Perancangan *Website* Untuk Layanan Kesehatan Pada Klinik Citra Sehat Bandung Menggunakan Laravel

1st Ihsan Furqani
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia
ihsanfurqani@student.telkomuniversity.ac.id

2nd Asep Mulyana
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia
asepmulyana@telkomuniversity.ac.id

3rd Hafidudin
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia
hafidudin@telkomuniversity.ac.id

Abstrak—Klinik merupakan institusi pelayanan publik yang bergerak di bidang pelayanan kesehatan. Saat ini masih banyak klinik yang secara manual mencatat kegiatan operasional sehari-hari menggunakan kertas atau buku. Arsip berbagai data klinik akan menumpuk seperti gunung dan membutuhkan ruang penyimpanan tambahan agar catatan kertas tidak hilang atau rusak. Dari permasalahan tersebut, maka diperlukan sebuah tempat penyimpanan yang dapat menampung dan mengelolah data, sehingga mempermudah *staff* dalam proses pengelolahan data. Pembangunan *website* menggunakan *Framework* Laravel dan *database* MySQL dengan menerapkan konsep MVC dalam teknik pemogramannya serta menggunakan metode *pagination*. Pada Proyek akhir ini menghasilkan *website* yang dapat mempermudah dalam proses pencarian data menjadi efisien. Dari hasil pengujian, *website* telah behasil dibuat dan diimplementasikan pada Klinik Citra Sehat Sadang untuk membantu pelayanan kesehatan pada Klinik. Pada pengujian fungsionalitas dinyatakan berhasil dengan aplikasi dapat bekerja dengan baik serta mendapatkan tingkat keberhasilan 100%. Pada pengujian *delay* mendapatkan total rata-rata *delay* sebesar 1.8 detik. Pada pengujian *lighthouse* mendapatkan *Performance* sebesar 95, *Accessibility* sebesar 82, *Best Practices* sebesar 92 dan *SEO* sebesar 80 serta berdasarkan pengujian kuesioner menunjukkan bahwa *website* dapat membantu pelayanan kesehatan pada klinik serta mendapatkan respon yang positif dari *staff* dan pasien Klinik Citra Sehat Sadang dengan angka 90.54% menyatakan puas.

Kata kunci—*website*, laravel, *mysql, pagination, mvc*.

Abstract—Clinic is a public service institution that is engaged in health services. Currently, there are still many clinics that manually record daily operational activities using paper or books. Archives of various clinic data will pile up like mountains and require additional storage space so that paper records are not lost or damaged. From these problems, we need a storage area that can accommodate and manage data, making it easier for staff in the data management process. Website development using the Laravel Framework and MySQL database by applying the MVC concept in programming techniques and using the pagination method. This final project produces a website that can simplify the data search process to be efficient. From the test results, the website has been successfully created and implemented at the Citra Sehat Sadang Clinic to assist health services at the Clinic. In testing the functionality is declared successful with the application can work well and get a 100% success rate. In the delay test, the total average delay is 1.8 seconds. In the lighthouse test, it got Performance of 95, Accessibility of 82, Best Practices of 92 and SEO of 80 and based on questionnaire testing it shows that the website can help health services at the clinic and get a positive response from staff and patients at the Citra Sehat Sadang Clinic with a figure of 90.54 % expressed satisfaction.

Keyword—website, laravel, mysql, pagination.

I. PENDAHULUAN

Pasal 1 ayat 1 Undang-Undang Nomor 25 Tahun 2009 tentang Pelayanan Publik mendefinisikan bahwa pelayanan publik adalah suatu kegiatan atau rangkaian kegiatan yang dilakukan untuk memenuhi kebutuhan pelayanan atas barang, jasa atau pelayanan administratif yang diberikan oleh peraturan perundang-undangan kepada setiap warga negara dan penduduk oleh pelayanan publik. Klinik merupakan sebuah institusi pelayanan publik yang bergerak dalam bidang jasa kesehatan. Untuk meningkatkan kualitas layanan kesehatan, perlu dilakukan perbaikan dalam pengelolaan data operasional pada klinik tersebut [1].

Website adalah kumpulan halaman-halaman yang berisi informasi yang disimpan oleh internet yang bisa diakses atau dilihat melalui jaringan internet pada perangkat-perangkat yang bisa mengakses internet itu sendiri seperti komputer dan *smartphone*. Pengguna *website* pada sebuah institusi akan mempermudah dalam pengenalan, promosi dan pesebaran informasi itu sendiri.

Pada peneliatan sebelumnya yang dilakukan oleh Hartati Sepediyanti pada tahun 2018 dengan judul Aplikasi Penjualan disertai Konsultasi (Studi

Kasus: Apotek Klinik Famili Sehat). Pada penelitian tersebut aplikasi dibuat dalam basis *website* dengan memiliki fitur yaitu fitur konsultasi dan memiliki fitur pembelian obat secara online melalui aplikasi tanpa mengantri. Data pada aplikasi dapat dicetak dalam bentuk format file xlsx (*excel*) [2].

Pada peneliatan sebelumnya yang dilakukan oleh Muhammad Fajar Nugroho Alam pada tahun 2022 dengan judul Pembuatan Sistem Informasi Kebidanan (*E-Tocologist*) Untuk Pelayanan *Monitoring* Kesehatan Pada Ibu Dan Anak Bebasis *Website* Dan Aplikasi *Mobile* Di Klinik Rohaeni, S.St. Pada penelitian tersebut terdapat 2 *platform* yaitu *website* yang digunakan untuk informasi pelayanan dari klinik serta pengolahan data pemasukan administrasi dan aplikasi berbasis *mobile* yang digunakan untuk *monitoring* kesehatan ibu dan anak. Pada *website* menggunakan *framework* laravel dan pada aplikasi menggunakan bahasa *react native* [3].

Pada peneliatan sebelumnya yang dilakukan oleh Asri Amaliza Fathia Matusea pada tahun 2021 dengan judul Rancang Bangun Aplikasi Pendaftaran Pasien Online Dan Pemeriksaan Dokter Di Klinik Pengobatan Berbasis *Web*. Pada penelitian tersebut aplikasi dibuat berbasis *web* dengan menggunakan bahasa pemrograman PHP dan menggunakan database MySQL. Fitur yang terdapat pada aplikasi yaitu pendaftaran pasien online, mendapatkan antrian online, melakukan pemeriksaan online dan dapat melihat riwayat pemeriksaan secara online melalui aplikasi [4].

Berdasarkan ketiga penelitian diatas, proyek akhir ini merancang sebuah *website* administrasi dan rekam medis pasien yang dapat mempermudah proses pengolahan data. *Website* dibangun dengan menggunakan *Framework* Laravel, *database* MySQL yang dilengkapi dengan sistem pendaftaran melalui *kiosk* dan antrian *audio visual*.

