UI sesuai kebutuhan.
- c) Meminimalisir Penulisan Custom CSS.
- d) Responsive karena setiap utilitas Tailwind memiliki jenis-jenis responsive yang
- e) mempermudah untuk membangun interface responsive tanpa perlu menggunakan
- f) CSS khusus.
- g) Component-friendly meskipun kita dapat melakukan banyak hal hanya dengan
- h) menggunakan kelas utilitas, seiring dengan berkembangnya sebuah proyek, akan
- i) berguna untuk mengkodekan pola yang umum menjadi abstraksi tingkat tinggi.



#### D. DATA SPASIAL OPENSTREETMAP

Banyak sekali contoh penggunaan OpenStreetMap dalam berbagai bidang. Dengan OpenStreetMap, Anda dapat melengkapi data peta dan membuat peta di lokasi yang sudah familiar dengan Anda misalnya dalam skala desa. Anda dapat memetakan secara detail karena setiap pengguna dimungkinkan untuk melakukan survei secara mandiri pada wilayah yang akan dipetakan. Berikut ini adalah contoh gambar peta Openstreetmap (Yusuf & Supriyadi, 2022);



Gambar 1 OpenStreetMap PETA

#### E. DASHBOARD

Penelitian yang dilakukan oleh (Dini Nurmalasari, 2015) membahas tentang Informational Dashboard adalah sebagai monitoring sistem drainase secara real time. Informational dashboard merupakan alat yang digunakan untuk menyajikan informasi dalam bentuk diagram, indikator visual, ataupun grafik berupa informasi yang ringkas dinamis serta relevan. Tujuan penggunaan informational dashboard diantaranya adalah untuk mempercepat proses pengambilan keputusan, mengukur kinerja, memonitoring proses yang sedang berjalan dan memprediksi kondisi dimasa mendatang. Proses pembangunan dashboard dengan cara memonitoring sistem drainase secara real-time, metode yang digunakan oleh penulis menggunakan metode identifikasi kebutuhan diantaranya memperhatikan kebutuhan fungsional dan kebutuhan non fungsioanal selain itu menggunakan metode analisa dan perancangan sehingga menghasilkan sistem monitoring drainase yang dibangun berfungsi untuk memantau kondisi drainase dibeberapa titik secara real time dan kontinu

dengan memanfaatkan jaringan sensor nirkabel dan juga beberapa node sensor yang disebar di beberapa titik yang akan di komunikasikan dengan mengirim data server berupa level air pada drainase dengan adanya sistem yang otomatis diharapkan dapat memantau kondisi drainase secara real time dan kontinu serta dapat menyimpan data dengan baik.

# BAB 3

## IMPLEMENTASI

### 5 TAHAP DESIGN SPRINT

---

#### A. STUDI LITERATUR DAN PEMAHAMAN

Pada tahap ini penulis mencari dan mempelajari referensi dari berbagai open source seperti artikel ilmiah, contoh-contoh dataset dan program, serta jurnal nasional/internasional. Adapun topik yang akan dijelaskan ialah; pengembangan, front-end, Dashboard Admin, ReactJS, Metode Design Sprint dan Metode Data Spasial OpenStreetMap Programming untuk pengembangan.

#### B. ALAT DAN BAHAN YANG DIGUNAKAN

Bahan dan alat yang digunakan untuk menjalankan dan pembuatan Dashboard ini dengan:

##### a) Node JS

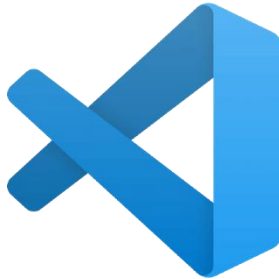


*Gambar 2 Node JS*

Sebagai runtime JavaScript berbasis peristiwa asinkron, Node.js dirancang untuk membangun aplikasi jaringan yang dapat diskalakan. Dalam contoh "halo dunia" berikut, banyak koneksi dapat ditangani secara

bersamaan. Pada setiap koneksi, callback diaktifkan, tetapi jika tidak ada pekerjaan yang harus diselesaikan.

## **b) Visual Studio Code**



*Gambar 3 Visual Studio Code*

Mempelajari kode itu menakutkan, jadi siapkan diri Anda untuk sukses dengan alat yang dibuat untuk Anda. Visual Studio Code adalah editor pengkodean gratis yang membantu Anda memulai pengkodean dengan cepat. Gunakan untuk membuat kode dalam bahasa pemrograman apa pun, tanpa berpindah editor. Visual Studio Code memiliki dukungan untuk banyak bahasa, termasuk Python, Java, C++, JavaScript, dan lainnya.

## **c) React JS**



*Gambar 4 React JS*

React telah dirancang sejak awal untuk adopsi bertahap, dan Anda dapat menggunakan React sesedikit atau sebanyak yang Anda butuhkan. Apakah Anda ingin merasakan React, menambahkan beberapa interaktivitas ke halaman HTML sederhana, atau memulai aplikasi bertenaga React yang kompleks, tautan di bagian ini akan membantu Anda memulai.

## **d) Tailwind CSS**



*Gambar 5 Tailwind CSS*

Tailwind CSS bekerja dengan memindai semua file HTML, komponen JavaScript, dan template lain untuk nama kelas, menghasilkan gaya yang sesuai, lalu menuliskannya ke file CSS statis. Cepat, fleksibel, dan andal — tanpa runtime.

## **C. SPESIFIKASI DAN PENJELASAN RANCANG BANGUN ALAT**

### **a) Node JS**

Setelah kita menginstal Node.js, mari kita buat server web pertama kita. Buat file bernama app.js yang berisi konten berikut:

```
const http = require('http');

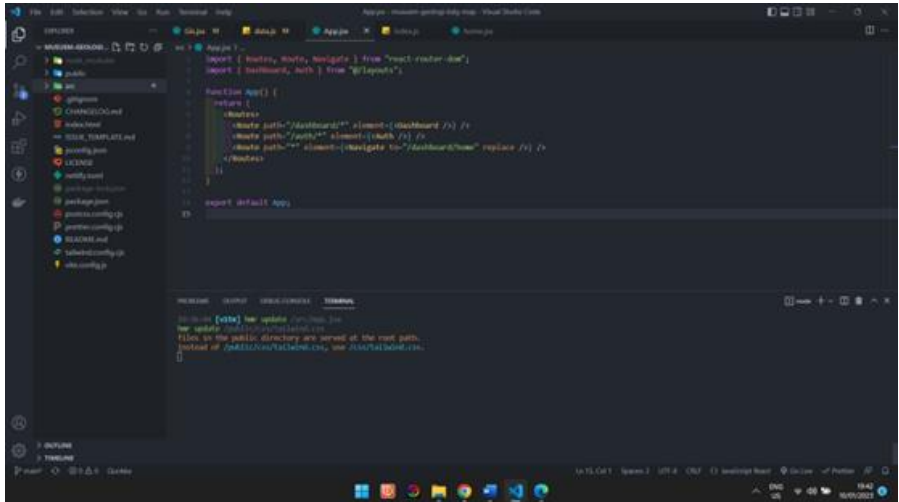
const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World');
});

server.listen(port, hostname, () => {
  console.log(`Server running at
http://${hostname}:${port}/`);
});
```

### **b) Visual Studio Code**

Setelah kita menginstal VScode, mari kita buat code web pertama kita. Buat file bernama app.js yang berisi konten data file:



*Gambar 6 Instalasi Visual Studio Code*

### c) React JS

Setelah kita menginstal VScode, mari kita buat code web pertama kita. Buat file bernama my-app yang berisi konten data file:

```
npx create-react-app my-app  
cd my-app  
npm start
```

### d) Tailwind CSS

Setelah kita menginstal ReactJS, mari kita buat code web pertama kita. Buat file dengan terminal VScode yang berisi konten data file:

```
npm install -D tailwindcss  
npx tailwindcss init
```

lalu tambahkan pada path tailwind.config.js

```
/** @type {import('tailwindcss').Config} */
```

```
module.exports = {
  content: ["/src/**/*.html,js"],
  theme: {
    extend: {},
  },
  plugins: [],
}
```

Lalu tambahkan pada path src/input.css

```
@tailwind base;
@tailwind components;
@tailwind utilities;
```

Jalankan terminal VSCode

```
npx tailwindcss -i ./src/input.css -o ./dist/output.css -watch
```

Lalu tambahkan link output.css pada path src/index.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <link href="/dist/output.css" rel="stylesheet">
</head>
<body>
  <h1 class="text-3xl font-bold underline">
    Hello world!
  </h1>
```

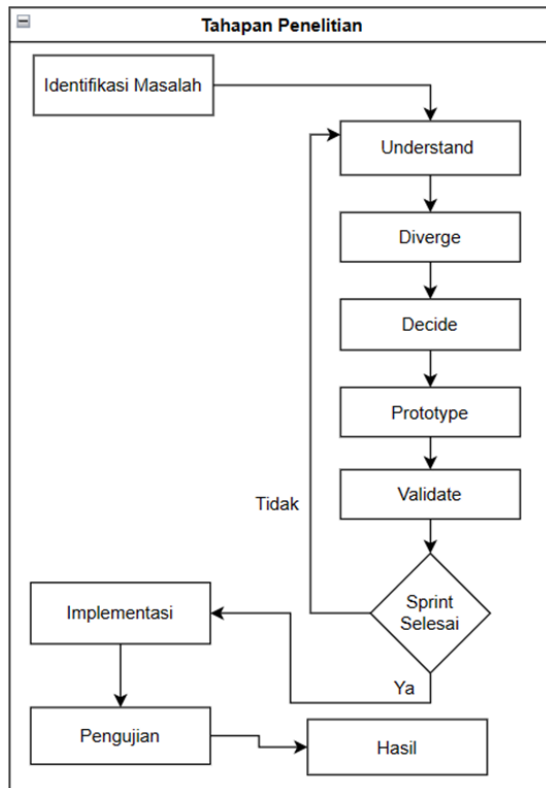
```
</body>
```

