

***Question and Answer Konsultasi Dokter Berbasis Website***

Laporan ini dibuat untuk memenuhi persyaratan  
kelulusan mata kuliah Proyek II



Oleh :

**1214007 RATU SUKMAKOMALA**

**1214033 ISNAENTI NUR LATIFAH**

**PROGRAM STUDI VOKASI UNIVERSITAS LOGISTIK DAN  
BISNIS INTERNASIONAL**

**D-IV TEKNIK INFORMATIKA**

**BANDUNG**

**2023**

***Website-Based Doctor Consultation Question and Answer***

*This report is made to fulfill the requirements for passing the Project II course*



Oleh :

**1214007 RATU SUKMAKOMALA**

**1214033 ISNAENTI NUR LATIFAH**

**PROGRAM STUDI VOKASI UNIVERSITAS LOGISTIK DAN  
BISNIS INTERNASIONAL**

**D-IV TEKNIK INFORMATIKA**

**BANDUNG**

**2023**

## **LEMBAR PENGESAHAN**

***Question and Answer Konsultasi Dokter Berbasis Website***

## **LEMBAR PENGESAHAN**

***Question and Answer Konsultasi Dokter Berbasis Website***

## SURAT PERNYATAAN TIDAK MELAKUKAN PLAGIARISME

Yang bertanda tangan di bawah ini:

Nama : Ratu SukmaKomala

NPM : 1214007

Program Studi : D4 Teknik Informatika

Judul Proyek : *Question and Answer* Konsultasi Dokter Berbasis *Website*

Menyatakan bahwa:

1. Proyek Pemrograman Aplikasi (PROYEK II) saya ini adalah asli dan belum pernah diajukan untuk memenuhi kelulusan matakuliah Proyek 2 pada Program Studi Vokasi Universitas Logistik dan Bisnis Internasional maupun di Perguruan Tinggi lainnya.
2. Proyek Pemrograman Aplikasi (PROYEK II) ini adalah murni gagasan, rumusan, dan penelitian saya sendiri tanpa bantuan pihak lain, kecuali arahan pembimbing.
3. Dalam Proyek Pemrograman Aplikasi (PROYEK II) ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan-penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya ini, serta sanksi lainnya sesuai dengan norma yang berlaku diperguruan tinggi lain.

Bandung, Januari 2023

Yang Membuat Pernyataan,

Materai

Ratu SukmaKomala

NPM 1214007

## **SURAT PERNYATAAN TIDAK MELAKUKAN PLAGIARISME**

Yang bertanda tangan di bawah ini:

Nama : Isnaenti Nur Latifah

NPM : 1214033

Program Studi : D4 Teknik Informatika

Judul Proyek : *Question and Answer* Konsultasi Dokter Berbasis *Website*

Menyatakan bahwa:

1. Proyek Pemrograman Aplikasi (PROYEK II) saya ini adalah asli dan belum pernah diajukan untuk memenuhi kelulusan matakuliah Proyek 2 pada Program Studi Vokasi Universitas Logistik dan Bisnis Internasional maupun di Perguruan Tinggi lainnya.
2. Proyek Pemrograman Aplikasi (PROYEK II) ini adalah murni gagasan, rumusan, dan penelitian saya sendiri tanpa bantuan pihak lain, kecuali arahan pembimbing.
3. Dalam Proyek Pemrograman Aplikasi (PROYEK II) ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan-penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya ini, serta sanksi lainnya sesuai dengan norma yang berlaku diperguruan tinggi lain.

Bandung, Januari 2023

Yang Membuat Pernyataan,

Materai

Isnaenti Nur Latifah

NPM1214033

## ABSTRAK

Teknologi informasi merupakan hal yang tidak dapat dihindari di jaman yang sangat *modern* ini. Hal tersebut akan juga berpengaruh pada keinginan masyarakat untuk mendapatkan pelayanan kesehatan yang bermutu dengan adanya kemajuan teknologi. Rumah sakit sebagai salah satu sarana kesehatan yang memberikan pelayanan kesehatan kepada masyarakat memiliki peran yang sangat strategis dalam mempercepat peningkatan derajat kesehatan masyarakat.