II. DASAR TEORI

A. Rekam Medis

Rekam Medis adalah berkas yang berisikan catatan dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan dan pelayanan lain yang telah diberikan kepada pasien dan dilanjutkan dengan penanganan berkas rekam medis yang meliputi penyelenggara penyimpanan serta pengeluaran berkas dari tempat penyimpanan untuk melayani permintaan atau peminjaman apabila dari pasien atau untuk keperluan. Peran rekam medis sangat penting menunjang tata tertib pengadministrasi klinik. Kualitas rekam medis merupakan cerminan baik buruknya pelayanan klinik. Rekam medis dapat dijadikan bukti kasus yang melanggar hukum misalnya malpraktik ke pengadilan. Isi rekam medis harus dijaga kerahasiannya oleh dokter, tenaga kesehatan, pengelola pelayanan kesehatan [5].

B. *Website*

Website merupakan kumpulan halaman yang dapat menampilkan data-data dalam bentuk teks, gambar, *video*, suara maupun gabungan semuanya. Rangkaian bangunan yang saling berhubungan, baik statis maupun dinamis, masing-masing dihubungkan oleh jaringan halaman (*Hyperlink*). Sebuah *website* dikatakan statis jika konten informasinya tetap sama, jarang berubah dan konten informasi hanya berasal dari arah yang sama dari pemilik *website*. Jika konten informasi *website* selalu berubah, *website* tersebut bersifat dinamis dikarenakan konten informasi merupakan interaksi dua arah antara pemilik *website* dan pengguna [6].

C. MySQL

MySQL merupakan sebuah *Database Management System* (DBMS) menggunakan perintah SQL (*Sturctured query Language*) yang banyak digunakan dalam pembuatan aplikasi berbasis *website*. MySQL termasuk dalam RDBMS (*Relational Database Management System*) sehingga menggunakan tabel, kolom, baris di dalam struktur databasenya, sehingga menjadi pengubung antara perangkat lunak dan database servernya [7].

D. *Pretext Hyper Processor* (PHP)

PHP adalah kependekan dari *Pretext Hyper-Processor* yang dibagun oleh Rasmus Lerdorf pada tahun 1994 [8]. PHP adalah bahasa pemrograman untuk membangun *website* dinamis dan berjalan di sisi server, sehingga PHP juga dikenal sebagai Bahasa *Server Side Scripting*. Artinya dalam setiap/menjalankan PHP, diperlukan *web server* lalu semua sintaks yang diberikan akan sepenuhnya dieksekusi di server dan hanya hasilnya yang akan dikirim ke *browser*. Beberapa keunggulan PHP antara lain kemampuan untuk membuat situs interaktif dengan forum diskusi, *guestbook* dan koneksi yang baik ke berbagai database seperti Oracle, MySQL, PostgreSQL dan lain-lain. PHP bersifat *open source* dan *cross platform* dengan dapat berjalan di sistem operasi Windows dan Linux. PHP juga dibangun sebagai modul pada server *web* apache dan *binary* yang dapat dijalankan sebagai CGI.

E. Laragon

Laragon adalah perangkat lunak yang bersifat *open source* yang mendukung banyak sistem operasi serta berfungsi sebagai server mandiri atau *localhost*. Laragon menyediakan banyak fitur, layanan dan *tools* yang diantaranya Apache, MySQL, PHP Server, PhpMyAdmin, Redis, Composer, Cmder, Xdebug, Memchaced dan Laravel. Laragon memiliki arsitektur modern yang cocok untuk dipakai mengembangkan aplikasi *web modern* dan juga penggunaan Apache & Nginx [9].

F. Laravel

Laravel adalah *framework* aplikasi *web* dengan sintaks yang ekspresif dan elegan yang dapat memaksimalkan penggunaan PHP di dalam proses pengembangan. Laravel adalah pengembangan *website* berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu [10]. Dimana laravel sendiri memiliki berbagai macam fitur yang berguna dalam proses pemograman *web* diantaranya *Model View Controller*, *Templeting Engine*, *Artisan console*, *Authentication* & *authorization*.

G. *Pagination*

Pagination merupakan sebuah fitur yang digunakan untuk membatasi tampilan data agar tidak terlalu panjang dan lebih rapi [11]. Pola desain antarmuka pengguna ini dirancang untuk membantu pengunjung situs *web* agar tidak kewalahan oleh sejumlah besar informasi yang cocok dari satu halaman ke halaman lainnya.

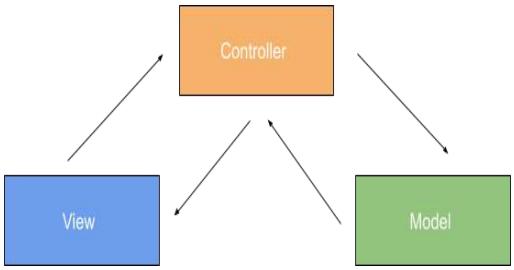
Pagination seperti buku, tidak semua informasi bisa dimuat dalam satu halaman. Maka, sebuah buku yang berisi sejumlah besar teks atau informasi terdiri dari banyak halaman. Untuk memudahkan pembaca, sebuah buku dengan banyak halaman akan menyertakan halaman daftar isi untuk membantu pembaca menemukan halaman yang diharapkan.

H. *Model View Controller*

Model View Controller (MVC) merupakan konsep yang cukup populer dalam pengembangan *website*, dimulai dengan bahasa pemrograman *small talk*, *Model View Controller* memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi, seperti memanipulasi data, *user interface* dan bagian yang mengontrol aplikasi [12]. Berikut tiga komponen dalam pembagunan MVC yaitu:

1. *Model*, bagian yang berhubungan langsung dengan database untuk memanipulasi data (*update, delete, insert, search*).
2. *View*, bagian yang akan menyajikan informasi kepada pengguna.
3. *Controller*, bagian yang menghubungkan *model* dan *view*, *controller* berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang akan diperoleh oleh aplikasi.