```
</html>
```

## D. PROSEDUR PENELITIAN

### a) Tahapan Penelitian

Tahapan penelitian digunakan sebagai bentuk penjabaran dari kerangka pemikiran yang dapat dilihat pada Gambar.



Gambar 7 Flowchart Tahapan Penelitian

### b) Persiapan

Tahap persiapan digunakan untuk menentukan pembahasan, tim dan sarana yang akan digunakan seperti pembahasan yang akan dilakukan terkait dengan proses pengolahan data koleksi, produk hasil pengumpulan data dan pengelolaan. Tim yang ditentukan berupa peneliti dan pihak developer. Sarana yang digunakan berupa lembar kertas yang digunakan



untuk membuat rancangan ide dan sketsa serta template html untuk membentuk prototype dari hasil sketsa yang diputuskan

## 1. Sprint

Sprint merupakan rancangan desain berdasarkan bagian admin serta memperoleh hasil persetujuan pada bagian admin atau pengurus kelompok sebagai berikut:

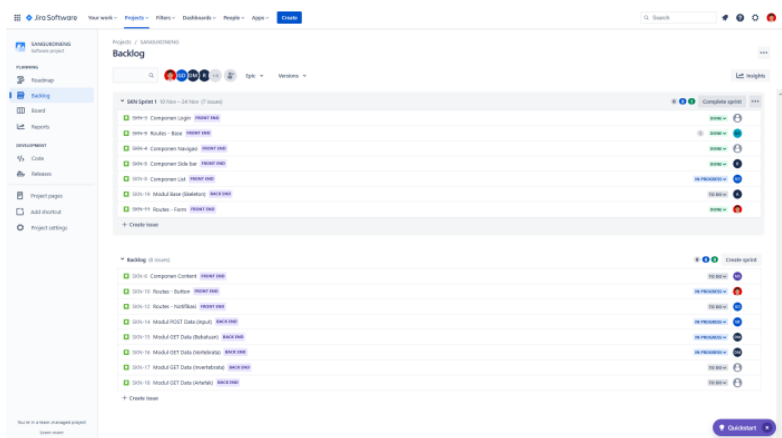
5 Tahap Design Sprint terdiri dari :

1. Understand
2. Diverge
3. Decide
4. Prototype
5. Validation

Adapun Penjelasan Masing masing tahap sebagai berikut:

## 1) Pemahaman (Understand)

Pemahaman merupakan bagian dari penentuan terhadap pengguna pada sistem yang akan dibangun dengan dilakukan diskusi untuk menentukan pengguna dan kebutuhan pada sistem berdasarkan ide yang diberikan kepada pengguna tersebut, berikut beberapa ide yang diberikan kepada bagian admin atau pengurus kelompok, hasil dokumentasi tahap understand dapat dilihat pada Gambar.



*Gambar 8 Pemahaman (JiraSoftware)*

Berdasarkan penentuan ide tersebut yang terdiri dari 7 ide atau gagasan yang diberikan telah dipilih sebanyak 4 ide seperti mengelola data kelompok, mengelola Components, mengelola Routes dan mengelola Table API.

## 2) Bercabang (Diverage)

Proses penggambaran terhadap ide yang dipilih dengan memberikan sketsa pada bagian pengguna maupun perusahaan hingga sketsa dipertahankan untuk tahap berikutnya. Proses pembuatan sketsa dilakukan dengan membuat pertanyaan terhadap dua bagian pengguna yang dapat dilihat sebagai berikut:

*Tabel 1 Pertanyaan Bercabang (Diverage)*

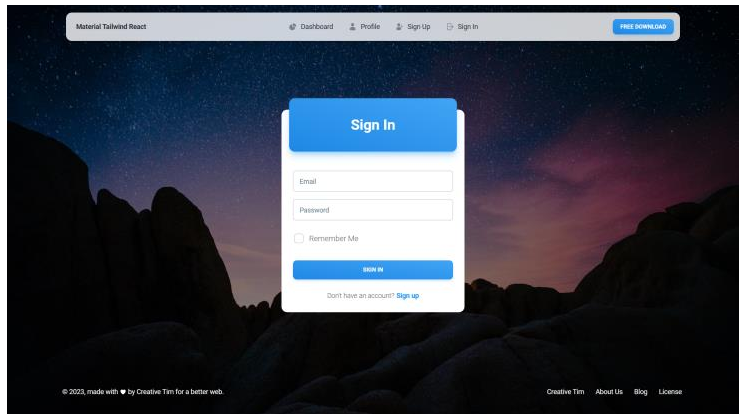
No.	Pertanyaan
1	Kebutuhan sistem seperti apa yang diinginkan oleh pihak admin ?
2	Media apa yang cocok untuk digunakan oleh admin ?
3	Fitur apa saja yang dibutuhkan pada bagian admin ?
4	Media apa saja yang mudah digunakan untuk menggambarkan sistem yang akan dibangun agar pengguna/user dapat dengan mudah memahami maksud dari rancangan ?
5	Untuk mengevaluasi hasil sistem yang di bangun , Metode apa yang cocok digunakan ?

## 3) Memutuskan (Decide)

Melakukan keputusan terhadap ide yang telah digambarkan pada tahap sebelumnya. Hasil sketsa yang telah dibentuk, dilakukan pemilihan terhadap ide yang terbaik dan dilanjutkan ketahap penggambaran sistem menggunakan sketsa interface seperti berikut:

1. Hasil dan Pembahasan
  - a) Sketsa Admin Login

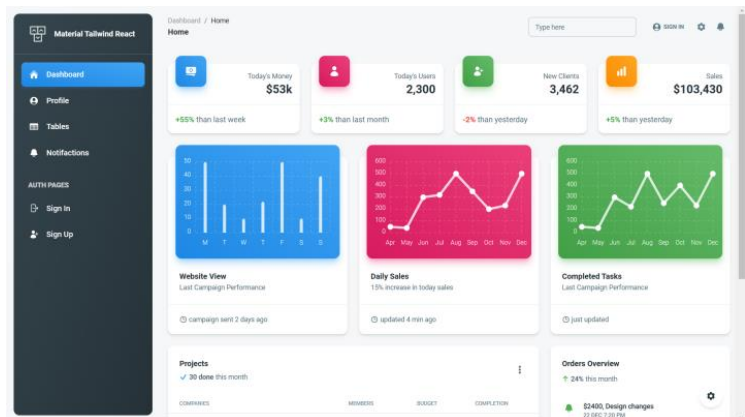
Sketsa Admin Login terdiri dari username, password dan tombol sign-in, berikut yang dapat dilihat pada gambar.



*Gambar 9 Sketsa Admin Login*

## b) Sketsa Tampilan Dashboard

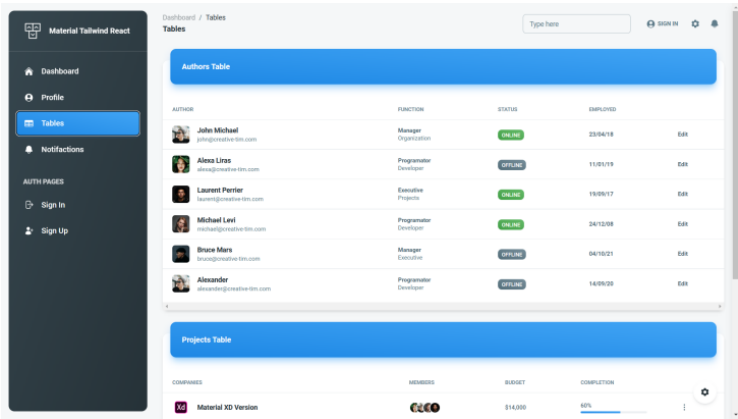
Sketsa Tampilan Dashboard terdiri dari side-bar, chart menggunakan pie chat, card jumlah item, berikutnya dapat dilihat pada gambar



*Gambar 10 Sketsa Tampilan Dashboard*

## c) Sketsa Tampilan Tabel

Sketsa Tampilan Tabel membuat data seperti No, No Register, No Inventaris, NUP BMN, Nama Koleksi, Kode Koleksi Awal, Lokasi Penyimpanan, Keterangan. Berikut dapat dilihat pada gambar.



Gambar 11 Sketsa Tampilan Tabel

d) Sketsa Tampilan Crud

Sketsa Tampilan Crud membuat data seperti No, No Register, No Inventaris, NUP BMN, Nama Koleksi, Kode Koleksi Awal, Lokasi Penyimpanan, Keterangan. Berikut dapat dilihat pada gambar

Dashboard / Batuan

Search: Type here

SIGN IN

### Batuan

Kategori BMN \*

NUP BMN

Tipe BMN

No Awal

Satuan

Kelompok Koleksi

Jenis Koleksi \*

Sub Jenis Koleksi

Settings icon

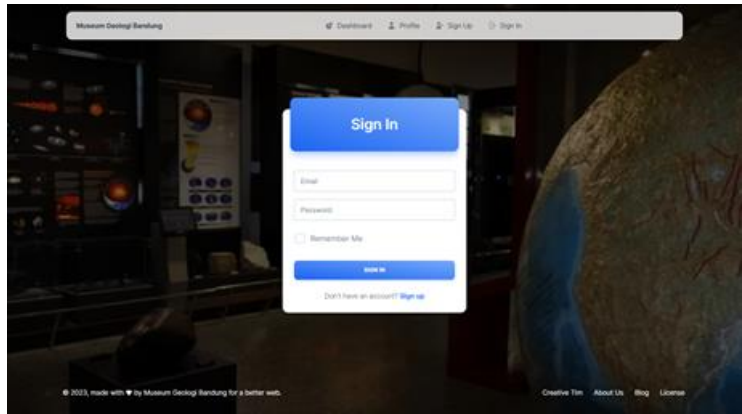
*Gambar 12 Sketsa Tampilan Crud*

#### 4) Pembentukan (Prototype)

Proses pembentukan atau penerapan dari ide yang terpilih dan fokus dalam pembangunan sistem menggunakan framework codeigniter, berikut merupakan bagian perancangan prototype menggunakan konsep berbasis website.

##### a) Pembentukan Login

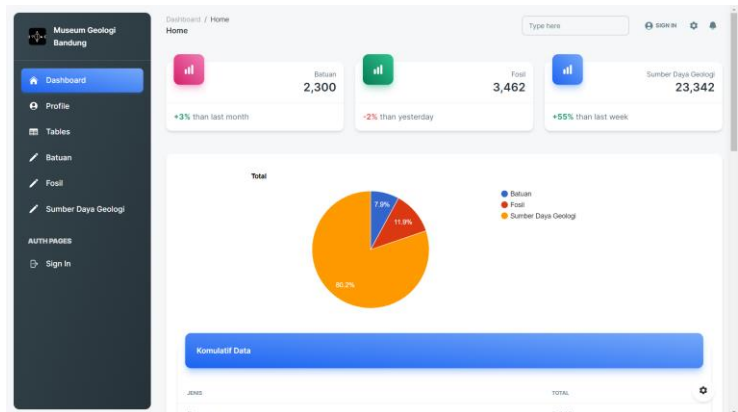
Berikut ini adalah prototype dari halaman login yang di tampilkan pada gambar.



Gambar 13 Pembentukan Login

## b) Pembentukan Dashboard

Berikut ini adalah prototype dari halaman dashboard yang di tampilkan pada gambar.



Gambar 14 Pembentukan Dashboard

## c) Pembentukan Tabel

Berikut ini adalah prototype dari halaman Tabel yang di tampilkan pada gambar

NUP BMN	NAMA KOLEKSI	KODE KOLEKSI AWAL	LOKASI PENYIMPANAN	KETERANGAN	DETAIL	EDIT
6.06.01.05.005	Batugamping	31070101		Batuan Sedimen (Klastik)	<a href="#">CIRIK</a> <a href="#">EDIT</a> <a href="#">RELATIF</a>	

NO	NO REGISTER	NO INVENTARIS	NUP BMN	NAMA KOLEKSI	KODE KOLEKSI AWAL	LOKASI PENYIMPANAN	KETERANGAN
638DA8ADB4540CF8B628892	MGB-00000012	FIM00000012		Clarus (Clarus) (Ballingensis (Martin 1895)			Label Fossil
638DA8ADB4540CF8B62889	MGB-00000003	FIM00000003		Trigonostoma (Scaptila) arjehense Cossigny			Label Fossil

*Gambar 15 Pembentukan Tabel*

#### d) Pembentukan Form

Berikut ini adalah prototype dari halaman Form yang di tampilkan pada gambar

*Gambar 16 Pembentukan Form*

## 5) Validasi (Validation)

Setelah prototype dibuat, maka masuk pada tahap validate, di mana prototype yang telah dikembangkan sebelumnya akan di review dengan menggunakan usability testing bersama stakeholder berdasarkan aspek parameter yang akan dicapai yaitu efisiensi, efektivitas, dan kepuasan pengguna terhadap kebutuhan dengan fungsi utama dan alternatif sistem. Proses sprint ini akan

dibagi menjadi empat iterasi design sprint berdasarkan keputusan yang diambil mengenai solusi yang akan dikerjakan terlebih dahulu. Tahapan terakhir adalah memberikan kesimpulan dan saran dari hasil analisis pengujian yang telah dilakukan.

## 6) Implementasi Sistem

### Data POST sistem

Berikut ini adalah code untuk POST untuk mengirimkan data API dari client to server.