Permasalahan yang terjadi masyarakat belum mengetahui dan belum menggunakan teknologi untuk dapat konsultasi dan membuat janji dengan dokter. Serta masyarakat belum mengetahui proses *Q&A* dan buat janji dengan dokter. Maka dari itu penulis membangun aplikasi *Question and Answer* Konsultasi Dokter berbasis *website* bertujuan agar masyarakat dapat melakukan tanya jawab mengenai konsultasi kesehatan dan membuat janji dengan dokter dan proses *Q&A* yang penulis bangun menggunakan fitur *livechat twak.to*.

Hasil analisa *Q&A* konsultasi dokter yang telah penulis bangun pada *website* yaitu sudah dapat digunakan oleh masyarakat (pasien) untuk bertanya kepada dokter melalui *handphone* tentang masalah kesehatannya dan membuat janji dengan dokter.

**Kata Kunci :** *Question and Answer, Konsultasi Dokter Berbasis Website*

## ***ABSTRACT***

*Information technology is something that cannot be avoided in this very modern era. This will also affect people's desire to get quality health services with advances in technology. The hospital as a health facility that provides health services to the community has a very strategic role in accelerating the improvement of public health status.*

*The problem that occurs is that people do not know about and have not used technology to be able to consult and make appointments with doctors. In addition, the community does not know about the Q&A process and make appointments with doctors. Therefore the authors built a website-based Doctor Consultation Question and Answer application with the aim that the public can ask questions and answers regarding health consultations and make appointments with doctors and the Q&A process which the author built using the twak.to livechat feature.*

*The results of the Q&A analysis of the doctor's consultation that the author has built on the website are already available for the public (patients) to ask doctors via mobile phones about their health problems and make appointments with doctors.*

**Keywords:** *Website-Based Doctor Consultation, Question and Answer*



## KATA PENGANTAR

Puji dan syukur ke hadirat Tuhan yang Maha Esa, karena atas berkat dan rahmat-Nya, penulis dapat mengerjakan dan menyelesaikan laporan analisis yang berjudul “*Question and Answer Konsultasi Dokter Berbasis Website*”. Penulis menggunakan bahasa yang sederhana dan mudah untuk dimengerti bagi para pembaca. Adapun Penulisan buku ini dibuat dalam rangka memenuhi salah satu syarat kelulusan Mata kuliah Proyek II di Universitas Logistik dan Bisnis Internasional.

Akhirnya dengan segala kerendahan hati, perkenankanlah penulis mengucapkan terima kasih yang tidak terhingga atas segala bantuan dan bimbingan yang telah diberikan dalam Penulisan laporan ini, kepada yang terhormat :

1. Kedua orang tua tercinta, yang telah memberikan doa dan dukungan baik secara moril maupun materil.
2. Roni Andarsyah,. ST., M.KOM.,SFPC. selaku Ketua Proram Studi D4 Teknik Informatika.
3. Nisa Hanum Harani, S.Kom.,M.T.,CDSP,SFPC selaku Koordinator Proyek II D4 Teknik Informatika.
4. Roni Andarsyah,. ST., M.KOM.,SFPC, selaku dosen pembimbing.

Penulis berusaha dengan sebaik-baiknya dalam Penulisan laporan ini, namun dalam banyak hal, penulis menyadari bahwa laporan ini masih jauh dari kata sempurna. Maka dari itu, saran dan kritik yang bersifat membangun dari semua pihak sangat penulis harapkan, agar laporan ini menjadi lebih baik dan bermanfaat.