Dengan konsep MVC, *website* seolah-olah memiliki bagian-bagian tersendiri yang dapat dikembangkan secara individual. Alhasil, proses pembuatan *website* bisa lebih cepat selesai karena *developer* akan lebih fokus pada satu bagian pekerjaan. Karena dianggap efektif, konsep *Model View Controller* banyak digunakan di berbagai *framework*. Misalnya pada Laravel, CodeIgniter, Symfony, Yii dan Zend sudah menggunakan konsep ini:



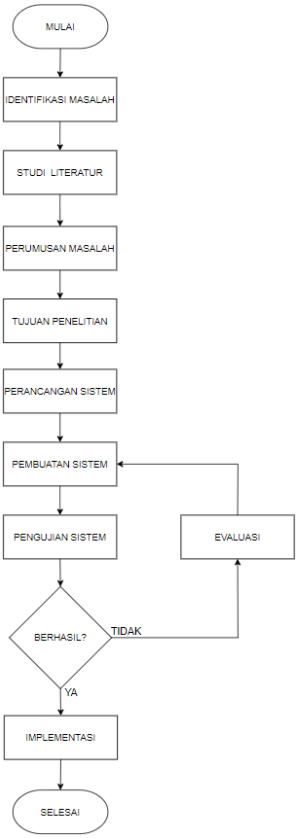
Berikut alur kerja dari *Model View Controller* dapat dilihat pada Gambar 1 :

1. Bagian *View* akan melakukan *request* informasi untuk bisa ditampilkan kepada pengguna.
2. *Request* tersebut kemudian diambil oleh *controller* dan diserahkan pada bagian *model* untuk diproses.
3. *Model* akan mengolah dan mencari data informasi tersebut di dalam database.
4. *Model* memberikan kembali pada *controller* untuk ditampilkan hasilnya di *view*.
5. *Controller* mengambil hasil olahan yang dilakukan di bagian *model* dan menatanya di bagian *view*.

III. PERANCANGAN *WEBSITE*

A. Deskripsi Proyek Akhir

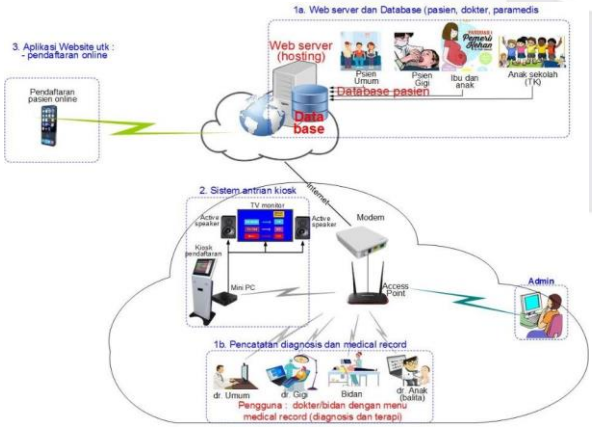
Pada Proyek Akhir ini terdapat beberapa tahapan yang akan dilakukan pada perancangan *website* untuk layanan kesehatan pada klinik citra sehat Bandung menggunakan laravel. Pada Gambar 2 merupakan *flowchart* tahapan pengerjaan pada proyek akhir ini.



Gambar 2 *Flowchart* Tahapan Pengerjaan Proyek Akhir

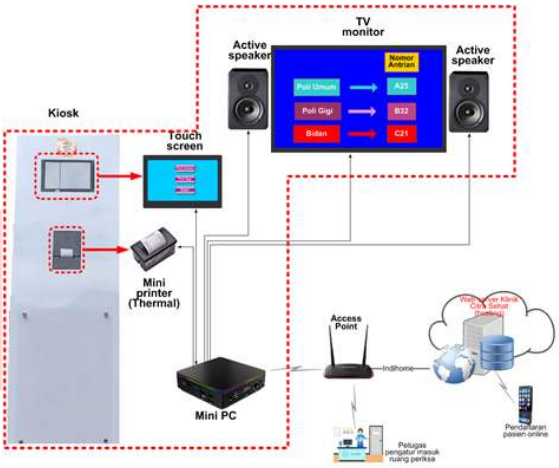
B. Blok Diagram Sistem

Pada blok diagram sistem menjelaskan cara kerja dari perancangan *website*. Blok diagram sistem adalah bagian beserta alur kerja sistem dengan tujuan untuk menerangkan cara kerja dan alur sistem secara garis besar yang berupa gambar dengan tujuan agar sebuah sistem dapat lebih dipahami. Gambar blok diagram sistem *website* sebagai berikut:



Gambar 3 Blok Diagram Sistem

Pada Gambar 3 merupakan blok diagram sistem keseluruhan yang menunjukan bahwa *website* akan saling terintegrasi dengan database MySQL, saat pasien melakukan registrasi melalui *website* atau kios data akan tersimpan sesuai dengan database dokter di masing-masing bidang. Pada saat dokter memulai proses praktek, dokter akan menekan tombol *next* antrian untuk langsung mengambil data nomor antrian yang ada didalam *database* yang sudah diurutkan sesuai dengan waktu pendaftaran pasien. Kemudian data akan dikirim ke *mini pc* dan diteruskan ke layar *tv monitor* dan *speaker* akan mengeluarkan suara untuk proses pemanggilan nomor antrian pasien.



Gambar 4 Blok Diagram Kiosk

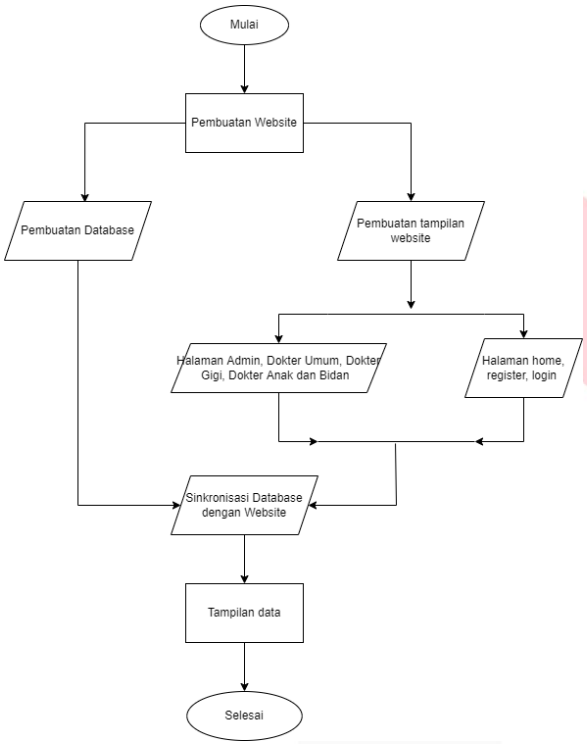
Pada Gambar 3 merupakan blok diagram kios yang terhubung dengan jaringan internet melalui *access point* yang terdiri dari beberapa perngkat yang diantaranya.

a. *Touch Screen* merupakan perangkat *input* komputer yang bekerja dengan cara menyentuh layar tersebut. *User* atau pengguna dapat mengoperasikan sistem komputer dengan menyentuh gambar atau tulisan yang tersedia pada layar.
b. *Monitor* merupakan perangkat keras komputer yang digunakan untuk menampilkan hasil proses dari komputer dalam bentuk teks, gambar dan video secara visual.
c. *Mini Printer* adalah perangkat eksternal komputer yang dapat menampilkan data komputer dalam bentuk cetakan, data tersebut bisa berupa teks atau gambar yang dicetak kedalam media kertas, kain dan lain sebagainya.
d. *Mini PC* adalah komputer yang dirancang dalam ukuran kecil dan telah dilengkapi teknologi *legacy-free* terbaru serta konektivitas modern.
e. *Speaker* adalah perangkat keras *output* yang berfungsi mengeluarkan hasil pemrosesan oleh CPU yang berupa *audio* atau suara.