```
1  const url = "https://sbc-sebatcabut.herokuapp.com";
2  const [formData, setFormData] = useState({
3    id: "539ac7f63153ff7083d5b797",
4    no_register: "MGB-00002938",
5    no_inventaris: "05f00000001",
6    kode_bmn: "G-06.01.05.005",
7    nup_bmn: "g",
8    merk_bmn: "Batuan",
9    satuan: "Buah",
10   kelompok_koleksi: "Batuan",
11   jenis_koleksi: "Sedimen",
12   sub_jenis_koleksi: "-",
13   kode_jenis_koleksi: "BSE",
14   ruang_simpan: "Gudang 3",
15   lokasi_simpan: "31070101",
16   kondisi: "B/Baik",
17   nama_koleksi: "Batugamping",
18   keterangan: "Batuan Sedimen (Klastik)",
19   nama_formasi: "Bojongmanik",
20   lokasi_temuan: "Desa Ujungjaya, Kec. Sumur, Kab. Pandeglang",
21   koordinat: "Lat/Lon: 6 49' 31.1268\" S, 105 28' 35.5764\" E",
22   pulau: "Jawa",
23   peta: "Geologi",
24   lembar_peta: "Ujungkulon",
25   skala: "1:100000",
26   cara_perolehan: "Penyelidikan geologi",
27   tahun_perolehan: "2012",
28   kolektor: "Mamang Garok",
29   kepemilikan: "Museum Geologi Bandung",
30   operator: "Administrator",
31   tanggal_dicatat: "12/12/2022 6:57",
32   nilai_perolehan: "-",
33   nilai_buku: "-",
34   foto: "test.jpg",
35   foto_2: "testB.jpg",
36   foto_3: "testC.jpg"
37 });
38
39 const [errors, setErrors] = useState({});
40
41 const handleChange = event => {
42   setFormData({ ...formData, [event.target.name]: event.target.value });
43   setErrors({ ...errors, [event.target.name]: '' });
44 }
45
46 const validate = () => {
47   const newErrors = {};
48   if (!formData.nup_bmn) {
49     newErrors.nup_bmn = "NUP BMN is required", alert("NUP BMN is required");
50   }
51   return newErrors;
52 }
53
54 const handleSubmit = event => {
55   event.preventDefault();
56   const newErrors = validate();
57   if (Object.keys(newErrors).length > 0) {
58     setErrors(newErrors);
59   } else {
60     axios.post(url + '/batuan', formData)
61       .then(res => {
62         console.log(res);
63         console.log(res.data);
64       })
65       .catch(error => {
66         console.log(error);
67       });
68   }
69 }
```

Gambar 17 Data POST sistem



## Data GET sistem

Berikut ini code GET untuk menampilkan data tabel dengan menggunakan useState.

*Gambar 18 Data GET sistem*

Berikut ini adalah data GET api yang berformat JSON

[illegible]

*Gambar 19 Data JSON api*

## Data Delete sistem

Berikut ini adalah code untuk mendelete data berdasarkan id API.



*Gambar 20 Data Delete sistem*

# **BAB 4**

## **IMPLEMENTASI**

### **DATA SPASIAL OPENSTREETMAP**

---

#### **A. OPENSTREETMAP**

##### **1. Sumber Data**

Sumber data yang digunakan dalam penelitian ini adalah menggunakan data primer dan data sekunder, adapun penjelasannya adalah sebagai berikut:

##### **1) Data Primer**

Data primer merupakan data yang diperoleh secara langsung, biasanya untuk mendapatkan data primer dilakukan melalui wawancara. Wawancara dilakukan penulis dengan para Dekan Museum dan Kepala Program Studi D4 Teknik Informatika, data yang dihasilkan berupa data Data Koleksi Museum.

##### **2) Data Sekunder**

Data sekunder merupakan segala keterangan yang diperoleh dari buku – buku, jurnal dan berbagai sumber lainnya yang diperlukan untuk mendukung dalam proses pembuatan maupun hasil dari penelitian yang dilakukan penulis dalam penelitian ini.