Bandung, Januari 2023

Penulis

## DAFTAR ISI

ABSTRAK .....	i
KATA PENGANTAR .....	iii
DAFTAR ISI.....	iv
DAFTAR GAMBAR .....	vi
DAFTAR LAMPIRAN .....	vii
BAB I PENDAHULUAN .....	I-1
1.1 Latar Belakang .....	I-1
1.2 Identifikasi Masalah .....	I-2
1.3 Tujuan.....	I-2
1.4 Lingkup Dokumentasi .....	I-3
BAB II LANDASAN TEORI .....	II-4
2.1 Teknologi Informasi .....	II-4
2.2 Pelayanan Rumah Sakit.....	II-4
2.4 Internet.....	II-6
2.5 <i>Visual Studio Code</i> .....	II-6
2.6 <i>XAMPP</i> .....	II-7
2.7 <i>MySQL</i> .....	II-8
2.8 <i>Question and Answer</i> .....	II-9
2.9 <i>FlowMap</i> .....	II-9
BAB III ANALISIS DAN PERANCANGAN .....	III-10
3. 1 Analisa Pelayanan Rumah Sakit.....	III-10
3.1.2 Alur Kerja Aplikasi Melalui <i>Flowmap</i> Yang Sedang Berjalan.....	III-11
3.2 Analisa <i>Website</i> Pelayanan Rumah Sakit Yang Akan Dibangun.....	III-12
3.2.1 <i>Flowmap Website</i> Yang Akan Dibangun .....	III-13
3.2.2 Alur Kerja <i>Website</i> Yang Akan Dibangun .....	III-13
3.3 <i>Use Case Diagram Question &amp; Answer</i> Konsultasi Dokter .....	III-14
3.3.1. Penjelasan Alur <i>Use case Diagram Q&amp;A</i> Konsultasi Dokter .....	III-14
3.4 <i>Sequence Diagram Question &amp; Answer</i> Konsultasi Dokter .....	III-15
3.4.1 Penjelasan Alur <i>Sequence Diagram Question &amp; Answer</i> Konsultasi Dokter .....	III-15

3.5 Struktur <i>Database</i> .....	III-16
3.5.1. <i>Source code SQL</i> Dokter pada <i>Database</i> .....	III-16
3.5.2 Struktur.....	III-20
3.5.3 <i>Browse</i> .....	III-21
3.6 Logika Fungsi atau Algoritma Fungsi.....	III-22
3.7 <i>Code Coverage</i> .....	III-23
3.7.2 Tampilan pada <i>CRUD</i> .....	III-27
BAB IV IMPLEMENTASI DAN PENGUJIAN.....	IV-30
4.1 Pembahasan Hasil Implementasi.....	IV-30
4.1.1 Analisis Daftar (Register).....	IV-30
4.1.2 Analisis <i>Login</i> .....	IV-33
4.1.3 Analisis Jadwal Dokter.....	IV-36
4.1.4 Analisis <i>Chat</i> Dengan Dokter.....	IV-38
4.1.5 Analisis Buat Janji Dengan Dokter .....	IV-39
4.2 Pengujian dan Hasil Pengujian.....	IV-42
4.2.1 Hasil Tampilan Pada Daftar (Register) .....	IV-42
4.2.2 Hasil Tampilan Pada <i>Login</i> .....	IV-42
4.2.3 Hasil Tampilan Pada Halaman Utama Admin .....	IV-43
4.2.4 Hasil Tampilan Pada Jadwal Dokter .....	IV-44
4.2.5 Hasil Tampilan Pada <i>Chat</i> Dokter.....	IV-44
4.2.6 Hasil Tampilan Pada Buat Janji Dengan Dokter.....	IV-45
BAB V KESIMPULAN DAN SARAN.....	V-46
5.1 Kesimpulan.....	V-46
5.2 Saran.....	V-46
DAFTAR PUSTAKA .....	viii
LAMPIRAN-LAMPIRAN.....	xv