C. Perancangan Sistem

Pada Proyek Akhir akan dirancang sebuah *website* administrasi dan rekam medis pasien

menggunakan *Software Virtual Studio Code* dengan bahasa pemrograman laravel dan akan menggunakan konsep *Model View Control* (MVC) dan metode *pagination*, yang dimana konsep *Model View Control* (MVC) dapat mempermudah dalam proses *design website* yang akan dibuat. Sedangkan *pagination* akan membagi data *website* seperti halaman-halaman terpisah sehingga dapat mempermudah dalam mencari informasi yang diperlukan:



Gambar 5 *Flowchart* Tahapan Perancangan *Website*

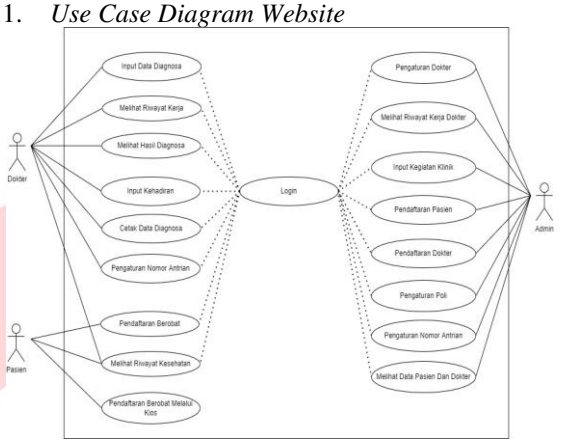
Berdasarkan Gambar 5 telah dilakukan perancangan dengan tahap-tahap sebagai berikut:

1. Pembuatan *Website*, dalam proses pertama pembuatan *website* adalah menentukan bahasa pemrograman yang ingin digunakan dan menggunakan *software Visual Studio Code*.
2. Membuat pendukung *website* yaitu dengan menentukan perancangan database dan pembuatan tampilan *website*. Database yang digunakan adalah MySQL. Pada pembuatan tampilan *website* menggunakan bahasa pemrograman HTML, CSS, PHP dan *Framework* Laravel.
3. Sinkronisasi *website*, pada tahap ini perlu melakukan sinkronisasi data antara database dengan *website* agar data yang telah diolah pada database dapat ditampilkan pada halaman *website*.
4. Tampilan data, pada tahap ini data yang sudah diolah dan tersimpan pada database akan diambil dan ditampilkan pada *website*.

D. Pemodelan Sistem

Pemodelan sistem digunakan untuk menjelaskan dan menggambarkan seluruh alur sistem *website* yang dibuat. Pada pemodelan sistem menggunakan metode UML (*Unitified Model Language*). UML (*Unitified Model Language*) adalah suatu metode dalam pemodelan secara visual yang digunakan untuk menggambarkan alur perancangan sistem melalui beberapa diagram seperti *Use Case Diagram* dan *Activity Diagram*.

1. *Use Case Diagram Website*

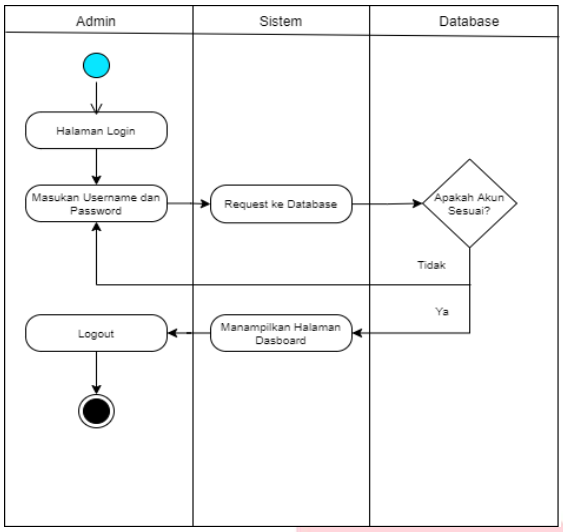


Gambar 6 *Use Case Diagram Website*

Pada Gambar 6 menjelaskan alur dari *website* yang digambarkan dalam *Use Case Diagram*. Adapun kebutuhan dari *website* ini adalah sebagai berikut:

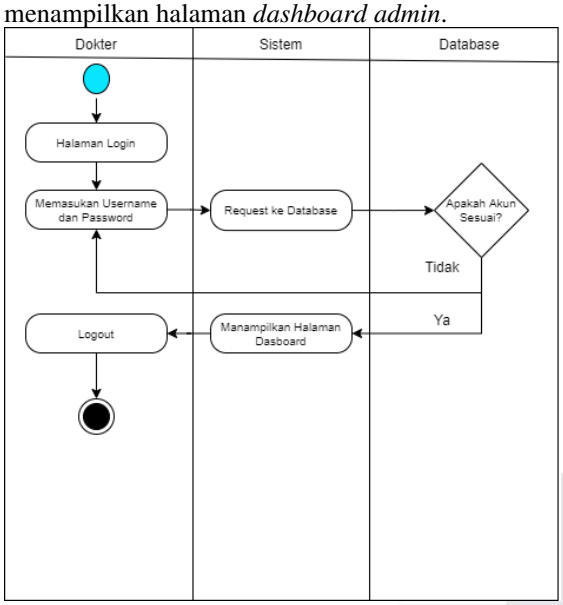
a. *Admin* dapat *login* sesuai dengan *username* dan *password* yang telah terdaftar. Setelah *login*, *admin* dapat melihat seluruh data *user*, mengelolah data pasien, mengelolah data administrasi, *input* data *user*, melihat riwayat kerja dan mengatur nomor antrian.
b. Dokter dan bidan harus melakukan registrasi terlebih dahulu, setelah melakukan registrasi, dokter dan bidan dapat *login* sesuai *username* dan *password* yang sudah terdaftar. Dokter dan bidan dapat melakukan memasukkan data diagnosa, melihat hasil diagnosa, mencetak hasil diagnosa, melihat riwayat kerja, mengatur nomor antrian dan memasukkan jadwal kerja.
c. Pasien dapat melakukan registrasi melalui *website* yang sudah disediakan dan dapat melakukan registrasi melalui kios yang telah disediakan oleh klinik secara langsung untuk mendapatkan nomor antrian.