##### **2. Pengumpulan Data**

##### **1) Observasi**

Dilakukan dengan cara mengamati kegiatan pencatatan perkuliahan yang dilakukan oleh dosen dan mahasiswa di

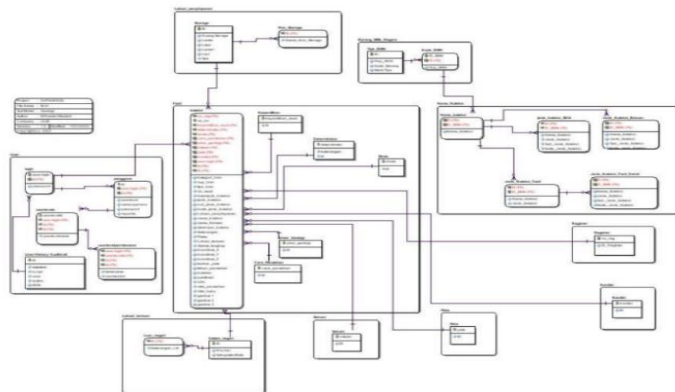
Fakultas Ilmu Komputer, hasil yang diperoleh berupa catatan. Catatan tersebut adalah berupa data – data yang diperlukan dalam penelitian ini

## 2) Wawancara

Metode wawancara adalah metode yang digunakan untuk mendapatkan data secara langsung, penulis melakukan wawancara dengan tanya jawab secara langsung kepada narasumber. Data yang diperoleh dari metode wawancara merupakan data primer. Wawancara dilakukan dengan kepala para Dosen, Wakil Dekan, Kepala Program Studi dan Kepala Tata Usaha.

## 3) Basis Data (Back-end)

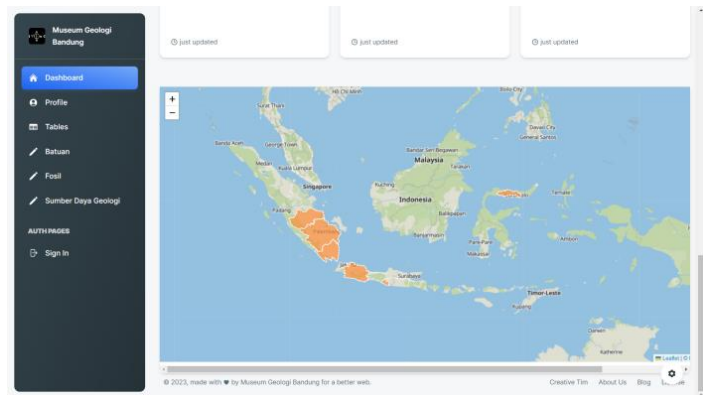
Berikut ini adalah gambar struktur basis data dari back-end.



Gambar 21 Basis Data (Back-end)

## 4) Implementasi Sistem Data Polygon

Menampilkan data Polygon dengan Openstreetmap.

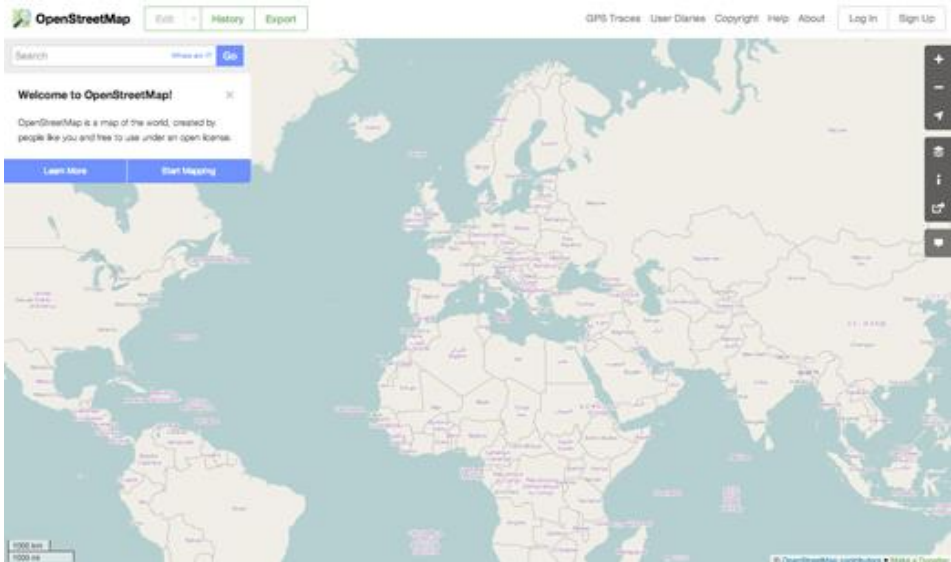


*Gambar 22 Data Polygon*



## B. PENGGUNAAN OPENSTREETMAP

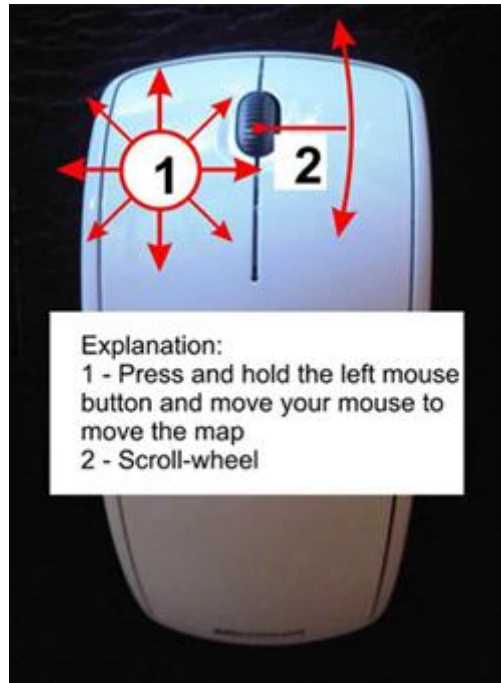
Di bagian ini kita akan belajar langkah demi langkah cara menavigasi situs web OpenStreetMap, melihat peta, dan mendaftar untuk akun pengguna. Setelah Anda memiliki nama pengguna dan kata sandi Anda sendiri, Anda akan dapat berkontribusi untuk titik pertama Anda ke peta.



*Gambar 24 Gambar Situs Openstreetmap*

- Buka browser web Anda
- Pada alamat bar diatas jendela, ketikkan link berikut dan tekan Enter: [www.openstreetmap.org](http://www.openstreetmap.org)
- Saat halaman sudah selesai memuat, Anda seharusnya melihat sesuatu seperti ini:

Hal penting yang seharusnya Anda lihat adalah peta. Geser peta dengan cara klik peta menggunakan mouse sebelah kiri, tahan tombol mouse, dan geser mouse Anda (lihat gambar dibawah).



*Gambar 25 Navigasi Peta*



# BAB 5

## ANALISIS DAN PERANCANGAN SISTEM

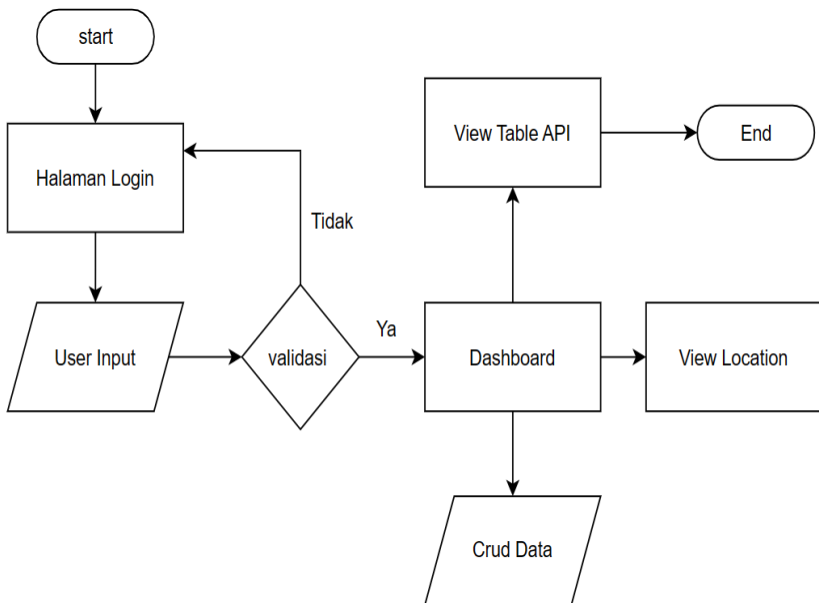
---

### A. PERANCANGAN DAN FINISHING SYSTEM

#### 1. Analisis Sistem yang akan dibangun

##### 1) Analisis Prosedur yang akan dibangun (Flowchart)

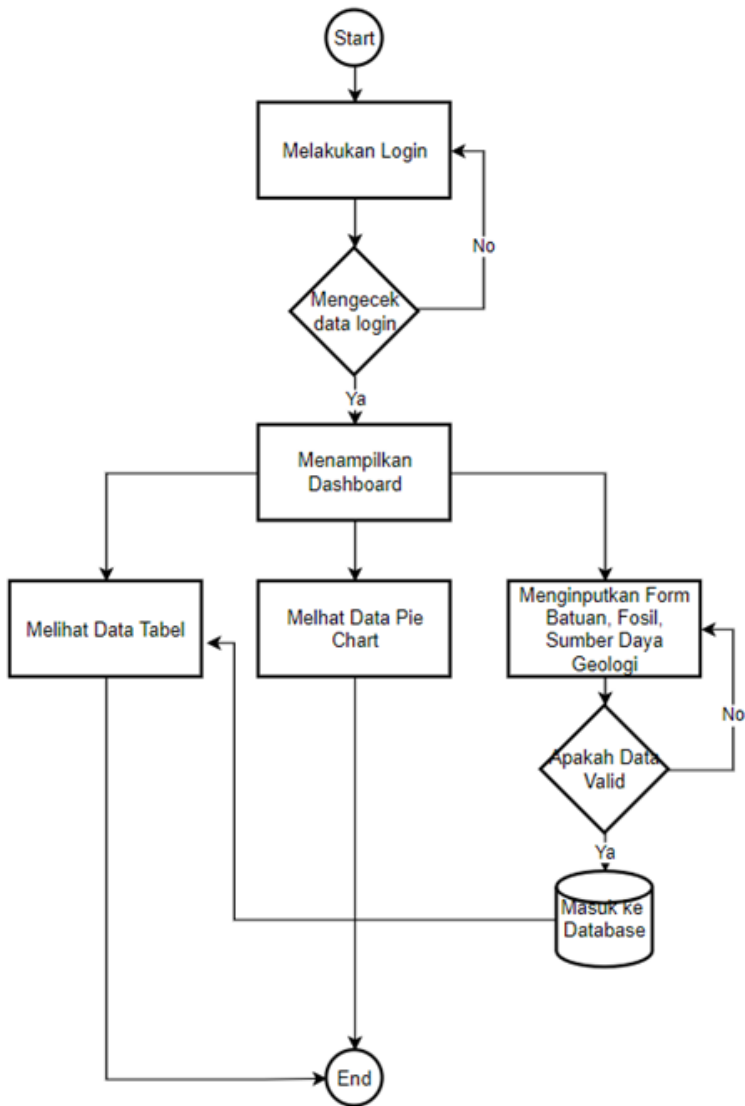
Berikut ini adalah Analisis Prosedur Flowchart Algoritma sistem yang dibangun.



*Gambar 26 Analisis Prosedur yang akan dibangun (Flowchart)*

## 2) Flowchart Algoritma yang diterapkan pada Alat

Berikut ini adalah Flowchart Algoritma sistem yang dibangun.

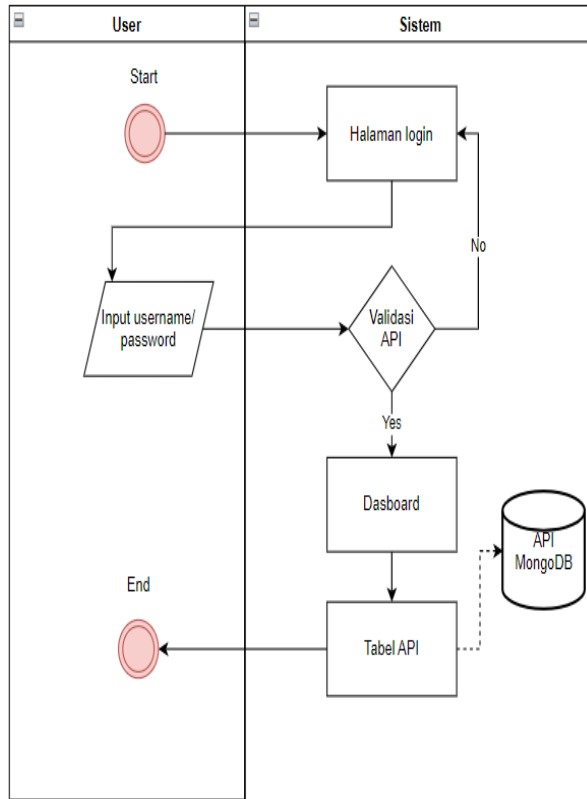


Gambar 27 Flowchart Algoritma yang diterapkan pada Alat

### 3) UML (Unified Modelling Language)

#### a) Activity Diagram Tabel

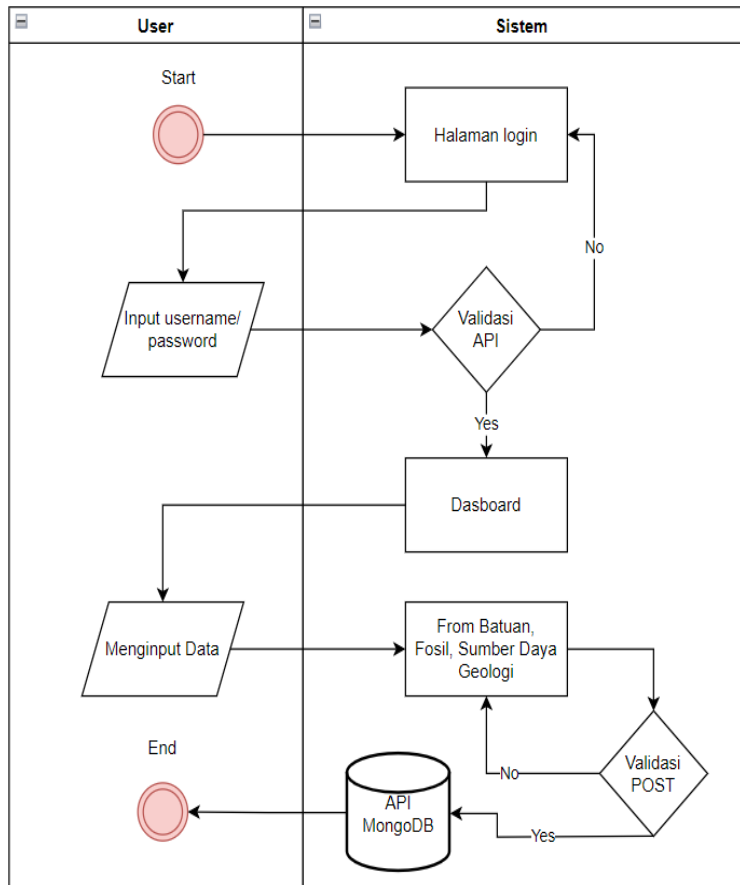
Berikut ini adalah Activity Diagram tabel yang dibangun.



*Gambar 28 Activity Diagram Tabel*

b) Activity Diagram Form Batuan, Fosil dan Sumber Daya Geologi

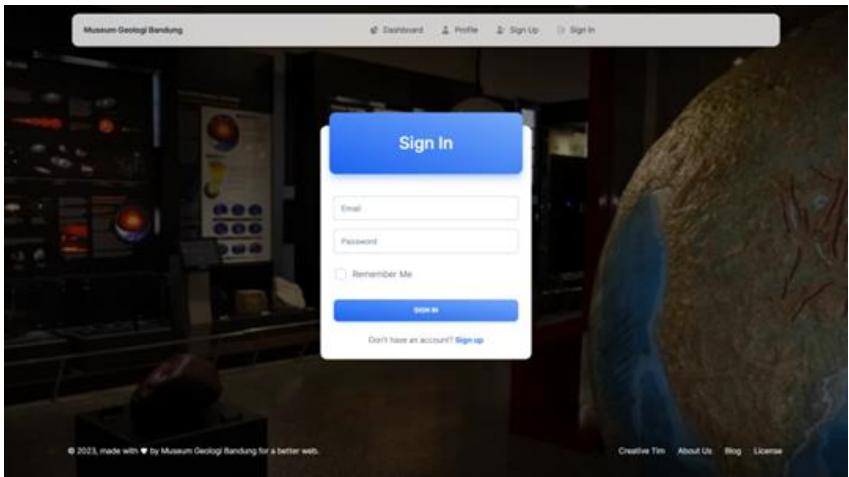
Berikut ini adalah Activity Diagram form yang dibangun.



*Gambar 29 Activity Diagram Form Batuan, Fosil dan Sumber daya geologi*

## B. PERANCANGAN USER INTERFACE SYSTEM

### 1. Rancangan User Interface Login



*Gambar 30 Rancangan User Interface Login*

#### Mengimport kebutuhan components