## DAFTAR GAMBAR

Gambar 2. 1 Live Chat Tawk To .....	6
Gambar 2. 2 Visual Studio Code.....	7
Gambar 2. 3 XAMPP .....	8
Gambar 3. 1 Aplikasi Halo Doc .....	10
Gambar 3. 2 Flowmap Aplikasi Yang sedang berjalan.....	11
Gambar 3. 3 Website Q&A Konsultasi dengan Dokter .....	12
Gambar 3. 4 Flowmap website Q&A konsultasi dengan dokter yang akan dibangun .....	13
Gambar 3. 5 Use case Diagram.....	14
Gambar 3. 6 Gambar 3.6 Sequence Diagram.....	15
Gambar 3. 7 Database dokter .....	20
Gambar 3. 8 Stuktur database .....	20
Gambar 3. 9 Struktur tabel pada database dokter .....	20
Gambar 3. 10 Data value pada database dokter .....	21
Gambar 3. 11 Data value pada database dokter .....	21
Gambar 3. 12 Browse Database .....	21
Gambar 3. 13 Flowchart.....	22
Gambar 3.14 Flowgraph.....	23
Gambar 3.15 Pengujian daftar (register) .....	24
Gambar 3.16 Pengujian login.....	24
Gambar 3.17 Pengujian halaman utama_user2 .....	25
Gambar 3.18 Pengujian halaman utama admin.....	25
Gambar 3.19 Pengujian Chat .....	26
Gambar 3. 20 Pengujian buat_janji.....	27
Gambar 3.21 pengujian tambah_data.....	27
Gambar 3.22 pengujian edit_data .....	28
Gambar 3.23 pengujian hapus_data .....	28
Gambar 3.24 Pengujian Code Coverage .....	29
Gambar 4. 1 Daftar (register) .....	42
Gambar 4.2 Login user.....	42
Gambar 4.3 Login admin .....	43

Gambar 4.4 Halaman utama_user (pasien) .....	43
Gambar 4.5 Halaman utama admin.....	44
Gambar 4.6 Jadwal dokter.....	44
Gambar 4.7 Chat dokter .....	45
Gambar 4.8 Buat janji dengan dokter .....	45
Gambar 4. 9 Form data diri pasien.....	45

## **DAFTAR LAMPIRAN**

Lampiran 1 Formulir Kegiatan Ratu .....	49
Lampiran 2 Formulir Kegiatan Isnaenti .....	50
Lampiran 3 Bukti Code Coverage.....	51

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Teknologi informasi merupakan hal yang tidak dapat dihindari di jaman yang sangat *modern* ini. Peranan teknologi pada aktifitas manusia saat ini memang begitu besar. Berkat teknologi berbagai kemudahan dapat dirasakan oleh manusia. Hal tersebut akan juga berpengaruh pada keinginan masyarakat untuk mendapatkan pelayanan kesehatan yang bermutu dengan adanya kemajuan teknologi. Dalam era revolusi digital, suara, data, dan gambar yang dapat dikirim dalam berbagai bentuk menuntut kita untuk selalu meningkatkan perkembangan teknologi dengan percepatan informasi [1].

Rumah sakit sebagai salah satu sarana kesehatan yang memberikan pelayanan kesehatan kepada masyarakat memiliki peran yang sangat strategis dalam mempercepat peningkatan derajat kesehatan masyarakat. Oleh karena itu rumah sakit dituntut untuk memberikan pelayanan yang bermutu sesuai dengan standar yang ditetapkan dan dapat menjangkau seluruh lapisan masyarakat, salah satunya yaitu kualitas pelayanan yang menjadi pertimbangan masyarakat untuk memilih suatu fasilitas kesehatan yang akan digunakannya [2].

Aplikasi *Question and Answer* Konsultasi Dokter berbasis *website* merupakan suatu aplikasi *web service* yang berfungsi melakukan tanya jawab mengenai konsultasi kesehatan dan membuat janji dengan dokter. Operator atau praktisi kesehatan yang berguna untuk memberikan pelayanan, dimana rumah sakit memberikan layanan bagi pasien secara *online* menggunakan aplikasi berbasis *website* untuk rumah sakit [3].

*Web Service* adalah suatu sistem perangkat lunak yang dirancang untuk mendukung *interaction and interoperability* antar sistem pada suatu jaringan. Teknologi *web service* digunakan sebagai suatu fasilitas yang menyediakan pelayanan (dalam bentuk tanya jawab untuk konsultasi dan membuat janji dengan dokter), sehingga dapat berinteraksi dengan sistem tersebut melalui layanan-layanan yang disediakan. Maka implementasi *web services* menjanjikan kemudahan dalam pelayanan melalui internet [4].