2. *Activity Diagram Website*



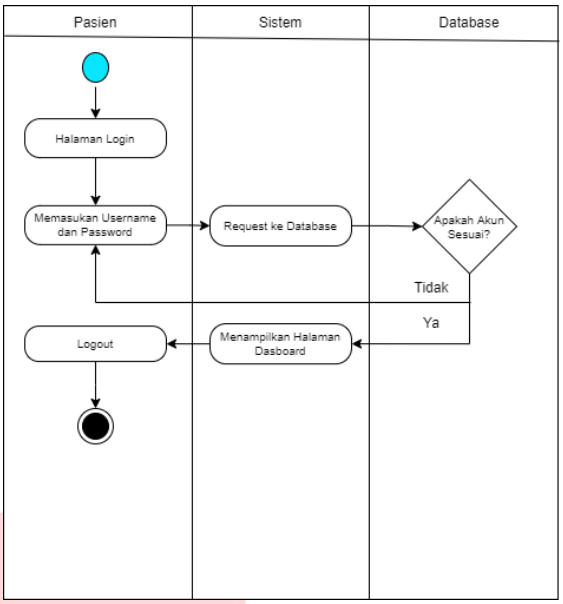
Gambar 7 *Activity Diagram Login Admin*

Pada Gambar 7 merupakan *activity diagram login admin* yang menjelaskan bahwa *admin* dapat melakukan *login* dengan memasukkan *username* dan *password* dengan benar. Apabila *username* dan *password* yang dimasukan benar maka akan menampilkan halaman *dashboard admin*.



Gambar 8 *Activity Diagram* Login Dokter

Pada Gambar 8 merupakan *activity diagram login* dokter yang menjelaskan bahwa dokter dapat melakukan *login* dengan memasukkan *username* dan *password* dengan benar. Apabila *username* dan *password* yang dimasukan benar maka akan menampilkan halaman *dashboard* dokter.

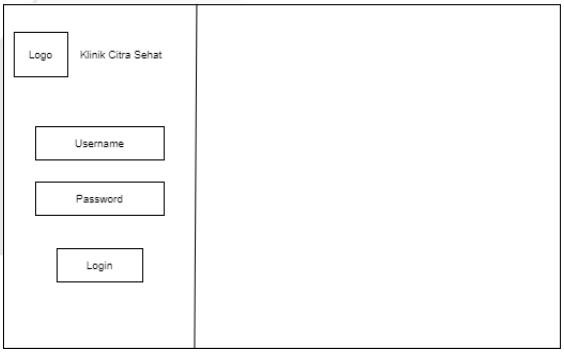


Gambar 9 *Activity Diagram* Login Pasien

Pada Gambar 9 merupakan *activity diagram login* pasien yang menjelaskan bahwa pasien dapat melakukan *login* dengan memasukkan *username* dan *password* dengan benar dan sudah terdaftar. Apabila *username* dan *password* yang dimasukan benar dan sudah terdaftar maka akan menampilkan halaman *dashboard* pasien.

E. Perancangan *Layout Website*

Pada pembuatan *layout website* proyek akhir ini dapat dibuat menggunakan bahasa pemrograman seperti PHP, HTML, CSS dan *Framework* Laravel yang didapat dari internet kemudian di *download*, lalu di *edit* sesuai keinginan dan kebutuhan. Berikut merupakan perancangan *layout* halaman *website* yang dibuat:

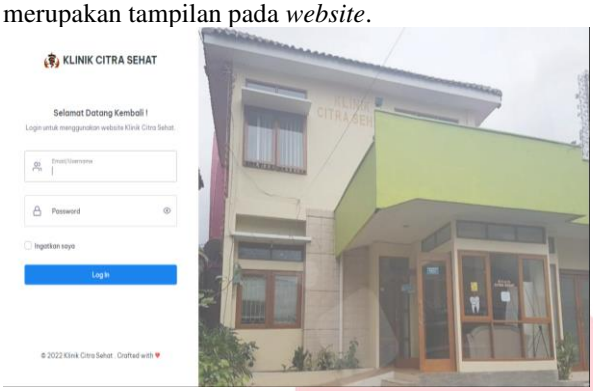


Gambar 10 *Layout* Login

Pada Gambar 10 merupakan perancangan *layout* pada halaman *login* yang dimana pada halaman tersebut *admin*, dokter dan pasien dapat melakukan *login* dengan memasukan *username*, *email* dan *password* yang sudah terdaftar dengan benar.

IV. PENGUJIAN SISTEM DAN ANALISA HASIL

Pada Proyek Akhir ini, telah berhasil dibuat perancangan *website* dan *website* diimplementasikan di Klinik Citra Sehat Sadang untuk membantu dan mempermudah proses pelayanan kesehatan. Berikut merupakan tampilan pada *website*.



Gambar 11 Hasil Tampilan Halaman *Login Website*

Pada Gambar 11 merupakan tampilan halaman *login* pada *website* yang digunakan untuk akun *admin*, dokter dan pasien yang telah terdaftar agar dapat mengakses dan menggunakan *website*.

A. Pengujian Fungsionalitas

Pengujian fungsionalitas dilakukan untuk mengetahui program yang dibuat apakah sudah berjalan dengan baik sesuai dengan rancangan yang telah dibuat atau tidak. Pengujian fungsionalitas dilakukan dengan cara menjalankan *website* dan mecoba semua fitur yang telah dibuat. Pengujian fungsionalitas tersebut ditunjukan pada Tabel 1:

Tabel 1 Pengujian Fungsionalitas

No	Nama Pengujian	Aksi	Hasil Pengujian	Keterangan
1.	Login (Admin)	Memasukkan *email, username* dan *password* yang telah terdaftar sebagai admin	Berhasil login dan berhasil masuk ke halaman *dashboard* admin	Berhasil 100%
2.	Login (Pasien)	Memasukkan *email, username* dan *password* yang telah terdaftar sebagai pasien	Berhasil login dan berhasil masuk ke halaman *dashboard* pasien	Berhasil 100%
3.	Login (Dokter)	Memasukkan *email, username* dan *password* yang telah terdaftar sebagai dokter	Berhasil login dan berhasil masuk ke halaman *dashboard* dokter	Berhasil 100%
4.	Logout (Admin)	Menekan Button Logout	Berhasil Logout	Berhasil 100%
5.	Logout (Pasien)	Menekan Button Logout	Berhasil Logout	Berhasil 100%
6.	Logout (Dokter)	Menekan Button Logout	Berhasil Logout	Berhasil 100%
7.	Menu Pendaftaran Akun Dokter Sebagai Admin	Mendaftarkan akun dokter serta input data diri dokter, bidang keahlian dan jam kerja dokter	Berhasil mendaftarkan akun dokter serta berhasil input data diri dokter, bidang keahlian dan jam kerja dokter	Berhasil 100%
		Menampilkan semua daftar akun dokter yang telah terdaftar	Berhasil menampilkan semua akun dokter yang telah terdaftar	Berhasil 100%
8.	Menu Data Dokter Sebagai Admin	Menampilkan semua data dokter yang telah terdaftar	Berhasil menampilkan semua data dokter yang telah terdaftar	Berhasil 100%