```
import { Link } from "react-router-dom";
import {
  Card,
  CardHeader,
  CardBody,
  CardFooter,
  Input,
  Checkbox,
  Button,
  Typography,
} from "@material-tailwind/react";
import { useJwt } from "react-jwt";
import { useEffect, useState } from "react";
```

## Menampilkan tampilan login

```


<div className="absolute inset-0 z-0 h-full w-
full bg-black/50" />

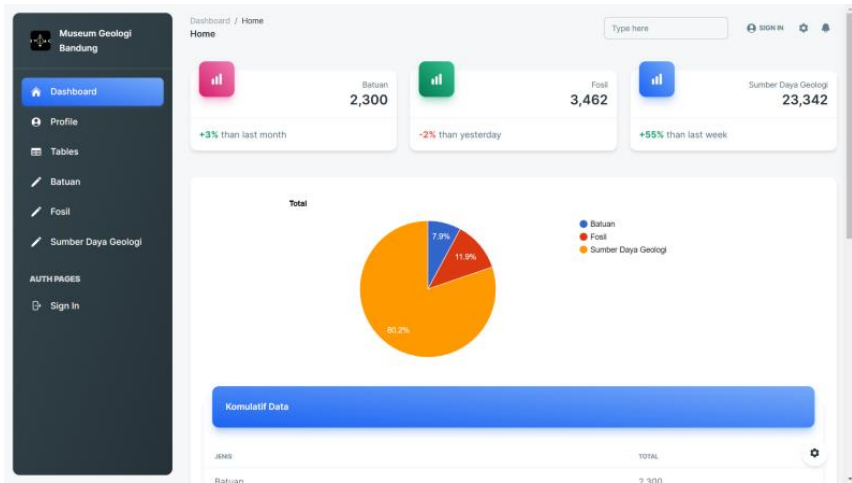
<div className="container mx-auto p-4">
  <Card className="absolute top-2/4 left-2/4 w-
full max-w-[24rem] -translate-y-2/4 -translate-x-2/4">
    <CardHeader
      variant="gradient"
      color="blue"
      className="mb-4 grid h-28 place-items-
center"
    >
      <Typography variant="h3" color="white">
        Sign In
      </Typography>
    </CardHeader>
    <CardBody className="flex flex-col gap-4">
      <Input type="email" label="Email" size="lg"
/>
      <Input type="password" label="Password"
size="lg" />
      <div className="-ml-2.5">
        <Checkbox label="Remember Me" />
      </div>
```

```

    </CardBody>
    <CardFooter className="pt-0">
      <Button variant="gradient" fullWidth>
        Sign In
      </Button>
      <Typography variant="small" className="mt-6
flex justify-center">
        Don't have an account?
        <Link to="/auth/sign-up">
          <Typography
            as="span"
            variant="small"
            color="blue"
            className="ml-1 font-bold"
          >
            Sign up
          </Typography>
        </Link>
      </Typography>
    </CardFooter>
  </Card>
</div>

```

## 2. Rancangan User Interface Dashboard



Gambar 31 Rancangan User Interface Dashboard

### Mengimport kebutuhan components

```
import React from "react";
import {
  Typography,
  Card,
  CardHeader,
  CardBody,
  IconButton,
  Menu,
  MenuHandler,
  MenuItem,
  Avatar,
  Tooltip,
  Progress,
```



```

    CardFooter,
} from "@material-tailwind/react";
import {
    ClockIcon,
    CheckIcon,
    EllipsisVerticalIcon,
    ArrowUpIcon,
} from "@heroicons/react/24/outline";

import { StatisticsCard } from "@widgets/cards";

import {
    statisticsCardsData,
    statisticsChartsData,
    projectsTableData,
    ordersOverviewData,
} from "@data";

import { Chart } from "react-google-charts";

import {
    totalData,
    optionsData,
    optionsSumberDayaGeologi,
    dataSumberDayaGeologi,
    dataFosil,

```

```

optionsFosil,
dataBatuan,
optionsBatuan,
} from "@data/pie-chart";

import { tableDataHome } from "@data";

```

## Memanggil Data Pie Chart

```

<Card>
  <CardBody>
    <CardHeader
      options={optionsData}
      variant="h3"
    >

    </CardHeader>

    <Chart
      chartType="PieChart"
      data={totalData}
      options={optionsData}
      width={"100%"}
      height={"400px"}
    />

    <Card>
      <CardHeader variant="gradient"
color="blue" className="mb-8 p-6">

        <Typography variant="h6" color="white">

```

```

        Komulatif Data
    </Typography>
  </CardHeader>
  <CardBody className="overflow-x-scroll
px-0 pt-0 pb-2">
    <table className="w-full min-w-[640px]
table-auto">
      <thead>
        <tr>
          {["Jenis", "Total"].map((el) => (
            <th
              key={el}
              className="border-b border-
blue-gray-50 py-3 px-5 text-left"
            >
              <Typography
                variant="small"
                className="text-[11px]
font-bold uppercase text-blue-gray-400"
              >
                {el}
              </Typography>
            </th>
          )))}
        </tr>
      </thead>
      <tbody>
        {tableDataHome.map(

```

```

        ({ jenis, total }, key) => {
            const className = `py-3 px-5
${key === tableDataHome.length - 1
                ? ""
                : "border-b border-blue-
gray-50"}

        `;
            return (
                <tr key={jenis}>
                    <td className={className}>
                        <Typography
                            variant="3"
                        >
                            {jenis}
                        </Typography>
                    </td>
                    <td className={className}>
                        <Typography
                            variant="3"
                        >
                            {total}
                        </Typography>
                    </td>
                </tr>
            );
        }
    )}

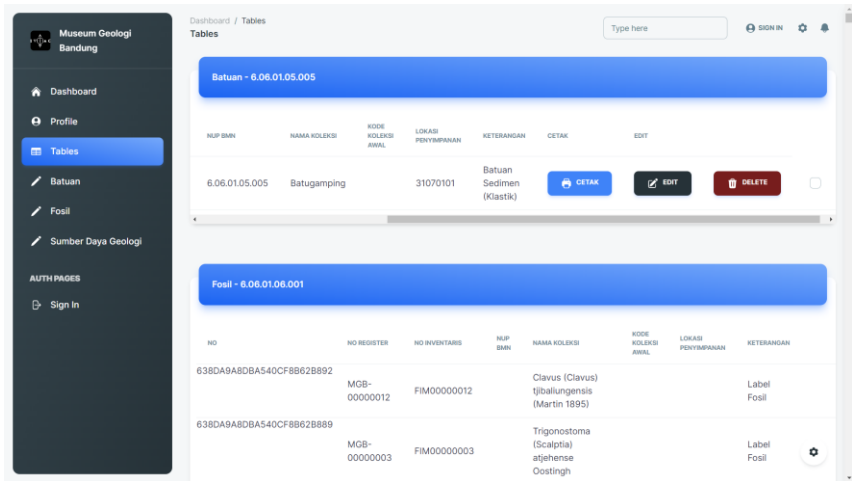
```

```

        </tbody>
      </table>
    </CardBody>
  </Card>
</CardBody>
<CardFooter>
  <Typography
    variant="small"
    className="flex items-center font-normal
text-blue-gray-600"
  >
    <ClockIcon strokeWidth={2} className="h-4
w-4 text-inherit" />
    &nbsp;just updated
  </Typography>
</CardFooter>
</Card>

```

### 3. Rancangan User Interface Tabel



Gambar 32 Rancangan User Interface Tabel

#### Mengimport kebutuhan components

```
import {
  Card,
  CardHeader,
  CardBody,
  Typography,
  Avatar,
  Checkbox,
  Button,
} from "@material-tailwind/react";
import { authorsTableData } from "@data";
import React, { useEffect, useState } from "react";
import axios from "axios";
```

#### Pemanggilan data API

```
const [data, setdata] = useState([]);
```

```

const [isLoading, setIsLoading] = useState(false);
const [isError, setIsError] = useState(false);

useEffect(() => {
    setIsLoading(true);
    // URL Ganti dengan alamat github atau API atau
URL API
    // Method @{get, post, put, patch, delete}
    axios
        .get("https://example-api.com/")
        .then((response) => {
            setData(response.data.data.data);
            console.log(response.data.data.data);
// Menampilkan console log
            setIsLoading(false);
        })
        .catch((err) => {
            // Jika Gagal
            console.log(err);
            setIsError(true);
            setIsLoading(false);
        });
}, []);

```

**Menampilkan tabel data**

```

<div className="mt-12 mb-8 flex flex-col
gap-12">

    <Card>

```

```

<CardHeader variant="gradient"
color="blue" className="mb-8 p-6">
    <Typography variant="h6"
color="white">
        Fossil -
    </Typography>
</CardHeader>
<CardBody className="overflow-x-
scroll px-0 pt-0 pb-2">
    <table className="w-full min-w-
[640px] table-auto">
        <thead>
            <tr>
                {["NO", "NO
REGISTER", "NO INVENTARIS", "NUP BMN", "NAMA KOLEKSI",
"KODE KOLEKSI AWAL", "LOKASI PENYIMPANAN",
"KETERANGAN", "", ""].map((e1) => (
                    <th
                        key={e1}
                        className="
border-b border-blue-gray-50 py-3 px-5 text-left"
                    >
                        <Typography
                            variant
="small"
                            classNa
me="text-[11px] font-bold uppercase text-blue-gray-400"
                        >
                            {e1}
                        </Typograph
y>

```



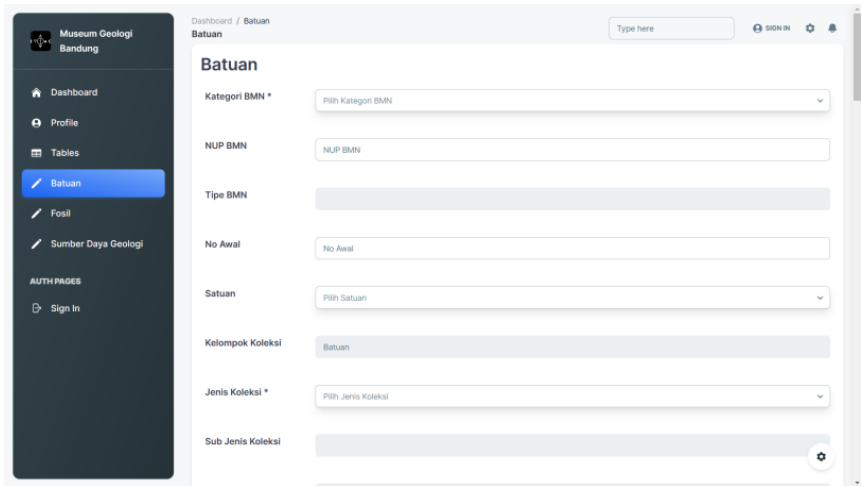
```

                </th>
            )))
        </tr>
    </thead>
    <tbody>

        {data &&
          data.map((item) =>
(
            <tr
className="border-b border-blue-gray-50 py-3 px-5 text-
left" key={item}>{item.id.toUpperCase()}
            <td
className="border-b border-blue-gray-50">
                {item.n
o_register}
            </td>
            <td>
                {item.n
o_inventaris}
            </td>
            <td>{item.n
ama_koleksi}</td>
            </tr>
          )))
        </tbody>
    </table>
  </CardBody>
</Card>

```

## 4. Rancangan User Interface Crud



*Gambar 33 Rancangan User Interface Crud*

### Mengimport kebutuhan components