Salah satunya yaitu kegiatan konsultasi merupakan salah satu langkah pendukung untuk menegakkan diagnosis. Seiring dengan perkembangan teknologi yang sangat

pesat, maka kegiatan konsultasi dapat dilakukan secara online. Kegiatan konsultasi online membutuhkan kemampuan komunikasi interpersonal yang efektif baik dari pihak dokter estetika maupun dari pihak pasien sehingga kegiatan konsultasi online dapat berjalan dengan baik [5].

Konsultasi dokter berbasis *website* melalui media *online* ini adalah hal baru, banyak diminati orang karena memberikan dampak positif. Dampak positifnya ialah orang dapat dengan cepat bertanya kepada dokter melalui *handphone* tentang masalah kesehatannya [6]. Dalam pelayanan kesehatan berbasis digital ini jika seseorang ingin melakukan konsultasi secara online atau membuat janji temu terdapat banyak pilihan dokter yang tersedia dengan berbagai kriteria [7].

Sebagai peneliti penulis membuat aplikasi serupa *Halodoc* yang berjudul *Question and Answer Konsultasi Dokter Berbasis Website*. Penulis memilih aplikasi *Halodoc* dan tertarik untuk menganalisis membuat bagian *Q&A* pada aplikasi dikarenakan *Q&A* berguna untuk mempermudah layanan konsultasi dan membuat janji dengan dokter melalui website yang penulis buat.

## 1.2 Identifikasi Masalah

Dalam pembuatan proposal transaksi pada aplikasi *Question and Answer Konsultasi Dokter Berbasis Website*, lebih memfokuskan materi terhadap proses *question and answer*. Adapun masalah yang akan diselesaikan oleh aplikasi berbasis *website* yang penulis bangun adalah sebagai berikut:

1. Bagaimana cara pelayanan untuk mengetahui konsultasi dan membuat janji dengan dokter pada *website*.
2. Bagaimana proses *Q&A* dan membuat janji dengan dokter yang dibuat dalam *website*.

## 1.3 Tujuan

Tujuan penulis membangun *website* ini adalah untuk mempermudah proses *Question and Answer Konsultasi Dokter Berbasis Website*, dalam menyelesaikan masalah yang sudah di bahas di bagian sebelumnya. Berikut tujuan dari dibuatnya *website* ini:

1. Pelayanan yang diberikan oleh rumah sakit kepada masyarakat yaitu dapat



bertanya dengan dokter mengenai konsultasi dan membuat janji dengan dokter melalui *website*.

2. *Q&A* dibuat dalam bentuk fitur *live chat* dimana tidak diperlukan nomer telpon rumah sakit sehingga masyarakat bisa klik fitur *live chat* tersebut dan bertanya.

#### **1.4 Lingkup Dokumentasi**

Pada laporan proyek TI II ini permasalahan yang dibahas dibatasi dalam hal sebagai berikut :

1. Dalam pembahasan proyek II ini, hanya membahas tentang bagaimana proses *Q&A* Konsultasi dan membuat janji dengan dokter pada *website*.
2. Membahas tentang fitur *live chat* pada *website*.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Teknologi Informasi**

Era globalisasi ini perkembangan teknologi sangat cepat terutama di bidang sistem informasi salah satunya rumah sakit yang pastinya membutuhkan peran teknologi informasi dalam keamanan atau pengelolaan data. Oleh sebab itu, sangatlah penting bagi sebuah pelayanan rumah sakit untuk menerapkan suatu sistem informasi yang terpercaya dalam mengatur kelancaran proses data serta membantu dan meningkatkan kecepatan pekerjaan yang dilakukan. Layanan teknologi informasi yang tepat waktu, aman, akurat dan relevan dengan kebutuhan pasien merupakan hal yang sangat penting diperhatikan dalam mendukung kelancaran pelaksanaan pelayanan rumah sakit [8].