9.	Menu Daftar Kehadiran Dokter Sebagai Admin	Menampilkan semua daftar kehadiran dokter	Berhasil menampilkan semua daftar kehadiran dokter	Berhasil 100%
10.	Menu Pengaturan Dokter Poli Sebagai Admin	Menambahkan poli, mengatur poli sesuai dengan keahlian dokter serta mengubah data poli dan menghapus data poli	Berhasil Menambahkan poli, mengatur poli sesuai dengan keahlian dokter serta mengubah data poli dan menghapus data poli	Berhasil 100%
11.	Menu Pendaftaran Pasien Sebagai Admin	Mendaftarkan akun pasien dengan cara input data diri pasien	Berhasil Mendaftarkan akun pasien	Berhasil 100%
9.	Menu Data Pasien Sebagai Admin	Menampilkan semua daftar akun pasien yang telah terdaftar	Berhasil menampilkan semua daftar akun pasien yang telah terdaftar	Berhasil 100%
10.	Menu Pengaturan Master Poli Sebagai Admin	Menambahkan poli dengan memasukkan nama poli dan deskripsi poli serta mengubah poli dan menghapus poli	Berhasil Menambahkan poli dengan memasukkan nama poli dan deskripsi poli serta mengubah poli dan menghapus poli	Berhasil 100%
11.	Menu Pengaturan Antrian Sebagai Admin	Menampilkan nomor antrian sesuai poli, mengubah status antrian dan reset antrian sesuai poli	Berhasil Menampilkan nomor antrian sesuai poli, mengubah status antrian dan reset antrian sesuai poli	Berhasil 100%
12.	Menu Kegiatan Sebagai Admin	Menambahkan jadwal kegiatan dengan input nama kegiatan, deskripsi kegiatan dan gambar kegiatan serta menampilkan daftar kegiatan	Berhasil Menambahkan jadwal kegiatan dengan input nama kegiatan, deskripsi kegiatan dan gambar kegiatan serta menampilkan seluruh daftar kegiatan	Berhasil 100%
13.	Menu Absensi Dokter Sebagai Dokter	Menekan button Clock In untuk absen masuk dokter dan button Clock Out untuk absen keluar dokter	Berhasil absen masuk dan absen keluar pada dokter	Berhasil 100%
		Menampilkan daftar kehadiran dokter	Berhasil menampilkan daftar kehadiran dokter	Berhasil 100%
14.	Menu Riwayat Kesehatan Pasien Sebagai Dokter	Menampilkan daftar riwayat kesehatan pasien	Berhasil menampilkan daftar riwayat kesehatan pasien	Berhasil 100%

15.	Menu Daftar Berobat Sebagai Pasien	Menampilkan daftar poli dan memilih poli dengan menekan button poli yang tersedia	Berhasil menampilkan daftar poli dan berhasil memilih poli	Berhasil 100%
		Menampilkan data diri pasien dan input keluhan pasien	Berhasil menampilkan data diri pasien dan input keluhan pasien	Berhasil 100%

Berdasarkan Tabel 1 dapat dilihat bahwa setiap pengujian fungsionalitas yang telah dilakukan dengan menguji semua fitur yang terdapat pada *website*. Pada pengujian fungsionalitas dapat dinyatakan berhasil dan memiliki rata-rata tingkat keberhasilan 100%.

B. Pengujian *Delay*

Pengujian *delay* dilakukan agar dapat mengetahui rata-rata waktu yang diperlukan untuk mengirim dan menerima data dalam satuan detik dari *website* ke database. Pada saat pengujian, digunakan koneksi wi-fi Indihome dengan bandwidth 20Mbps.

Tabel 2 Pengujian *Delay Login* Pada Admin

Percobaan Ke-	Pengujian	Keterangan	Delay (Detik)
1.	Login Sebagai Admin	Berhasil Login Sebagai Admin	1.6
2.	Login Sebagai Admin	Berhasil Login Sebagai Admin	1.4
3.	Login Sebagai Admin	Berhasil Login Sebagai Admin	1.7
4.	Login Sebagai Admin	Berhasil Login Sebagai Admin	1.5
5.	Login Sebagai Admin	Berhasil Login Sebagai Admin	1.7
6.	Login Sebagai Admin	Berhasil Login Sebagai Admin	2.5
7.	Login Sebagai Admin	Berhasil Login Sebagai Admin	1.9
8.	Login Sebagai Admin	Berhasil Login Sebagai Admin	1.3
9.	Login Sebagai Admin	Berhasil Login Sebagai Admin	1.4
10.	Login Sebagai Admin	Berhasil Login Sebagai Admin	2.4
Rata-rata Delay			1.74 Detik

Tabel 3 Pengujian *Delay Login* Pada Dokter

Percobaan Ke-	Pengujian	Keterangan	Delay (Detik)
1.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	1.8
2.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	1.2
3.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	2.1
4.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	1.9
5.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	1.3

6.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	1.7
7.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	2.0
8.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	1.4
9.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	2.3
10.	Login Sebagai Dokter	Berhasil Login Sebagai Dokter	1.5
Rata-rata Delay			1.72 Detik

Tabel 4 Pengujian *Delay Login* Pada Pasien

Percobaan Ke-	Pengujian	Keterangan	Delay (Detik)
1.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	2.2
2.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	1.8
3.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	1.6
4.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	2.4
5.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	2.2
6.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	2.1
7.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	1.9
8.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	1.5
9.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	1.7
10.	Login Sebagai Pasien	Berhasil Login Sebagai Pasien	2.0
Rata-rata Delay			1.94 Detik

Tabel 5 Pengujian *Delay Clock In* dan *Clock Out*

Percobaan Ke-	Pengujian	Keterangan	Delay (Detik)
1.	Button Clock In	Berhasil Menekan Button Clock In	1.4
2.	Button Clock In	Berhasil Menekan Button Clock In	2.1
3.	Button Clock In	Berhasil Menekan Button Clock In	1.7
4.	Button Clock In	Berhasil Menekan Button Clock In	1.5
5.	Button Clock In	Berhasil Menekan Button Clock In	2.4
6.	Button Clock Out	Berhasil Menekan Button Clock Out	1.9
7.	Button Clock Out	Berhasil Menekan Button Clock Out	1.9

