

Penerapan Next.js dan GraphQL dalam Pengembangan Mobile Web Sistem Informasi Manajemen Puskesmas

Application of Next.js and GraphQL in Mobile Web Development of Puskesmas Management Information System

Dwi Aulia Anugerah^{a,1}, Sandy Kosasi^{b,2}

^{a,b}STMIK Pontianak, Jalan Merdeka No. 372, Pontianak, 78116, Indonesia
dwiailiaaaa@gmail.com¹, sandykosasi@stmikpontianak.ac.id²

ABSTRAK

Penelitian ini berfokus pada pengembangan Sistem Informasi Manajemen Puskesmas berbasis Web Mobile yang dilatarbelakangi permasalahan yang sedang dihadapi Puskesmas Rawat Inap Jungkat, pengelolaan data pasien yang masih manual dan dilakukan secara berulang-ulang mengakibatkan redundansi data dan memperlambat pelayanan. Penelitian ini menggunakan metode Design Science Research, data yang dikumpulkan melalui observasi, wawancara, dan studi dokumentasi. Observasi mencakup proses, langkah dalam alur pelayanan, serta hambatan yang dihadapi petugas dan pasien. Wawancara dilakukan dengan kepala puskesmas, petugas administrasi, dan perawat untuk menggali kebutuhan sistem. Hasil penelitian menunjukkan bahwa sistem berbasis Next.js untuk front-end, Laravel untuk back-end, dan GraphQL sebagai protokol pengambilan data mampu menyederhanakan proses pendaftaran pasien dan integrasi data antar layanan, dan memberikan kemudahan akses informasi kepada petugas karena input dilakukan hanya satu kali. Pengujian White-Box menunjukkan sistem berjalan sesuai skenario fungsionalitas yang dirancang. Kesimpulannya, penerapan sistem ini meningkatkan efisiensi operasional, meminimalkan redundansi data, dan memudahkan akses informasi bagi pasien maupun petugas.

Kata Kunci : Sistem Informasi Manajemen Puskesmas, Web Mobile, Next.js, GraphQL, Design Science Research (DSR).

ABSTRACT

The research focuses on the development of a Mobile Web-based Puskesmas Management Information System based on the problems being faced by the Puskesmas Rawat Inap Jungkat, patient data management which is still manual and carried out repeatedly, resulting in data redundancy and slowing down services. This research uses the Design Science Research method, data collected through observation, interviews, and documentation studies. Observation includes the process, steps in the service flow, and obstacles faced by officers and patients. Interviews were conducted with the head of the health center, administrative officers, and nurses to explore system needs. The results showed that the Next.js-based system for the front-end, Laravel for the back-end, and GraphQL as a data retrieval protocol were able to simplify the patient registration process and data integration between services, and provide easy access to information to officers because input was done only once. White-Box testing shows the system runs according to the designed functionality scenario. In conclusion, the implementation of this system improves operational efficiency, minimizes data redundancy, and facilitates access to information for both patients and officers.

Keywords : Puskesmas Management Information System (SIMPUS), Web Mobile, Next.js, GraphQL, Design Science Research (DSR).

Disubmit:

Info Artikel :
Direview:

Diterima :

Copyright © 2024 – CSRID Journal. All rights reserved.

1. PENDAHULUAN

Pelayanan di bidang kesehatan merupakan bentuk konteks pelayanan publik dan mutlak dilaksanakan dengan baik oleh pemerintahan. Agar penyelenggaraan pelayanan kesehatan dapat mencapai tujuan yang

diinginkan maka pelayanan harus menyediakan sarana dan prasarana, saling berhubungan antara pasien dan pemberi pelayanan, mudah dijangkau, dan bermutu [1]. Puskesmas Rawat Inap Jungkat memiliki kebutuhan yang besar untuk meningkatkan pelayanan informasi dan manajemen penyelenggaraan dalam mendukung Sistem Informasi Puskesmas yang mampu menjamin ketersediaan data dan informasi secara cepat, akurat, terkini, berkelanjutan, dan dapat dipertanggungjawabkan [2].

Keterbatasan dalam pengelolaan data pada Puskesmas Rawat Inap Jungkat disebabkan pendaftaran masih dilakukan secara berulang-ulang dengan mendatangi loket kemudian mendapatkan nomor antrian berupa kertas, setelah antrian dipanggil, kemudian pasien menuju poliklinik untuk mendaftar kembali sebagai pasien poliklinik. Pengelolaan data yang terbilang masih berulang-ulang pada pelayanan ini mengharuskan setiap poliklinik menginput data kembali sebelum melakukan pemeriksaan atau tindakan. Puskesmas Rawat Inap Jungkat membutuhkan sistem terintegrasi yang efisien yang dapat menunjang pelayanan puskesmas dan membantu manajemen data untuk menghasilkan informasi yang bermanfaat dan akurat. Selain itu, pemilihan menggunakan mobile daripada website juga karena sekarang layanan yang tersedia dalam mobile telah memungkinkan komunikasi, kolaborasi dan koordinasi melalui perangkat mobile [3]. Penggunaan Responsive Web Design pada sistem ini, dapat membuat satu situs yang terlihat dan berfungsi dengan baik dari semua perangkat, mulai dari tampilan yang dipasang di web hingga telepon genggam [4].