Teknologi informasi yang berkembang saat ini dimanfaatkan hampir dalam semua aspek kehidupan manusia. Salah satu aspek yang tidak terlepas dari pemanfaatan teknologi informasi adalah pelayanan pada rumah sakit. Penerapan media pelayanan rumah sakit berbasis teknologi informasi adalah penggunaan perangkat keras (*hardware*) teknologi informasi seperti komputer, laptop, *infocus* yang didukung dengan pemanfaatan perangkat lunak (*software*) seperti *software* untuk melakukan presentasi, menampilkan gambar bergerak (*video*) dan suara (*audio*) sebagai alat bantu bagi admin pelayanan rumah sakit dalam menyampaikan atau menanyakan mengenai pelayanan rumah sakit [9].

#### **2.2 Pelayanan Rumah Sakit**

Pelayanan kesehatan di Indonesia masih tertinggal jauh dibandingkan negara-negara *ASEAN* lainnya seperti Malaysia. Dalam menilai kualitas pelayanan rumah sakit maka *National Health Services (NHS)* memperkenalkan 6 (enam) syarat dalam menilai kinerja pelayanan rumah sakit, salah satunya yaitu efisiensi [10].

Rumah sakit harus peduli dengan kualitas pelayanan yang diberikan kepada pasien agar rumah sakit mengetahui seberapa puas pasien dengan kualitas pelayanan di rumah sakit. Pelayanan kesehatan menimbulkan persaingan antara penyedia pelayanan kesehatan termasuk diantaranya adalah rumah sakit. Dengan adanya

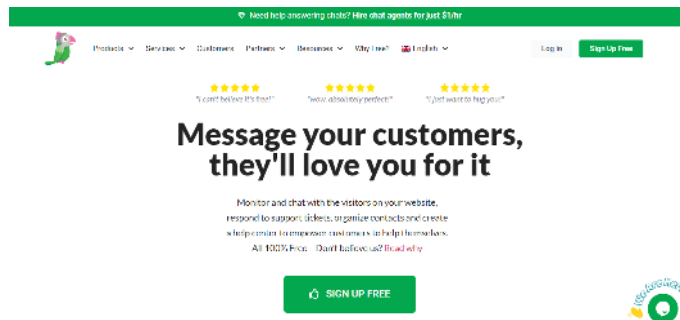
persaingan antar rumah sakit yang semakin tinggi disertai dengan banyaknya pembangunan klinik baru maka klinik perlu terus mengembangkan diri dengan menyelenggarakan pelayanan yang bermutu dan meningkatkan kualitas baik pelayanan *medice* maupun pelayanan administrasinya guna memberikan kepuasan terhadap pasien [11].

Jumlah jam kerja dokter yang dipandang berlebihan sudah menjadi sorotan nasional dan internasional dalam beberapa waktu terakhir. Banyaknya jumlah jam kerja berbanding lurus dengan beban kerja dokter yang besar. Beberapa hal yang diduga menjadi penyebab peningkatan beban kerja dokter, antara lain jumlah pasien yang meningkat, persebaran dokter yang tidak merata, kompleksitas penyakit pasien yang semakin tinggi, serta jumlah proyek penelitian dokter yang semakin banyak. Dengan ini maka di perlukan sebuah *website* yang bertujuan untuk *Question and Answer* yang berfungsi untuk menanyakan konsultasi dan membuat janji dengan dokter kepada pihak pelayanan rumah sakit melalui *web* tersebut [12].

### 2.3 Live Chat

Pelayanan rumah sakit yang sudah terotomasi dapat mengoptimalkan pelayanannya dengan mengimplementasikan teknologi berupa aplikasi komunikasi secara virtual. Hal tersebut dilakukan demi mendukung program rumah sakit . *Tawk.to* merupakan salah satu aplikasi *live chat online* gratis yang dapat dipasang pada situs *web* perpustakaan. Manfaat dari penggunaan *Tawk.to* pada rumah sakit adalah sebagai alat komunikasi antara pasien dan admin rumah sakit dalam menangani segala kendala mengenai pertanyaan konsultasi dokter dengan memanfaatkan layanan digital [13].