8.	Button Clock Out	Berhasil Menekan Button Clock Out	2.0
9.	Button Clock Out	Berhasil Menekan Button Clock Out	1.4
10.	Button Clock Out	Berhasil Menekan Button Clock Out	1.4
Rata-rata Delay			1.77 Detik

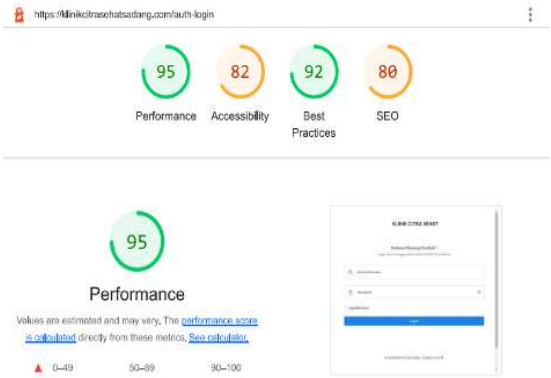
Tabel 6 Pengujian *Delay Button Next* Antrian

Percobaan Ke-	Pengujian	Keterangan	Delay (Detik)
1.	Button Next Antrian	Berhasil Menekan Button Next Antrian	1.7
2.	Button Next Antrian	Berhasil Menekan Button Next Antrian	1.3
3.	Button Next Antrian	Berhasil Menekan Button Next Antrian	2.3
4.	Button Next Antrian	Berhasil Menekan Button Next Antrian	2.1
5.	Button Next Antrian	Berhasil Menekan Button Next Antrian	1.6
6.	Button Next Antrian	Berhasil Menekan Button Next Antrian	1.9
7.	Button Next Antrian	Berhasil Menekan Button Next Antrian	1.5
8.	Button Next Antrian	Berhasil Menekan Button Next Antrian	2.3
9.	Button Next Antrian	Berhasil Menekan Button Next Antrian	1.6
10.	Button Next Antrian	Berhasil Menekan Button Next Antrian	2.0
Rata-rata Delay			1.83 Detik
Total Rata-rata Delay			1.8 Detik

Berdasarkan pada seluruh pengujian *delay* yang dilakukan terdapat masing-masing rata-rata *delay* yang berbeda beda. Pada pengujian *delay login* sebagai *admin* mendapatkan rata-rata *delay* sebesar 1.74 detik, pada pengujian *delay login* sebagai dokter mendapatkan rata-rata *delay* sebesar 1.72 detik, pada pengujian *delay login* sebagai pasien mendapatkan rata-rata delay sebesar 1.94 detik, pada pengujian *delay button clock in* dan *button clock out* mendapatkan rata-rata *delay* sebesar 1.77 detik dan pada pengujian *delay button next* antrian mendapatkan rata-rata *delay* sebesar 1.83 detik. Dari Semua pengujian dilakukan, terdapat total rata-rata *delay* sebesar 1.8 detik.

C. Pengujian *Lighthouse*

Pengujian *Lighthouse* merupakan sebuah *tools* yang digunakan untuk melakukan pengukuran dan audit untuk mengetahui *performance* kualitas dan aksesibilitas sebuah *website*. *Lighthouse* digunakan untuk menghitung performa dan kualitas sebuah *website* secara keseluruhan berdasarkan dengan data simulasi yang dilakukan. Berikut merupakan hasil dari pengujian *lighthouse*.



Gambar 12 Hasil Pengujian *Lighthouse*

Berdasarkan Gambar 12 telah dilakukan pengujian *lighthouse* dengan terdapat masing-masing hasil pengukuran yang diantaranya *performance*, *accessibility*, *Best Practices* dan SEO. Hasil pengukuran mendapatkan *performance* sebesar 95, *accessibility* sebesar 82, *Best Practices* sebesar 92 dan SEO sebesar 80. *Performance* merupakan suatu pengukuran seberapa cepat *website* dapat ditampilkan pada pengguna yang mengaksesnya. *Accessibility* merupakan suatu pengukuran website untuk mengetahui apakah *website* dapat diakses siapa saja tanpa adanya kendala. *Best Practices* merupakan untuk menganalisis pembangunan sebuah *website* dengan dinilainya penggunaan HTTPS untuk keamanan, *outbound link* yang aman dan jenis javasciprt yang terbaru. SEO merupakan suatu pengukuran unuk menilai sebuah konten terhadap aturan-aturan algoritma yang berlaku. Dari hasil pengukuran dan analisa pengujian *website* yang telah dilakukan pada *lighthouse* mendapatkan tingkat nilai yang tinggi dan sangat baik pada *performance*, *accessibility*, *Best Practices* dan SEO.

D. Pengujian Kuesioner

Pengujian kuesioner dilakukan untuk mengetahui apakah *website* yang telah dibuat sudah sesuai rancangan dan dapat diimplementasikan pada Klinik Citra Sehat Sadang. Dari data kuesioner yang telah diambil, didapatkan 18 responden yang diisi oleh *staff* dan pasien Klinik Citra Sehat Sadang. Berikut merupakan hasil pengujian kuesioner yang telah diambil:

Tabel 7 Pengujian Kuesioner

No	Pertanyaan	Presentase		
		Ya	Tidak	Mungkin
1.	Pada pandemi covid - 19 apakah penting untuk menjaga jarak ?	100%	0%	0%
2.	Untuk menghentikan penyebaran covid-19 apakah diperlukan website pada klinik ?	77.8 %	0%	22.2%
3.	Apakah tampilan website menarik ?	83.3 %	0%	16.7%
4.	Apakah website mudah untuk dioperasikan ?	94.4 %	0%	5.6%
5.	Apakah website berguna bagi pasien dan dokter ?	100%	0%	0%
6.	Apakah website mempermudah dalam pengolahan dan penyimpanan data ?	88.9 %	0%	11.1%
7.	Apakah website mempermudah dalam proses pengolahan informasi kesehatan ?	94.4 %	0%	5.6%
8.	Apakah website memudahkan dokter dalam proses pemeriksaan pasien ?	77.8 %	5.6%	16.7%
9.	Apakah website membantu pasien dalam proses pendaftaran berobat ?	94.4 %	0%	5.6%