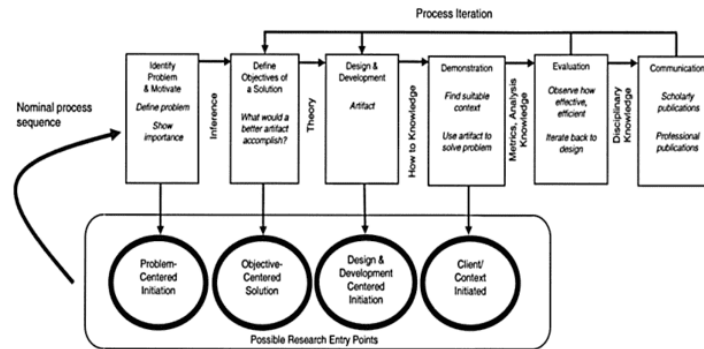
Keberhasilan mengatasi permasalahan ini menggunakan mobile web yang fleksibel dapat meningkatkan efisiensi dan efektivitas pelayanan, kualitas pelayanan, kepuasan pasien serta meningkatkan kinerja puskesmas. Penerapan teknologi informasi perlu mendapat dukungan penuh dari pihak manajemen rumah sakit sehingga penggunaan teknologi informasi dapat dimaksimalkan dalam mencapai tujuan organisasi [5]. Dengan penerapan desain antarmuka yang responsive dibutuhkan tampilan yang menarik dan tata letak desain yang dapat bekerja dengan baik pada semua perangkat devices [6], sehingga memudahkan akses bagi pasien dan petugas.

Pengembangan sistem dengan menggunakan Framework Next.js dan Laravel dan Protokol GraphQL diharapkan dapat mengoptimasi pengembangan mobile web yang akan menyederhanakan proses pendaftaran atau registrasi dan pengelolaan data pasien. Perancangan sebuah website dengan menggunakan Framework dapat membantu perancangan dan pengembangan website sehingga pengembang tidak perlu membangun sebuah website dari awal lagi jika akan memberikan fitur-fitur kepada website yang akan dibuat [7]. Selain itu, penggunaan GraphQL dalam pengambilan data yang lebih spesifik akan meningkatkan efisiensi sistem, sebuah kueri GraphQL hanya meminta data yang dibutuhkan [8] hal ini dapat meningkatkan efektivitas kinerja sistem.

Berdasarkan penelitian terdahulu menunjukkan bahwa penerapan sistem informasi manajemen di Puskesmas mampu meningkatkan pelayanan kesehatan dan kepuasan pasien [9], dimana data pasien dan penyampaian informasi puskesmas dapat diakses secara online [10]. SIMPUS dengan kelengkapan data tidak mengganggu pelayanan dan sistem pelaporan mampu memberikan informasi baik bagi puskesmas maupun untuk jenjang administrasi yang lebih tinggi untuk manajemen kesehatan Puskesmas yang lebih optimal [11], diharapkan mobile sistem informasi manajemen puskesmas yang dibangun dengan dapat meningkatkan pelayanan dan mendukung pengelolaan data yang lebih baik dan terintegrasi antar layanan yang ada di Puskesmas Rawat Inap Jungkat sesuai dengan perkembangan teknologi yang pesat saat ini.

2. METODE

Pengembangan sistem dalam penelitian ini mengadopsi metode Design Science Research (DSR). Setiap langkah dalam proses DSR dilakukan dengan mengikuti pendekatan sistematis untuk memecahkan masalah dengan membangun dan mengevaluasi artefak [12].



Gambar 1. Design Science Research (DSR)

A. Identify Problem and Motivate

Kegiatan ini menentukan spesifikasi masalah penelitian dan nilai solusi, dengan melakukan observasi langsung ke lokasi Puskesmas Rawat Inap Jungkat dengan mengamati proses semua alur pelayanan dari proses pendaftaran hingga tindakan yang diberikan kepada pasien. Selain itu, juga dilakukan wawancara terhadap Kepala Puskesmas Rawat Inap Jungkat, petugas administrasi, petugas kesehatan, dan pasien untuk menggali permasalahan dan kebutuhan sistem. Sehingga didapatkan permasalahan berupa waktu tunggu yang lama, proses input data pasien dengan data yang sama dilakukan berulang-ulang dari loket, poliklinik, dan kembali meninput data tersebut satu persatu saat dokter/perawat melakukan tindakan. Hal tersebut menghambat proses tindakan pengobatan karena banyaknya proses peninputan data yang sama di setiap tahapan layanan.

B. Define Objectives of a Solution

Tahap menentukan tujuan solusi, tujuan yang diharapkan Puskesmas Rawat Inap Jungkat dapat meningkatkan pelayanannya dengan mengembangkan sistem informasi manajemen yang terintegrasi antar layanan Puskesmas Rawat Inap Jungkat dan pendaftaran dapat dilakukan oleh pasien langsung yang ingin mendaftar dengan mengakses situs web yang disediakan.

C. Design and Development

Pada tahap ini dilakukan pembuatan design dan pengembangan sistem yang mencakup arsitektur sistem dan perancangan basis data, kemudian dilanjutkan dengan pengembangan melalui pengkodean untuk menghasilkan sistem. Pada pengkodean menggunakan framework Next.js sebagai front-end code, Laravel digunakan sebagai back-end code, dan GraphQL digunakan sebagai protocol pengambilan data yang lebih spesifik dan meminta data yang dibutuhkan berdasarkan data awal pada pendaftaran pasien.