*Tawk To* yang merupakan sebuah fitur *chat online* di *iDuHelp!* Berguna untuk sebuah pelayanan informasi question and answer dalam pelayanan rumah sakit . Fitur ini merupakan *plug-in* dari *iDuHelp!* yang bisa menghubungkan pasien rumah sakit langsung kepada admin pelayanan rumah sakit yang akan melayani permasalahan yang di tanyakan oleh pasien [14]. Berikut gambar live chat tawk to, dapat dilihat pada gambar 2.1, sebagai berikut:



Gambar 2. 1 Live Chat Tawk To

## 2.4 Internet

Internet merupakan “Jaringan yang bersifat global dan menghubungkan komputer-komputer ke seluruh dunia dengan menggunakan internet sebuah komputer tahun 1982 istilah Internet pertama kali digunakan, dan *TCP/IP* diadopsi sebagai bisa mengakses data yang terdapat pada komputer lain dibenua yang berbeda” [15].

Pada *protocol universal* untuk jaringan tersebut. Pada tahun 1986 diperkenalkan nama sistem domain, yang sekarang dikenal *DNS (Domain Name System)* [16].

Internet menawarkan alternatif baru dalam pemerolehan informasi dan sekaligus menyebarluaskan informasi. Jika sebelumnya, informasi berbasis cetak merupakan primadona perpustakaan tradisional, sekarang tersedia format baru dalam bentuk digital melalui *web*. Koleksi bahan digital yang ditransmisikan secara elektronik dan disebut perpustakaan digital, keberadaannya semakin penting dalam pemenuhan kebutuhan informasi pengguna [17].

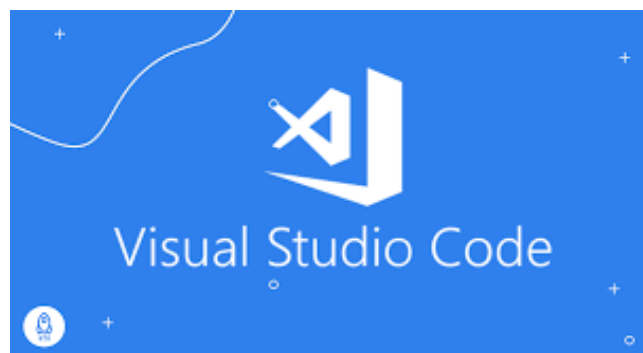
## 2.5 Visual Studio Code

*Visual Studio Code (VS Code)* ini merupakan sebuah teks editor ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows*. Teks editor ini secara langsung mendukung bahasa pemrograman *JavaScript*, *Typescript*, dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang *via marketplace Visual Studio Code* (seperti *C++*, *C#*, *Python*, *Go*, *Java*, dst) [18].

Banyak fitur yang disediakan oleh *Visual Studio Code*, diantaranya *Intellisense*, *Git Integration*, *Debugging*, dan fitur ekstensi yang menambah kemampuan teks editor. Fitur tersebut akan bertambah seiring dengan bertambahnya versi *VS Code*.

Pembaruan versi *Visual Studio Code* ini juga dilakukan berkala setiap bulan, dan inilah yang membedakan *VS Code* dengan teks editor-teks editor yang lain [19].

*Visual Studio Code* merupakan sebuah aplikasi editor *code open source* yang dikembangkan oleh *Microsoft* untuk sistem operasi *Windows*, *Linux*, dan *MacOS*. *Visual Code* memudahkan dalam penulisan *code* yang mendukung beberapa jenis pemrograman, seperti *C++*, *C#*, *Java*, *Python*, *PHP*, *GO*. *Visual Code* memiliki kemampuan untuk mengidentifikasi jenis bahasa pemrograman yang digunakan dan memberi variasi warna sesuai dengan fungsi dalam rangkaian *code* tersebut. *Visual Studio Code* juga telah terintegrasi ke *Github*. Selain itu fitur lainnya adalah kemampuan untuk menambah ekstensi dimana para pengembang dapat menambah ekstensi untuk menambah fitur yang tidak ada di *Visual Studio Code* [20].