10.	Apakah website membantu dalam proses antrian pasien ?	94.4 %	0%	5.6%

Pada Tabel 7 merupakan pengujian kuesioner yang diambil pada saat implementasi *website* dilakukan di Klinik Citra Sehat Sadang. Dari hasil tersebut dapat disimpulkan bahwa *website* dapat berguna bagi pasien dan dokter dalam membantu pelayanan kesehatan pada klinik serta mendapatkan respon yang positif dari *staff* dan pasien Klinik Citra Sehat Sadang.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil perancangan, pengujian dan analisa hasil yang telah dilakukan maka dapat diambil beberapa kesimpulan yaitu pada Proyek Akhir telah berhasil dibuat *website* pelayanan kesehatan pada Klinik Citra Sehat Sadang dengan menggunakan *framework* Laravel dan database MySQL. Pada pengujian fungsionalitas *website*, semua fitur yang telah dibuat dapat dijalankan serta dapat bekerja dengan baik dan pengujian fungsionalitas *website* dapat dinyatakan berhasil serta memiliki rata-rata tingkat keberhasilan 100%. Pada pengujian *delay* pada *website* mendapatkan total rata-rata *delay* sebesar 1.8 detik. Pada pengujian *lighthouse* untuk menganalisa *website* mendapatkan hasil *performance* sebesar 95, *accessibility* sebesar 82, *best practices* sebesar 92 dan SEO sebesar 80. Dari hasil analisa website pada lighthouse mendapatkan hasil nilai yang tinggi dan sangat baik. Pada pengujian kuesioner yang telah dilakukan dapat disimpulkan bahwa *website* dapat berguna bagi pasien dan dokter dalam membantu pelayanan kesehatan pada klinik serta mendapatkan respon yang positif dari *staff* dan pasien Klinik Citra Sehat Sadang..

B. Saran

Berdasarkan hasil Proyek Akhir ini, dapat disampaikan beberapa saran untuk pengembangan selanjutnya yaitu dengan menambahkan kreatifitas dengan membuat desain yang lebih menarik pada *website*, menambahkan fitur-fitur terbaru sesuai dengan kebutuhan pelayanan kesehatan pada Klinik Citra Sehat Sadang dan untuk pengembangan selanjutnya, diperlukannya pembuatan aplikasi *mobile* berbasis Android ataupun iOS agar mempermudah *user* atau pasien untuk melakukan pendaftaran berobat. Pada aplikasi pembuatan selanjutnya dapat saling terhubung dengan *website* yang telah dibuat.

REFERENSI

[1] Ferdiansyah Devy, "Penerapan Konsep *Model View Controller* Pada Rancang Bangun Sistem Informasi Klinik Kesehatan Berbasis *Web*," Jurnal Kajian Ilmiah, vol. 18, no. 2, Mei 2018.

[2] H. Sepdiyanti, A. P.K, and E. Hernawati, "Aplikasi Penjualan disertai Konsultasi (Studi Kasus: Apotek Klinik Famili Sehat)," *e-Proceeding Appl. Sci.*, vol. 4, no. 3, pp. 1748–1752, 2018, [Online]. Available: https://openlibrary.telkomuniversity.ac.id/pustaka/files/144523/jurnal_eproc/aplikasi-penjualan-disertai-konsultasi-berbasis-website-studi-kasus-apotek-klinik-famili-sehat-.pdf.

[3] M. Fajar, N. Alam, D. N. Ramadhan, and R. Tulloh, "Pelayanan Monitoring Kesehatan Pada Ibu Dan Anak Bebasis Website Dan Aplikasi Mobile Di Klinik Rohaeni , S . St Development of Midwife Information System (E-Tocologist) for Health Monitoring Services in Mother and Child Based on Website and Mobile Appli," vol. 8, no. 1, pp. 49–61, 2022.

[4] A. A. F. Matusea, "Rancang Bangun Aplikasi Pendaftaran Pasien Online Dan Pemeriksaan Dokter Di Klinik Pengobatan Berbasis Web," *J. Rekayasa Inf.*, vol. 10, no. 2, pp. 136–149, 2021.

[5] Hendrawan, V. Y., Winardi, S., dan Surbakti, "Sistem Informasi Rekam Medis Rawat Jalan Dan Pemeriksaan Penunjang Diagnosa Berbasis *Website* (Studi Kasus : Rumah Sakit Khusus Bedah Klinik Sinduadi, Melati, Sleman, Yogyakarta).,"Jurnal Teknologi Infor masi, vol. 9, no. 27, pp. 53–58, 2014.

[6] Suprianto, Andi dan Matsea, A. A. Fathia, "RANCANG BANGUN APLIKASI PENDAFTARAN PASIEN ONLINE DAN PEMERIKSAAN DOKTER DI KLINIK PENGOBATAN BERBASIS *WEB*," Jurnal Rekayasa Informasi, vol. 7, no. 1, pp. 48-58, April 2018.

[7] Dewi, Christine Dewi dan Sasongko, A. Widhyo, "Sistem Pelaporan Infrastruktur Dinas Bina Marga Dan PSDA Kota Salatiga Menggunakan NodeJs Berbasis *Web*," *Indonesian Journal of Computing and Modeling*, vol. 1, no. 1, Januari 2018.

[8] Bagas M., dan I Kadek D. N., "SISTEM OPTIMASI PEMBERIAN BONUS PEGAWAI DI *MIX'S TOUR TRAVEL* (Studi Kasus : *MIX'S TOUR TRAVEL*)," Jurnal Manajemen Informatika, vol. 5, no. 2, pp. 160-165, 2016.

[9] Maharani, Siti Zahra, "RANCANG BANGUN APLIKASI PENGANGGARAN BARANG BERBASIS *WEB* STUDI KASUS UNIT SARANA DAN PRASARANA SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI," Sekolah Tinggi Teknologi Terpadu

Nurul Fikri, Teknik Informatika, pp. 6 – 15, Agustus 2021.

[10] Bambang H., Machudor Y., dan Nagara, "SISTEM INFORMASI MANAJEMEN KEUANGAN PADA PT. HULU BALANG MANDIRI MENGGUNAKAN *FRAMEWORK* LARAVEL," Jurnal Komputasi, vol. 7, no. 1, pp. 17-26, 2019.

[11] Adelia, Dina, "APLIKASI PEMESANAN PAKET WEDDING PADA HOUSE OF ZANTIQ MENGGUNAKAN METODE RAD (*RAPID APPLICATION DEVELOPMENT*) BERBASIS *WEB*," Politeknik Negeri Sriwijaya, Palembang, September 2020.

[12] Kartika, L. G. Surya dan Rinartha, Komang, "Pengaruh *Pagination* dan Kompleksitas *Query* Data Terhadap Aspek Kehijauan dari Sistem Informasi Manajemen," Jurnal Sistem dan Informatika, vol. 13, no. 2, Mei 2019.