```
import React from "react";
import {
  Card,
  CardHeader,
  CardBody,
  Input,
  Typography,
  Select,
  Option,
  Button,
  Textarea,
  Radio,
  Checkbox,
} from "@material-tailwind/react";
```

## Pembuatan Form Crud Data

```
<Card className="mb-12">
  <div className="">
    <CardHeader
      floated={false}
      shadow={false}
      className="lg:w-1/3 w-full">
      <Typography className=""
variant="h3">
        Fossil
      </Typography>
    </CardHeader>
    <CardBody>
      <div className="flex justify-
start">
        <Typography
          className="justify-start w-
60 md:md-inset-0"
          variant="h6"
        >
          Kategori BMN *
        </Typography>
        <Select className="w-full
shadow-lg" label="Pilih Kategori BMN" required>
          <Option>6.02.02.99.999</Opt
ion>
          <Option>6.06.01.05.005</Opt
ion>
```

```

                                <Option>6.06.01.06.001</Opt
ion>

                                </Select>

                                </div>

                                </CardBody>

                                <CardBody>
                                <div className="flex justify-
start">

                                    <Typography
                                    className="justify-start w-
60 md:md-inset-0"
                                    variant="h6"
                                    >
                                    NUP BMN
                                </Typography>
                                <div className="w-full">
                                    <Input
                                    label="NUP BMN"
                                    />
                                </div>
                                </div>
                                </CardBody>

                                <CardBody>
                                <div className="flex justify-
start">

                                    <Typography

```

```

        className="justify-start w-
60 md:md-inset-0"
        variant="h6"
    >
        Tipe BMN
    </Typography>
    <Input
        disabled
    />
</div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            No Awal
        </Typography>
        <Input label="No Awal" />
    </div>
</CardBody>

<CardBody>

```

```

        <div className="flex justify-
start">

            <Typography
                className="justify-start w-
60 md:md-inset-0"
                variant="h6"
            >
                Satuan
            </Typography>
            <Select className="shadow-lg"
label="Pilih Satuan">
                <Option>Buah</Option>
                <Option>Unit</Option>
                <Option>Set</Option>
            </Select>
        </div>
    </CardBody>

    <CardBody>
        <div className="flex justify-
start">

            <Typography
                className="justify-start w-
60 md:md-inset-0"
                variant="h6"
            >
                Kelompok Koleksi
            </Typography>

```

```

        <Input
            label="Fosil"
            value="Fosil"
            disabled
        />
    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Jenis Koleksi *
        </Typography>
        <Select className="shadow-lg"
label="Pilih Jenis Koleksi">
            <Option>Invertebrata</Optio
n>
            <Option>Mikrofosil</Option>
            <Option>Palobotani</Option>
            <Option>Vetebrata</Option>
        </Select>
    </div>
</CardBody>

```

```

        <CardBody>
            <div className="flex justify-
start">

                <Typography
                    className="justify-start w-
60 md:md-inset-0"
                    variant="h6"
                >
                    Sub Jenis Koleksi
                </Typography>
                <Select className="shadow-lg"
label="Pilih Jenis Koleksi">
                    <Option></Option>

                </Select>
            </div>
        </CardBody>

        <CardBody>
            <div className="flex justify-
start">

                <Typography
                    className="justify-start w-
60 md:md-inset-0"
                    variant="h6"
                >
                    Kode Jenis Koleksi

```



```

        </Typography>
        <Input
            className="grid justify-
items-start"
            label=""
            disabled
        />
    </div>
</CardBody>

<CardBody>
    <div className="grid grid-flow-row
auto-rows-max gap-4 md:auto-rows-min">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Ruang Penyimpanan
        </Typography>
        <div className="grid grid-rows-
1 grid-flow-col gap-4">
            <Radio id="dalamnegeri"
name="type" label="Storage" />
            <Radio id="luarnegeri"
name="type" label="Non Storage" />
        </div>
    </div>
</CardBody>

```

```

<CardBody>
    <div className="grid grid-flow-row
auto-rows-max gap-4 md:auto-rows-min">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Lokasi Penyimpanan :
        </Typography>
        <Select className="shadow-lg"
label="Pilih Storage">
            <Option>1</Option>
            <Option>2</Option>
            <Option>3</Option>
            <Option>4</Option>
            <Option>5</Option>
            <Option>6</Option>
            <Option>7</Option>
            <Option>8</Option>
            <Option>9</Option>
            <Option>10</Option>
            <Option>11</Option>
            <Option>12</Option>
            <Option>13</Option>
            <Option>14</Option>

```

```

        <Option>15</Option>
    </Select>
    <Select className="shadow-lg"
label="Pilih Lantai">
        <Option>1</Option>
        <Option>2</Option>
        <Option>3</Option>
        <Option>4</Option>
        <Option>5</Option>
        <Option>6</Option>
        <Option>7</Option>
        <Option>8</Option>
        <Option>9</Option>
        <Option>10</Option>
        <Option>11</Option>
        <Option>12</Option>
        <Option>13</Option>
        <Option>14</Option>
        <Option>15</Option>
    </Select>
    <Select className="shadow-lg"
label="Pilih Lajur">
        <Option>01</Option>
        <Option>02</Option>
        <Option>03</Option>
        <Option>04</Option>
        <Option>05</Option>

```

```

        <Option>06</Option>
        <Option>07</Option>
        <Option>08</Option>
        <Option>09</Option>
        <Option>10</Option>
        <Option>11</Option>
        <Option>12</Option>
        <Option>13</Option>
        <Option>14</Option>
        <Option>15</Option>
    </Select>
    <Select className="shadow-lg"
label="Pilih Laci">
        <Option>01</Option>
        <Option>02</Option>
        <Option>03</Option>
        <Option>04</Option>
        <Option>05</Option>
        <Option>06</Option>
        <Option>07</Option>
        <Option>08</Option>
        <Option>09</Option>
        <Option>10</Option>
        <Option>11</Option>
        <Option>12</Option>
        <Option>13</Option>
        <Option>14</Option>

```

```

        <Option>15</Option>
    </Select>
    <Input
        className=""
        label="Slot"
    >
</Input>
</div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">

        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Kondisi
        </Typography>
        <Select className="shadow-lg"
label="Pilih Jenis Koleksi">
            <Option>B - BAIK</Option>
            <Option>RR - Rusak
Ringan</Option>
            <Option>RB - Rusak
Berat</Option>
        </Select>

```

```

        </div>
      </CardBody>

      <CardBody>
        <div className="flex justify-
start">
          <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
          >
            Nama Koleksi
          </Typography>
          <Input
            className="grid justify-
items-start"
            label="Nama Koleksi"
          />
        </div>
      </CardBody>

      <CardBody>
        <div className="flex justify-
start">
          <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"

```

```

        >
            Deskripsi Koleksi
        </Typography>
        <Input
            className="grid justify-
items-start"
            label="Deskripsi Koleksi"
        />
    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Keterangan
        </Typography>
        <Textarea
            className="grid justify-
items-start"
            label="Keterangan"
        />
    </div>
</CardBody>

```

```

<CardBody>
  <div className="flex justify-
start">
    <Typography
      className="justify-start w-
60 md:md-inset-0"
      variant="h6"
    >
      Umur Geologi
    </Typography>
    <Select className="shadow-lg"
label="Pilih Umur Geologi">
      <Option>Prakambrium</Option>
    >
      <Option>Paleozoikum -
Kambrium</Option>
      <Option>Paleozoikum -
Ordovium</Option>
      <Option>Paleozoikum -
Silur</Option>
      <Option>Paleozoikum -
Devon</Option>
      <Option>Paleozoikum -
Karbon</Option>
      <Option>Paleozoikum -
Perem</Option>
      <Option>Mesozoikum -
Trias</Option>
      <Option>Mesozoikum -
Jura</Option>

```



```

        <Option>Mesozoikum -
Kapur</Option>

        <Option>Kenozoikum -
Paleogen</Option>

        <Option>Kenozoikum -
Eosen</Option>

        <Option>Kenozoikum -
Oligosen</Option>

        <Option>Kenozoikum -
Miosen</Option>

        <Option>Kenozoikum -
Pliosen</Option>

        <Option>Kenozoikum -
Plistosen</Option>

        <Option>Kenozoikum -
Holosen</Option>

    </Select>
</div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Nama Formasi
        </Typography>

```

```

        <Input
            className="grid justify-
items-start"
            label="Nama Formasi"
        />
    </div>
</CardBody>

<CardBody>
    <div className="grid grid-flow-row
auto-rows-max gap-4 md:auto-rows-min">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Ditemukan :
        </Typography>
        <div className="grid grid-rows-
1 grid-flow-col gap-4">
            <Radio id="dalamnegeri"
name="type" label="Dalam Negeri" />
            <Radio id="luarnegeri"
name="type" label="Luar Negeri" />
            <Radio id="nn" name="type"
label="NN" defaultChecked />
        </div>
    </div>
</CardBody>

```

```

        <CardBody>
            <div className="grid grid-flow-row
auto-rows-max gap-4 md:auto-rows-min">
                <Typography
                    className="justify-start w-
60 md:md-inset-0"
                    variant="h6"
                >
                    Pulau
                </Typography>
                <Input
                    className="grid justify-
items-start"
                    label="Pulau"
                />
            </div>
        </CardBody>

        <CardBody>
            <div className="grid grid-flow-row
auto-rows-max gap-4 md:auto-rows-min">
                <Typography
                    className="justify-start w-
60 md:md-inset-0"
                    variant="h6"
                >
                    Lokasi Temuan :

```

```

        </Typography>
        <Select className="shadow-lg"
label="Pilih Provinsi">
            <Option>Jawa Barat</Option>
            <Option>Jawa
Tengah</Option>
            <Option>Jawa Timur</Option>
        </Select>
        <Select className="shadow-lg"
label="Pilih Kabupaten/Kota">
            <Option>Bandung</Option>
            <Option>Bekasi</Option>
            <Option>Jakarta</Option>
        </Select>
        <Typography
        className="justify-start w-
60 md:md-inset-0"
        variant="h6"
        >
            Alamat Lengkap :
        </Typography>
        <Textarea
        className="grid justify-
items-start"
        label="Alamat"
        />
    </div>
</CardBody>

```

```

        <CardBody>
            <div className="grid grid-flow-row
auto-rows-max gap-4 md:auto-rows-min">
                <Typography
                    className="justify-start w-
max md:md-inset-0"
                    variant="h6"
                >
                    Kordinat :
                </Typography>
                <div>
                    <Typography>
                        Latitude
                    </Typography>
                    <Input
                        className="shadow-lg"
                        label="Latitude"
                    >
                    </Input>
                </div>
                <div>
                    <Typography>
                        Longitude
                    </Typography>
                    <Input
                        className="shadow-lg"

```