1) Next.js

Next.js menyediakan performa aplikasi/sistem untuk melakukan pre-rendering yaitu melalui Server-Side Rendering (SSR) memungkinkan pengembangan aplikasi yang lebih dinamis dengan menghasilkan HTML secara real-time pada saat request diterima, sehingga cocok untuk data yang selalu diperbarui atau aplikasi yang memerlukan personalisasi konten berdasarkan data pengguna dan pengguna dapat mengoperasikan sistem lebih cepat dan efisien.

2) GraphQL

GraphQL digunakan untuk Next.js mengambil data yang spesifik dari server, sehingga hanya data dari pasien yang dibutuhkan saja yang akan dikirimkan ke poliklinik lain dalam tindakan pengobatan. GraphQL juga digunakan pada Laravel di back-end sebagai GraphQL server.

D. Demonstration

Demonstrasi yang dilakukan berupa solusi yang dikembangkan yang menghasilkan Sistem Informasi Manajemen Puskesmas dari penerapan Next.js dan GraphQL. Hal ini ditunjukkan dapat berupa interface untuk pengguna yang terdiri atau pasien, dokter, dan admin.

E. Evaluation

Evaluasi mengukur seberapa baik artefak mendukung solusi untuk masalah. Evaluasi ini dilakukan dengan melalui pengujian White-Box menggunakan teknik Basic Path yang bertujuan untuk mengidentifikasi dan memperbaiki kesalahan dalam struktur internal sistem [12]. Basic Path memungkinkan perancang test case untuk menghasilkan pengukuran kompleksitas logika dari perancangan sistem dan mengidentifikasi jalur dasar eksekusi [13], serta memastikan bahwa setiap jalur dalam sistem telah diuji secara menyeluruh, guna menemukan kesalahan atau bug yang mungkin terjadi pada mobile web sistem informasi manajemen puskesmas.

F. Communication

Di sini semua aspek masalah dan artefak yang dirancang dikomunikasikan kepada pemangku kepentingan terkait. Bentuk komunikasi yang tepat dikerjakan tergantung pada tujuan penelitian dan audiens, seperti berlatih profesional. Pada penelitian ini komunikasi akan dilakukan dengan menyampaikan hasil demonstrasi kepada Puskesmas Rawat Inap Jungkat dan melihat apakah permasalahan yang dihadapi sudah teratasi.

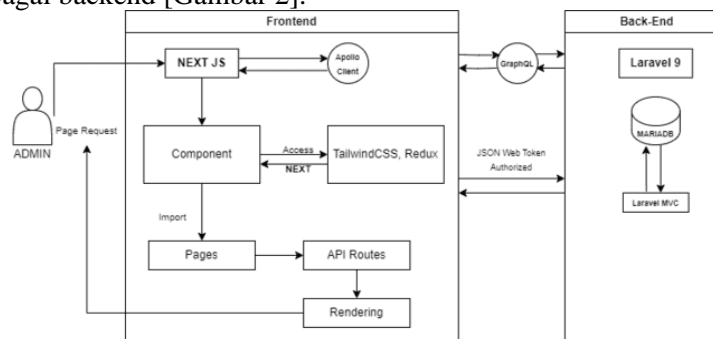
3. HASIL DAN PEMBAHASAN

Penerapan Next.js dan GraphQL dalam pengembangan Mobile Web Sistem Informasi Manajemen Puskesmas, Next.js diterapkan sebagai front-end yang menyajikan halaman dan komponen User Interface dengan tampilan berbasis Responsive Web Design untuk Mobile Web, GraphQL diterapkan untuk memungkinkan Next.js meminta data secara spesifik sesuai kebutuhan, dimana Next.js akan mengambil data dari back-end melalui GraphQL. GraphQL memberikan kontrol penuh pada Next.js untuk meminta data yang diperlukan. Selain itu, Next.js memanfaatkan SSR (Server-Side Rendering) dan SSG (Static Site Generation) untuk mempercepat pemuatan halaman dan meningkatkan performa SEO. Dengan pendekatan ini, sistem informasi manajemen puskesmas berbeda dari sebelumnya, dimana sistem yang dikembangkan tidak hanya memiliki kinerja yang cepat dan optimal tetapi juga efisiensi dalam pengelolaan data yang tidak memakan waktu.

A. Design and Development

1) Arsitektur Sistem

Arsitektur dari sistem merupakan sekumpulan dari model-model terhubung yang menggambarkan sifat dasar dari sebuah sistem. Keanekaragaman dari banyak model menggambarkan bagian berbeda dan aspek atau pandangan yang berbeda dari suatu sistem. Pada penelitian ini menggunakan Next.js sebagai front-end. Dimana Next.js dan React akan membantu meningkatkan kecepatan, performa, dan responsivitas web profil, serta memperkaya pengalaman pengguna dalam mengakses dan berinteraksi dengan web profil [14], dan Laravel digunakan sebagai backend [Gambar 2].



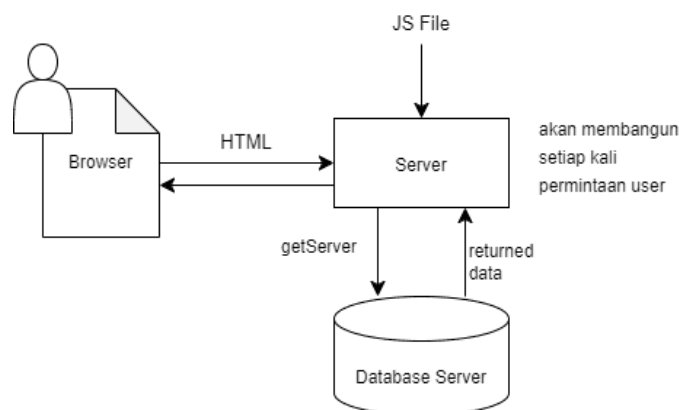
Gambar 2. Arsitektur Sistem Mobile Web SIMPUS

Gambar 2 merupakan arsitektur sistem dengan menerapkan konsep Pages-Component-Context-API routes dari Next.js pada Mobile Web SIMPUS pada Puskesmas Rawat Inap Jungkat. Pada sisi Frontend dengan Next.js, pages digunakan untuk menampilkan konten dan menerima

interaksi pengguna. Components digunakan untuk membuat tampilan dan perilaku halaman. Kemudian, Context dapat digunakan untuk menyediakan data dan fungsionalitas ke komponen-komponen. Components dapat menggunakan context untuk mengakses data dan fungsionalitas tersebut. Misalnya, context dapat digunakan untuk menyediakan data pengguna ke semua komponen di aplikasi. Pages dapat digunakan untuk memanggil API routes. API routes dapat digunakan untuk mengakses data dari sumber daya melalui GraphQL. Apollo Client digunakan untuk berkomunikasi dengan server GraphQL yang di-host oleh Laravel di backend. Apollo Client memungkinkan frontend untuk mengambil dan mengelola data dari backend melalui query GraphQL secara efisien. Selain itu, frontend menggunakan TailwindCSS untuk styling, dan Redux untuk manajemen state global di aplikasi.

Di sisi backend, Laravel 9 bertindak sebagai kerangka kerja utama yang mengelola logika bisnis, autentikasi, dan interaksi dengan basis data MariaDB. Laravel juga mengimplementasikan pola desain MVC (Model-View-Controller) untuk mengorganisir kode dan memastikan pemisahan tanggung jawab yang baik. Di Laravel, GraphQL digunakan sebagai antarmuka API, memungkinkan frontend untuk melakukan query dan mutasi data dengan fleksibilitas yang tinggi. Pengguna mengakses aplikasi melalui antarmuka yang dirender oleh Next.js, yang kemudian berkomunikasi dengan backend Laravel untuk mengelola data dan logika bisnis, dengan Apollo Client sebagai jembatan antara kedua bagian tersebut menggunakan GraphQL sebagai protokol komunikasi.

Adapun Server-Side Rendering (SSR) dalam Next.js yang berperan dalam sistem ini. SSR adalah teknik untuk merender komponen React di server, sebelum mengirim HTML ke klien. Hal ini dapat meningkatkan performa dan SEO aplikasi React, karena dapat mengurangi waktu yang dibutuhkan untuk memuat halaman konten pertama dari pendaftaran pasien, dimana SSR dapat merender halaman Next.js tanpa harus mengkonfigurasi tools tambahan apa pun. Cara kerjanya, file JS dikirimkan ke server, ketika user (browser) meminta, server akan menjalankan fungsi `getServerSideProps()` [Gambar 3]. Setelah data diambil, data tersebut akan dibangun di server termasuk data dari API dan server akan mengirimkan HTML ke user, sehingga dengan arsitektur ini dapat mengoptimalkan waktu saat sistem digunakan oleh pengguna.

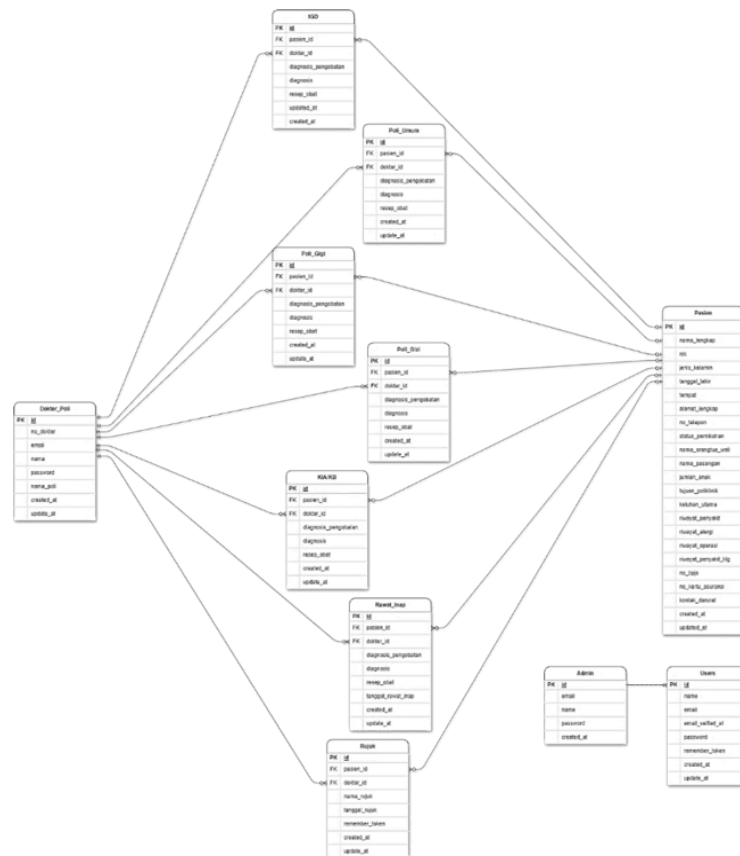


Gambar 3. Arsitektur Server-Side Rendering

2) Perancangan Basis Data

Perancangan basis data dilakukan agar semua data dapat tersimpan secara terstruktur di dalam database, setiap web wajib memiliki basis data sesuai dengan keperluannya masing-masing, maka perlu dibuat satu basis data baru yang akan tersimpan ke dalam server agar dapat