```

        label="Longitude"
      >
    </Input>
  </div>
  <div>
    <Typography>
      Elevasi
    </Typography>
    <Input
      className="shadow-lg"
      label="Elevasi"
    >
    </Input>
  </div>
</div>
</CardBody>

<CardBody>
  <div className="grid grid-flow-row
auto-rows-max gap-4 md:auto-rows-min">
    <Typography
      className="justify-start w-
60 md:md-inset-0"
      variant="h6"
    >
      Peta
    </Typography>

```

```

<div className="grid grid-rows-
2 grid-flow-col gap-4">
    <Checkbox id="1"
label="Rupa Bumi" />
    <Checkbox id="2"
label="Geologi" />
    <Checkbox id="3"
label="Blad" />
    <Checkbox id="4"
label="Luar Negeri" />
</div>
</div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Skala
        </Typography>
        <Select className="w-full
shadow-lg" label="Pilih Skala Peta">
            <Option>1:50.000</Option>
            <Option>1:100.000</Option>
            <Option>1:250.000</Option>

```

```

        </Select>

    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">

        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Lembar Peta
        </Typography>
        <Input
            className="grid justify-
items-start"
            label="Lembar Peta"
        />
    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">

        <Typography
            className="justify-start w-
60 md:md-inset-0"

```



```

        variant="h6"
      >
        Cara Perolehan
      </Typography>
      <Select className="shadow-lg"
label="Pilih Cara Perolehan">
        <Option>Pembuatan</Option>
        <Option>Pembelian</Option>
        <Option>Hibah</Option>
        <Option>Penyelidikan
Geologi</Option>
      </Select>
    </div>
  </CardBody>

  <CardBody>
    <div className="flex justify-
start">
      <Typography
        className="justify-start w-
60 md:md-inset-0"
        variant="h6"
      >
        Tahun Perolehan
      </Typography>
      <Input
        className=""
        label="Tahun Perolehan"

```

```

        >
        </Input>
    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Determinator
        </Typography>
        <Input
            className="grid justify-
items-start"
            label="Determinator"
        />
    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography

```

```

        className="justify-start w-
60 md:md-inset-0"
        variant="h6"
    >
        Kolektor
    </Typography>
    <Input
        className="grid justify-
items-start"
        label="Kolektor"
    />
</div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Kepemilikan Awal
        </Typography>
        <Select className="shadow-lg"
label="Pilih Kepemilikan Awal">
            <Option>Data Koleksi
Museum</Option>

```

```

                                <Option>Dienst van den
Mijnbouw</Option>
                                <Option>-</Option>
                                </Select>
                                </div>
                                </CardBody>

                                <CardBody>
                                <div className="flex justify-
start">
                                <Typography
                                className="justify-start w-
60 md:md-inset-0"
                                variant="h6"
                                >
                                Publikasi
                                </Typography>
                                <Textarea
                                className="grid justify-
items-start"
                                label="Publikasi"
                                />
                                </div>
                                </CardBody>

                                <CardBody>
                                <div className="flex justify-
start">

```

```

        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            URL
        </Typography>
        <Input
            className="grid justify-
items-start"
            label="URL"
        />
    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Nilai Perolehan
        </Typography>
        <Input
            className="grid justify-
items-start"

```

```

        label="Nilai Perolehan"
    />
</div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Nilai Buku
        </Typography>
        <Input
            className="grid justify-
items-start"
            label="Nilai Buku"
        />
    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography

```

```

        className="justify-start w-
60 md:md-inset-0"
        variant="h6"
    >
        Gambar 1
    </Typography>
    <Input
        className="grid justify-
items-start"
        type="file"
    />
    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Gambar 2
        </Typography>
        <Input
            className="grid justify-
items-start"
            type="file"

```

```

        />
    </div>
</CardBody>

<CardBody>
    <div className="flex justify-
start">
        <Typography
            className="justify-start w-
60 md:md-inset-0"
            variant="h6"
        >
            Gambar 3
        </Typography>
        <Input
            className="grid justify-
items-start"
            type="file"
        />
    </div>
</CardBody>

<CardBody>
    <div className="flex w-max gap-4">
        <Button
            variant="outlined"
color="red">
            Reset

```



```

</Button>

<Button>
  Simpan
</Button>

</div>

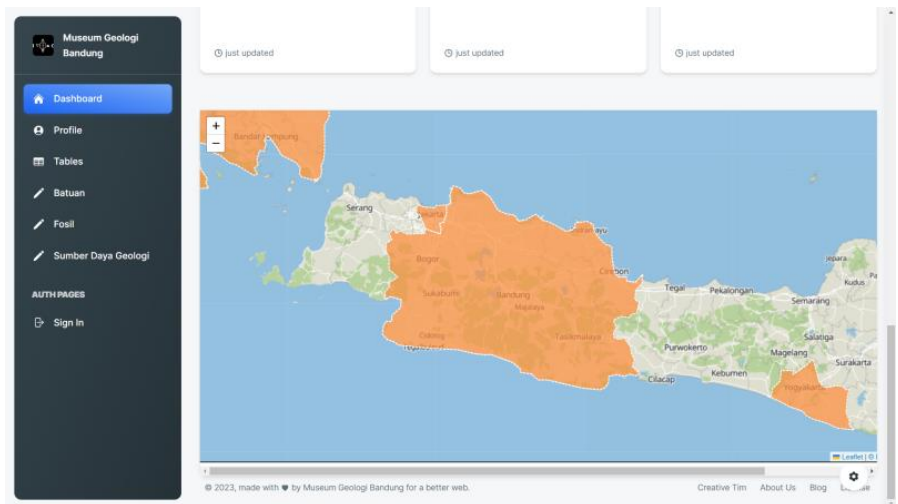
</CardBody>

</div>

</Card>

```

## 5. Rancangan User Interface OpenStreetMap



*Gambar 34 Rancangan User Interface Openstreetmap*

### Menampilkan data Openstreetmap dan Polygon

```

import React from 'react'

import { MapContainer, TileLayer, Polygon } from
'react-leaflet'

```

```

import 'leaflet/dist/leaflet.css';
import { statesData } from '../configs/data'

const center = [-6.8995231941771, 107.6210690166196];
const position = [-6.8995231941771, 107.6210690166196];
export default function gis() {
  return (
    <>
      <MapContainer
        center={position}
        zoom={8}
        style={{ width: '90vw', height: '70vh'
      }}
        className='w-full h-full md:md-cols-2'
      >
        <TileLayer
          url="https://api.maptiler.com/maps/basic-v2/256/{z}/{x}/{y}.png?key=AovbylPLN0yWA6s1eoNJ"
          attribution='<a href="https://www.maptiler.com/copyright/" target="_blank">&copy; MapTiler</a> <a href="https://www.openstreetmap.org/copyright" target="_blank">&copy; OpenStreetMap contributors</a>'
        />
        {
          statesData.features.map((state) =>
            {
              const coordinates =
state.geometry.coordinates[0].map((item) => [item[1],
item[0]]);

```

```

        return (
            <Polygon
                pathOptions={{
                    fillColor:
"#FD8D3c",
                    fillOpacity: 0.7,
                    weight: 2,
                    opacity: 1,
                    dashArray: 3,
                    color: 'white'
                }}
                positions={coordinates}
                eventHandlers={{
                    mouseover: (e) => {
e.target;

                        layer.setStyle(
{
                            fillOpacity
: 0.7,
                            weight: 5,
                            dashArray:
"",
                            color:
'#666',
                            fillColor:
'#D45962'
                        })
                    },

```

```

                                mouseout: (e) => {
                                    const layer =
e.target;

                                    layer.setStyle(
{
                                fillOpacity
: 0.7,
                                weight: 2,
                                dashArray:
"3",
                                color:
'white',
                                fillColor:
"#FD8D3c",

                                })
                                },
                                click: (e) => {

                                }

                                }}
                                />

                                )

                                })

                                }

                                </MapContainer>

                                </>

                                )