digunakan secara bersamaan. Perancangan basis data dilakukan dengan beberapa metode dan pendekatan yang dapat digunakan untuk menciptakan struktur database yang efisien, terstruktur dan sesuai dengan kebutuhan aplikasi data disusun dalam bentuk kumpulan yang terdiri dari sejumlah bidang yang membentuk database dalam suatu program computer. Saat informasi spesifik diperlukan, data ini diambil untuk menjaga database tetap terstruktur dan terorganisir, diperlukan suatu Entity Relationship Diagram (ERD) Diagram hubungan entitas atau entity relation diagram digunakan untuk menggambarkan relationship antar entitas yang relevan dari suatu sistem yang umumnya digunakan untuk memodelkan rancangan atau blue print dari basis data [15].



Gambar 4. Entity Relationship Diagram (ERD)

Diagram hubungan entitas Mobile Web Simpus pada Puskesmas Rawat Inap Jungkat menggambarkan interkoneksi antar tabel database, memperlihatkan bagaimana setiap tabel berhubungan satu sama lain. Melalui gambaran ini, dapat dipahami bahwa setiap tabel memiliki peran dan fungsi khusus yang sesuai dengan tujuan masing-masing. Data dalam setiap tabel saling melengkapi dan berinteraksi melalui hubungan yang telah dirancang. Meskipun data terdistribusi di berbagai tabel, melalui diagram ini, data tersebut dapat disajikan secara komprehensif sesuai dengan kebutuhan penggunaan. Diagram hubungan entitas membantu memvisualisasikan struktur dan relasi data dalam database dengan cara yang jelas dan terstruktur.

3) Coding (Pengkodean)

Tahapan coding menghasilkan tampilan dari sistem yang dirancang. Perancangan website dibuat menggunakan framework Next.js sebagai front-end merequest data melalui GraphQL sebagai web service dalam pertukaran data yang ideal. Implementasi GraphQL menggunakan

‘useLazyQuery’ Hook dari ‘@apollo/client’ untuk menjalankan query GraphQL secara manual pada halaman Login Admin [Gambar 14].

```
pages > JS index.js > ...  
11 async function handleLogin(id,email){  
27 }  
28 export default function Home() {  
29   const [userLogin,{loading,data,error}]=useLazyQuery(LOGIN);  
30   const [errorText,setErrorText]=useState({  
31     email:'masukan email harus terisi',  
32     password: 'masukan password harus terisi'  
33   });  
34   const {loadingAuth,loggedIn}=useAuth();  
35   const router=useRouter();  
36   const [toggle,setToggle]=useState(false)  
37   const [formValue,setFormValue]=useState({  
38     email:'',  
39     password:''  
40   })  
41 }
```

Gambar 14. Implementasi GraphQL pada Front-end

Pada Gambar 14 useAuth juga digunakan untuk memeriksa apakah pengguna sudah login (‘loggedIn’). Jika ya, pengguna akan diarahkan ke halaman utama (‘/main’).

```
lib > apollo > queries > JS login.js > LOGIN  
1 import { gql } from "@apollo/client";  
2 const LOGIN = gql`  
3   query LoginAdminQuery($email:String,  
4     $passct:String,  
5     $passiv:String,  
6     $passs:String)  
7   {  
8     LoginAdminQuery(  
9       email:$email,  
10      passct:$passct,  
11      passiv:$passiv,  
12      passs:$pass  
13    )  
14    {  
15      id  
16      email  
17    }  
18  }`;  
19 export default LOGIN;
```

Gambar 15. Library GraphQL Apollo/queries login.js

Gambar 15 mendefinisikan sebuah query GraphQL untuk melakukan login sebagai admin “LoginAdminQuery”. Query ini mengambil empat parameter (email, passsct, passiv, passss), lalu mengirimkan query ini ke server untuk mendapatkan respons berupa “id” dan “email”.

```
app > GraphQL > Types > LoginAdminType.php > LoginAdminType > fields  
1 <?php  
2  
3 declare(strict_types=1);  
4  
5 namespace App\GraphQL\Types;  
6  
7 use Rebing\GraphQL\Support\Type as GraphQLType;  
8 use GraphQL\Type\Definition\Type;  
9  
10 class LoginAdminType extends GraphQLType  
11 {  
12   protected $attributes = [  
13     'name' => 'LoginAdminType',  
14     'description' => 'A type'  
15   ];  
16  
17   public function fields(): array  
18   {  
19     return [  
20       'id' => [  
21         'type' => Type::int(),  
22         'description' => 'id admin'  
23       ],  
24       'email' => [  
25         'type' => Type::string(),  
26         'description' => 'email admin'  
27       ],  
28     ];  
29   }  
30 }  
31
```

Gambar 16. GraphQL LoginAdminType.php

Gambar 16 mendefinisikan GraphQL Type dari back-end dengan Laravel yang terdiri dari dua field utama yaitu “id” dan “email”. Type ini digunakan dalam query GraphQL untuk mengembalikan data admin setelah login. Setiap field dilengkapi dengan tipe datanya dan deskripsinya untuk memperjelas tujuan dari field tersebut dalam skema GraphQL.

```

app > GraphQL > Query > LoginAdminQuery.php
16 class LoginAdminQuery extends Query
17 {
18     public function type(): Type
19     {
20         return GraphQL::type('LoginAdminType');
21     }
22
23     public function args(): array
24     {
25         return [
26             'email' => [
27                 'name' => 'email',
28                 'type' => Type::string(),
29             ],
30             'password' => [
31                 'name' => 'password',
32                 'type' => Type::string(),
33             ],
34             'password2' => [
35                 'name' => 'password2',
36                 'type' => Type::string(),
37             ],
38             'password3' => [
39                 'name' => 'password3',
40                 'type' => Type::string(),
41             ],
42             'password4' => [
43                 'name' => 'password4',
44                 'type' => Type::string(),
45             ],
46             'password5' => [
47                 'name' => 'password5',
48                 'type' => Type::string(),
49             ],
50 ];
51 }
52
53 public function resolve($root, array $args, $context, ResolveInfo $resolveInfo, Closure $getSelectFields)
54 {
55     // @var SelectFields $fields
56     // return Admin::where('email' => $args['email'])->first();
57     $passwordWithoutHashing = $this->cryptoHashDecrypt($args['password'], $args['password2'], $args['password3'], $args['password4'], $args['password5']);
58     $dataUser = Admin::where('email', $args['email'])->first();
59     if ($dataUser) {
60         return collect([
61             'id' => $dataUser->id,
62             'name' => $dataUser->name,
63             'email' => $dataUser->email,
64         ]);
65     } else {
66         if (Hash::check($passwordWithoutHashing, $dataUser->password)) {
67             return $dataUser;
68         } else {
69             return collect();
70         }
71     }
72 }

```

Gambar 17. GraphQL LoginAdminQuery.php

Gambar 17 mendefinisikan sebuah GraphQL Query untuk login admin dengan beberapa argumen berupa email dan tiga bagian password. Query ini berfungsi untuk mencari data admin berdasarkan email yang diberikan, menggabungkan dan mendeskripsikan tiga bagian password untuk mencocokkan dengan password yang tersimpan di database. Jika data admin dan password cocok, informasi admin “id” dan “email” akan dikembalikan. Jika tidak, akan dikembalikan data default dengan id “0” dan email kosong.

B. Demonstration

Demonstrasi membantu menunjukkan bahwa solusi menerapkan Next.js dan GraphQL dapat bekerja sesuai spesifikasi dalam pengembangan Sistem Informasi Manajemen Puskesmas, dimana permintaan Next.js kepada back-end melalui GraphQL berhasil ditampilkan. Demonstrasi yang ditunjukkan dapat berupa antarmuka untuk pengguna Tampilan Kelola Poliklinik kepada admin ketika pasien berhasil mendaftar sesuai poliklinik yang ingin dituju [Gambar 18].

No.	Nama Pasien	NIK	Kelola
1	Prof. Alek Cole	5621107562791080	Kelola
2	Hunter Brown	4925044484797353	Kelola
3	Pablo Köhler	5171759962798691	Kelola
4	Otilia VonRueden	0688538632414001	Kelola
5	Prof. Malika Beahan	7913684296631882	Kelola
6	Jeffery Reichel	3672469038974051	Kelola
7	Miss Mariot Senger	0609670625243808	Kelola
8	Gabe McLaughlin	9773299952328904	Kelola

Gambar 18. Tampilan Daftar Kelola Data Poliklinik

Tampilan kelola data pasien poliklinik diakses oleh dokter yang mengelola poliklinik, pada halaman ini data dari pendaftaran pasien akan dikelola oleh dokter poli gigi untuk menginput diagnosis pengobatan dan menentukan tindakan yang akan dilakukan selanjutnya berupa rawat inap, rawat jalan, atau rujukan ke Rumah Sakit Lanjutan [Gambar 19]

Gambar 19. Tampilan Kelola Data Pasien Poliklinik

Kelola riwayat pasien menampilkan riwayat pasien yang pernah mendaftar dan sudah didiagnosis oleh dokter pengampu, dan admin menggunakannya sebagai pelaporan untuk puskesmas. Halaman riwayat dapat dicetak dengan generate excel agar lebih fleksibel digunakan, halaman yang dicetak dengan klik button 'cetak' dapat dipilih dan difilter berdasarkan kebutuhan operasional [Gambar 20].