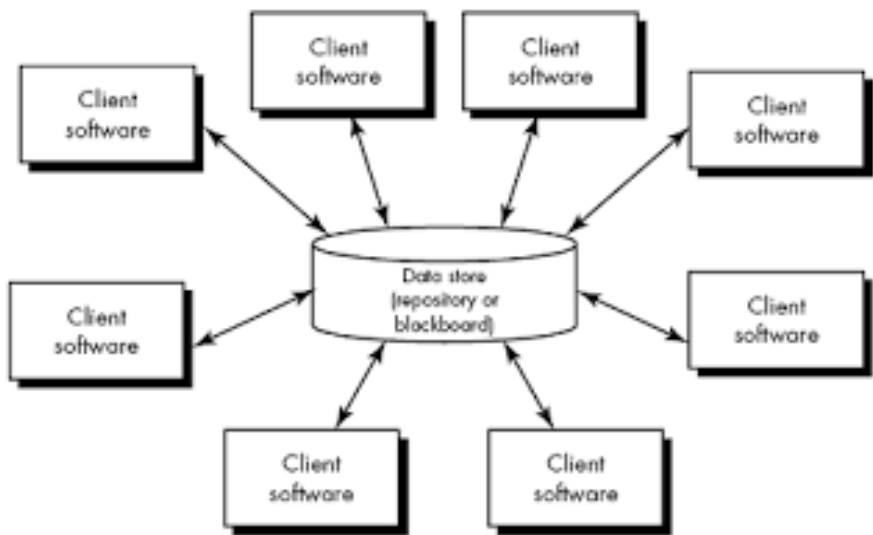
                                }

```

C. PERANCANGAN ARSITEKTUR PERANGKAT LUNAK DAN PERANGKAT KERAS SISTEM

1. Arsitektur Perangkat Lunak

Berikut ini adalah arsitektur perangkat lunak dengan menggunakan diagram.



Gambar 35 Arsitektur Perangkat Lunak

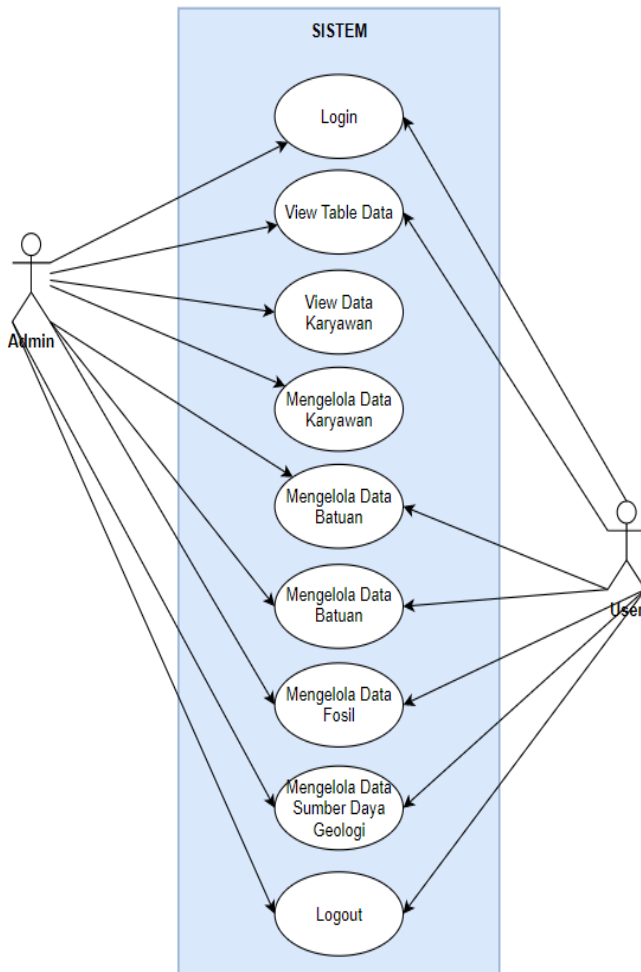
2. Perangkat Keras Sistem

Tabel 2 Perangkat Keras Sistem

No	Arsitektur Perangkat Keras Sitem
1	Memiliki hardware dasar yaitu CPU, Memori dan Input Output sistem.
2	Komputer merupakan stored program.
3	Bisa menjalankan instruksi secara berurutan.
4	Memiliki jalur (bus) yang letaknya diantara memory dan CPU.

D. PEMETAAN STRUKTUR DIAGRAM USER / AKTOR SISTEM

Berikut ini adalah Pemetaan Struktur Diagram User/Aktor Sistem yang dibangun



Gambar 36 Pemetaan Struktur Diagram User / Aktor Sistem

# BAB 6

## PENUTUP

---

### A. KESIMPULAN

Hasil kesimpulan dari penelitian yang telah dilakukan yaitu proses menerapkan Metode Design Sprint dan Data Spasial OpenStreetMaps untuk membantu membangun sistem Dashboard hasil olahan Batuan, Fosil dan Sumber Daya Geologi, Data Koleksi Museum terdiri dari 5 tahap yaitu tahap understand atau memahami kebutuhan sistem, tahap diverge yaitu proses penggambaran terhadap ide yang dipilih dengan memberikan sketsa, tahap decide yaitu melakukan keputusan terhadap ide yang telah digambarkan, tahap prototype yaitu proses pembentukan atau penerapan dari ide dan tahap validate yaitu proses persetujuan dari hasil prototype.

# DAFTAR PUSTAKA

- BADAN GEOLOGI - KEMENTERIAN ENERGI . (n.d.). (BADAN GEOLOGI) Retrieved Desember Jumat, 2022 from <https://geologi.esdm.go.id/id/profil/tugas-fungsi/museum-geologi>
- Borchard, L., Biondo, M., Kutay, S., Morck, D., & Weiss, A. P. (2015). Making journals accessible front & back: examining open journal systems at CSU. *OCLC Systems & Services: International digital library perspectives*, 31(1), 35 - 50.
- Dini Nurmalasari, R. T. (2015). Informational Dashboard untuk Monitoring Sistem Drainase secara Real-Time. *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI)*, 4(3), 141-146.
- Fataha, M. K. (2022). APLIKASI PENYEWAAN PERLENGKAPAN PENDAKIAN DENGAN MENGGUNAKAN TAILWIND CSS FRAMEWORK “STUDI KASUS OUTDOOR NGOPI\_AH YOGYA”. *Universitas Teknologi Digital Indonesia.*, 11-12.
- Fathoni, N. Z. (2018). *PENGEMBANGAN SISTEM INFORMASI KEGIATAN DAN TAGIHAN IUARAN WARGA BERBASIS WEB DENGAN LAYANAN SMS GATEWAY MENGGUNAKAN NODE.JS (STUDI KASUS : DUSUN KLUMPIT). YOGYAKARTA.*
- Indriani, S. D., Siti Khadijah, U. L., CMS, S., & Khoerunnisa, L. (2022). Konservasi Preventif Untuk Mencegah Kerusakan Koleksi pada Museum Geologi Bandung. *UNILIB : Jurnal Perpustakaan*, 13, 119 - 125.
- Peterson, A. T. (2013). Open access and the author-pays problem: Assuring access for readers and authors in the global academic community. *Journal of Librarianship and Scholarly Communication*.
- Raharjo, S. (2013). Building Web Journal Directory And Its Articles With Drupal. *Data Manajemen dan Teknologi Informasi (DASI)*.
- Ramadan, R., Az-Zahra, H. M., & Rokhmawati, R. I. (2019). Perancangan User Interface Aplikasi EzyPay menggunakan Metode Design Sprint (Studi Kasus PT. Arta Elektronik Indonesia). *Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3.



- SY, M. D., A, R. R., & Riana, R. (2011). Pembangunan Sistem Informasi Data Fossil Vertebrata Pada Museum Geologi Bandung Berbasis Web. *Perpustakaan UNIKOM*.
- Tiaranisa, A. Z., Siti Khadijah, U. L., CMS, S., & Khoerunnisa, L. (2022). PERLINDUNGAN KOLEKSI DI MUSEUM GEOLOGI MELALUI TINDAKAN PRESERVASI PREVENTIF. *BIBLIOTIKA: Jurnal Kajian Perpustakaan dan Informasi*, 6(2), 213 -221.
- Yusuf, D., & Supriyadi, S. (2022). PENERAPAN SISTEM KEHADIRAN MAHASISWA BERBASIS WEB. *JURNAL NUANSAN INFORMATIKA*, 16.

# GLOSARIUM

A

---

B

---

C

---

D

**Dashboard:** sebagai monitoring sistem drainase secara real time. Informational dashboard merupakan alat yang digunakan untuk menyajikan informasi dalam bentuk diagram, indikator visual, ataupun grafik berupa informasi yang ringkas dinamis serta relevan. Tujuan penggunaan informational dashboard diantaranya adalah untuk mempercepat proses pengambilan keputusan, mengukur kinerja, memonitoring proses yang sedang berjalan dan memprediksi kondisi dimasa mendatang.

**Design Sprint:** metodologi desain yang dikembangkan oleh Google untuk menyelesaikan masalah dengan melibatkan pengguna melalui perancangan, pembuatan prototype, dan pengujian ide dengan cepat.

---

E

---

F

---

G

---

H

---

I

---

J

---

K

---

L

---

M

---

N

**Node JS:** Sebagai runtime JavaScript berbasis peristiwa asinkron, Node.js dirancang untuk membangun aplikasi jaringan yang dapat diskalakan. Dalam contoh "halo dunia" berikut, banyak koneksi dapat ditangani secara bersamaan. Pada setiap koneksi, callback diaktifkan, tetapi jika tidak ada pekerjaan yang harus diselesaikan.

---

O

**OpenStreetMap:** Anda dapat memetakan secara detail karena setiap pengguna dimungkinkan untuk melakukan survei secara mandiri pada wilayah yang akan dipetakan.

---

P

---

Q

---

R

**React JS:** salah satu web framework paling populer di dunia Node.js, Dokumentasinya yang lengkap dan penggunaannya yang cukup mudah, dapat membuat kita mengembangkan berbagai produk seperti aplikasi web ataupun RESTful API dan dapat digunakan menjadi pijakan untuk membangun web framework yang lebih kompleks. Menurut penulis React adalah front-end library yang dikembangkan oleh Facebook. React digunakan untuk menangani lapisan tampilan untuk web dan aplikasi seluler.

---

S

---

T

**Tailwind CSS:** Tailwind CSS dikembangkan dengan tujuan untuk mempercepat proses prototyping halaman web agar sesuai kebutuhan (custom).

---

U

---

V

**Visual Studio Code:** Mempelajari kode itu menakutkan, jadi siapkan diri Anda untuk sukses dengan alat yang dibuat untuk Anda. Visual Studio Code adalah editor pengkodean gratis yang membantu Anda memulai pengkodean dengan cepat. Gunakan untuk membuat kode dalam bahasa pemrograman apa pun, tanpa berpindah editor. Visual Studio Code memiliki dukungan untuk banyak bahasa, termasuk Python, Java, C++, JavaScript, dan lainnya.

---

W

---

X

---

Y

---

z

---

## KREDIT GAMBAR

Gambar 1 OpenStreetMap PETA	
(sumber gambar: <a href="https://www.openstreetmap.org">https://www.openstreetmap.org</a> )	8
Gambar 2 Node JS	
(sumber gambar: <a href="https://nodejs.org/en/">https://nodejs.org/en/</a> )	10
Gambar 3 Visual Studio Code	
(sumber gambar: <a href="https://code.visualstudio.com/">https://code.visualstudio.com/</a> )	11
Gambar 4 React JS	
(sumber gambar: <a href="https://reactjs.org/">https://reactjs.org/</a> )	11
Gambar 5 Tailwind CSS	
(sumber gambar: <a href="https://tailwindcss.com/">https://tailwindcss.com/</a> )	12
Gambar 8 Pemahaman (JiraSoftware)	
(sumber gambar: <a href="https://www.atlassian.com/">https://www.atlassian.com/</a> )	16
Gambar 24 Gambar Situs Openstreetmap	
(sumber gambar: <a href="https://www.openstreetmap.org">https://www.openstreetmap.org</a> )	30
Gambar 25 Navigasi Peta	
(sumber gambar: <a href="https://www.openstreetmap.org">https://www.openstreetmap.org</a> )	31

# INDEKS

**A**

**B**

**C**

**D**

Design Sprint 4,

Data Spasial OpenStreetMap 8,9

Dashboard 4,9

**E**

**F**

**G**

**H**

**I**

**J**

**K**

**L**

**M**

**N**

Node JS 10,11

**O**

OpenStreetMap 27

**P**

Polygon 29,30

**Q**

## **R**

React JS 6,7

## **S**

## **T**

Tailwind CSS 7,8

## **U**

## **V**

Visual Studio Code 11

## **W**

## **X**

## **Y**

## **Z**



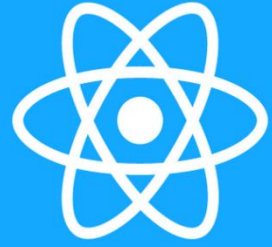
## TENTANG PENULIS



Roni Andarsyah, lahir di Padaherang pada tanggal 20 Mei 1988. Pendidikan D3 Politeknik Pos Indonesia, Melanjutkan Pendidikan S1 Sekolah Tinggi Sians & Teknologi Indonesia dan diselesaikannya S2 di Sekolah Tinggi Manajemen Informatika dan Komputer LIKMI.



Ariq Rafi Kusumah, lahir di Kabupaten Bekasi pada tanggal 13 Februari 2000. Pendidikan tingkat dasar hingga menengah dan atas ditempuh di Bekasi, Melanjutkan Pendidikan D4 di Universitas Logistik dan Bisnis Internasional (ULBI) Bandung.



kami menyampaikan tuntunan praktis 5 Tahap Menjadi Dashboard Admin Untuk Kemudahan Programmer Dengan ReactJS dan TailwindCSS ini kepada para pembaca yang terhormat. Karena hanya dengan karuniaNya kami dapat menyelesaikan buku ini. Buku ini membahas langkah demi langkah bahasa pemrograman Dashboard, dengan disertai contoh-contoh untuk memberikan petunjuk dan kemudahan bagi pembaca yang mempelajari Bahasa ReactJS dan TailwindCSS. Dengan dibuatnya buku ini, penulis berharap bahwa pembuatan program aplikasi dengan bahasa ReactJS dan TailwindCSS tidaklah menjadi sesuatu yang sulit.



GitHub