No.	Nama Pasien	Layanan	Status	Kelola
1	Natalia Matana	kia/kb	status	Cetak
2	Mawar Melinda	poli gigi	rawat jalan	Cetak
3	Markus Aban	poli gigi	rawat jalan	Cetak
4	Ivan Setiawan	poli umum	rawat jalan	Cetak

Gambar 20. Tampilan Kelola Riwayat Pasien

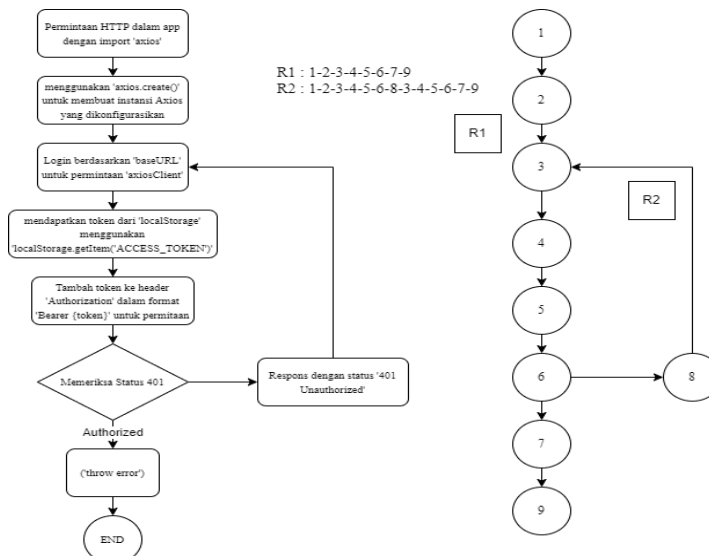
C. Evaluation

White-Box testing dengan Basic Path memungkinkan perancang test case untuk menghasilkan pengukuran kompleksitas dari perancangan procedural dan menggunakan pengukuran ini sebagai perkiraan untuk menguraikan jalur dasar eksekusi. Pada proses pengujian, kode program akan dibedakan berdasarkan route-nya saja, lalu kemudian kode program juga akan dilihat berdasarkan fungsi dan kompleksitasnya. Route Axios Client API merupakan route GraphQL sebagai perantara permintaan Next.js ke back-end dari Laravel.

Tabel 1. Route Axios Client API

Code program	Node
import axios from "axios";	1
const axiosClient=axios.create({	2
baseUrl:`http://127.0.0.1:8000/api`	3
});	4
axiosClient.interceptors.request.use((config)=>{	5
const token = localStorage.getItem('ACCESS_TOKEN')	6
config.headers.Authorization=`Bearer \${token}`	8
return config;	
});	
axiosClient.interceptors.request.use((response)=>{	
return response;	
try {	
const {response}=error;	
if(response.status===401){	

console.log(response.status) localStorage.removeItem('ACCESS_TOKEN')}	
throw error; }	7
export default axiosClient;	9



Gambar 20. Flowchart dan Flowgraph Route Axios Client API

Berdasarkan flow graph pada Gambar 20 diketahui bahwa jumlah edge (E) = 9 yang merupakan garis yang menghubungkan node, jumlah node (N) = 9 yang menggambarkan suatu aktifitas, dan jumlah region (R) = 2 yang menandakan suatu area dalam flow graph, sehingga jika dimasukkan ke dalam rumus perhitungan cyclomatic complexity pada Tabel 2, maka akan menghasilkan sebagai berikut.

Tabel 2. Cyclomatic Complexity Route Axios Client API

Cyclomatic Complexity	
$V(G) = E - N + 2$	$V(G) = R$
$V(G) = 9 - 9 + 2$	$V(G) = 2$
$V(G) = 2$	

Dari hasil perhitungan pada Tabel 2, didapatkan 2 jalur Independent Path pada Tabel 3 Dibawah ini, yaitu :

Tabel 3. Independent Path Route Axios Client API

R	Route	Skenario
R1	1-2-3-4-5-6-7-9	Berhasil 'Authorized' menghubungkan frontend dan backend.
R1	1-2-3-4-5-6-7-8-3-4-5-6-7-9	Server merespons dengan status '401 Unauthorized'

D. Communication

Proses ini melibatkan penyampaian informasi secara jelas dan efektif melalui presentasi dan demonstrasi program yang akan memudahkan khalayak public dan Puskesmas Rawat Inap Jungkat. Selain itu, melibatkan umpan balik dari rekan sejawat dan audiens dapat meningkatkan kejelasan dan dampak dari hasil penelitian, serta memperluas penerimaan dan penerapan temuan di berbagai konteks.

Penerapan Next.js dan GraphQL memberikan kebaharuan dalam pengembangan sistem, karena memungkinkan pengambilan data yang spesifik dan efisien, memanfaatkan SSR untuk mempercepat pemuatan halaman serta meningkatkan SEO, sehingga sistem menjadi lebih responsif, hemat sumber daya, dan relevan untuk kebutuhan sistem layanan puskesmas yang sebelumnya tidak efisien.

4. KESIMPULAN

Berdasarkan penelitian dan hasil Penerapan Next.js dan GraphQL dalam Pengembangan Mobile Web Sistem Informasi Manajemen Puskesmas pada Puskesmas Rawat Inap Jungkat, disimpulkan bahwa Puskesmas tersebut mengalami kendala dalam pengelolaan data, proses pendaftaran yang berulang-ulang di setiap poliklinik, menyebabkan redudansi data dan waktu tunggu yang lama. Untuk mengatasi hal ini, dihasilkan sebuah sistem Mobile Web terintegrasi melalui pengembangan mobile web yang responsive dengan menerapkan Framework Next.js dan Laravel dan Protokol GraphQL. Penerapan SSR dari Next.js dan GraphQL yang terintegrasi dapat meningkatkan kualitas pelayanan kesehatan di puskesmas secara signifikan, sehingga petugas medis dapat mengembangkan data tersebut tanpa perlu menginput ulang, sistem menjadi lebih responsif, dan relevan untuk kebutuhan sistem layanan puskesmas yang sebelumnya tidak efisien. Sistem ini memerlukan pemeliharaan secara berkesinambungan, mengingat sistem puskesmas mencakup ruang lingkup yang besar, sehingga diperlukan pengembangan lebih lanjut untuk mencapai optimalisasi kinerja. Diharapkan Sistem Informasi Manajemen Puskesmas dapat terus dikembangkan dengan fitur-fitur yang lebih baik agar dapat diaplikasikan tidak hanya di Puskesmas Rawat Inap Jungkat, tetapi juga di layanan kesehatan lainnya, sehingga dapat meningkatkan efisiensi pelayanan secara keseluruhan.

UCAPAN TERIMA KASIH

Ucapan terima kasih untuk penelitian ini disampaikan kepada dosen yang membimbing sehingga penulisan ini dapat ditulis dengan baik, Puskesmas Rawat Inap Jungkat yang telah memberikan izin dan dukungan selama penelitian ini berlangsung. Ketersediaan Puskesmas Rawat Inap Jungkat untuk memfasilitasi upaya penelitian ini sangat dihargai, dan kontribusi mereka sangat berharga bagi penyelesaian penelitian ini. Penulis menyadari bahwa, karena keterbatasan penulis sendiri, ada banyak kekurangan dalam penyusunan artikel jurnal ini. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun untuk membantu penulis memperbaiki artikel jurnal berikutnya.

REFERENSI

- [1] Hariyoko, Y., Jehaut, Y. D., & Susiantoro, A. (2021). Efektivitas pelayanan kesehatan masyarakat oleh puskesmas di kabupaten manggarai, Jurnal Good Governance, No. 2, Vol.17, 169-178.
- [2] Rewah, D. R., Sambiran, S., & Pangemanan, F. (2020). Efektivitas Penerapan Sistem Informasi Manajemen Puskesmas (SIMPUS) di Kota Manado (Studi Puskesmas Bahu). Jurnal Eksekutif, 2(5).
- [3] Gow, V. K., Kosasi, S., Wijaya, T., David, D., & Laipaka, R. (2021). Penerapan Teknologi Restful Web Service Aplikasi Mobile Jasa Make Up. Information System Journal (INFOSYS), 6(2), 127-140.
- [4] McFedries, P. (201). Web Coding & Development All-in-One For Dummies. John Wiley & Sons, Canada.
- [5] Widayanti, T. (2023). Perencanaan Enterprise Architecture Pada Rumah Sakit Ibu Dan Anak Mulia Menggunakan Togaf Adm. Jikom: Jurnal Informatika dan Komputer, 13(2), 26-34.
- [6] Putra, M. Y. (2020). Responsive Web Design Menggunakan Bootstrap Dalam Merancang Layout Website. Information System For Educators And Professionals: Journal of Information System, 5(1), 61-70.
- [7] Hermanto, H. (2023) Pengembangan sistem penyewaan mobil secara online. Jurnal Teknologi Informasi dan Komunikasi Bisnis, 6(1), 34-47.
- [8] Porcello, E., & Banks, A. (2018). Learning GraphQL: declarative data fetching for modern web apps. O'Reilly Media, Inc, USA.

- [9] Nur, A., Pasaribu, A. F. O., & Romlah, S. (2021). Sistem Informasi Manajemen Pendaftaran Rekam Medik Pasien Berbasis Mobile (Studi Kasus: Klinik Bersalin Nurhasanah). *Jurnal Teknologi dan Sistem Informasi*, 2(2), 109-115.
- [10] Halimah, H., Wasilah, W., Marshella, O., & Saleh, S. (2023, August). Sistem Informasi Manajemen Puskesmas (Simpus) Pada Puskesmas BIHA Pesisir Barat Lampung. In *Prosiding Seminar Nasional Darmajaya* (Vol. 1, pp. 152-160).
- [11] Putra, D. M., Yasli, D. Z., Leonard, D., & Yulia, Y. (2020). Penerapan Sistem Informasi Manajemen Puskesmas (SIM-PUS) Pada Unit Rekam Medis Dan Informasi Kesehatan Di Puskesmas Lubuk Buaya Kota Padang. *J. Abdimas Saintika*, 2(2), 67-72.
- [12] Vom Brocke, J., Hevner, A., & Maedche, A. (2020). *Design science research: Cases*. Springer Nature Switzerland AG 2020, Switzerland.
- [13] Katiyar, V. P., & Patel, S. (2019). White-Box Testing Technique For Finding Defects. *Glob, J. Res. Anal*, No.7, Vol.8, 83-85.
- [14] Rachman, A. F., Shalahudin, M. I., & Radifa, F. A. (2023). Implementasi Website Full-Stack Menggunakan Teknologi Next. Js, React, Dan Sanity. *Jurnal GENIEMAS: Generasi Teknologi Melayani Masyarakat*, 19-22.
- [15] Bagui, S.S., & Richard, W.E. (2023). *Database Design Using Entity-Relationship Diagrams*. Boca Ratn : CRC Press, Edisi Ketiga.
- [16] Sonata, F. (2019). Pemanfaatan UML (Unified Modeling Language) dalam perancangan sistem informasi e-commerce jenis customer-to-customer. *Jurnal Komunika: Jurnal Komunikasi, Media Dan Informatika*, 8(1), 22-31.