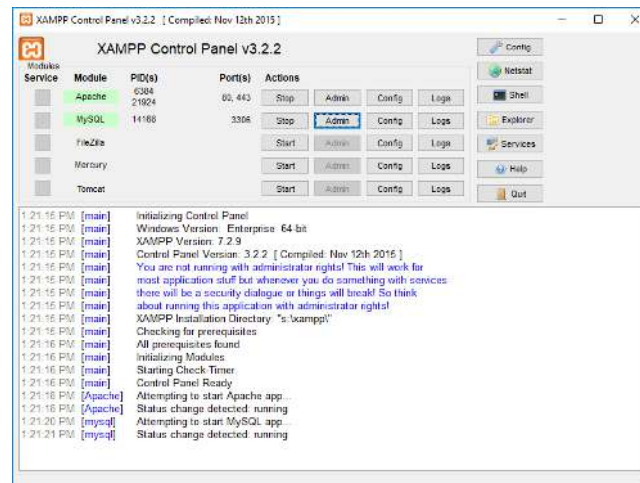
Gambar 2. 2 *Visual Studio Code*

## 2.6 XAMPP

*XAMPP* ialah *software* yang di dalamnya terdapat *server MySQL* dan didukung oleh *PHP* sebagai bahasa pemrograman untuk membuat *website* dinamis serta terdapat *web server apache* yang bisa dijalankan pada beberapa *platform* antara lain *OS X*, *Windows*, *Linux*, *Mac*, dan *Solaris*. *XAMPP* adalah *software server apache* dimana dalam *XAMPP* yang telah tersedia *database server* seperti *MySQL* dan juga *PHP programming*. *XAMPP* memiliki keunggulan yaitu cukup mudah dioperasikan, tidak memerlukan biaya serta mendukung instalasi pada *Windows* dan *linux* [21].

*Xampp* adalah sebuah paket kumpulan *software* yang terdiri dari *Apache*, *MySQL*, *PhpMyAdmin*, *PHP*, *Perl*, *Filezilla*. *Xampp* berfungsi untuk memudahkan instalasi lingkungan *PHP*, di mana biasanya lingkungan pengembangan *web* memerlukan *PHP*, *Apache*, *MySQL* dan *PhpMyAdmin* [22].

*XAMPP* merupakan singkatan dari *X* (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP*, dan *Perl*. *XAMPP* adalah *tool* yang menyediakan paket perangkat lunak dalam satu buah paket. Dalam paket *XAMPP* sudah terdapat *Apache* (*web server*), *MySQL* (*database*), *PHP* (*server side scripting*), *Perl*, *FTP server*, *PhpMyAdmin* dan berbagai pustaka bantu lainnya [23].



Gambar 2. 3 *XAMPP*

## 2.7 *MySQL*

*MySQL* adalah sebuah *database* yang terkenal karena hampir sebagian besar aplikasinya yaitu berbasis *website* seperti *wordpress*, yang dilengkapi dengan *MySQL*. *MySQL* juga ditawarkan dalam berbagai versi termasuk versi gratis. *MySQL* adalah sistem manajemen *database SQL* yang sifatnya *open source* (terbuka) dan paling banyak digunakan saat ini [24].

*MySQL* adalah penghubung (*RDBMS*) yang bersifat *open source*. Perangkat lunak *database* pada umumnya dibandingkan dengan bahasa pemrograman *server web* seperti *PHP* maupun *JSP*. *MySQL* (*My Structured Query Language*) yaitu sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan *DBMS* (*Database Management System*), sifat *DBMS* ini ialah *open source*. *MySQL* juga merupakan program pengakses *database* yang sifatnya jaringan, sehingga dapat digunakan untuk aplikasi *Multi User* [25].

*MySQL* adalah salah satu jenis *database server* yang sangat terkenal. Kepopulerannya disebabkan *Mysql* menggunakan *SQL* sebagai bahasa dasar untuk mengakses *databasenya*. *Mysql* termasuk jenis *RDBMS* (*Relational Database*

*Management System*). Pada *Mysql*, sebuah database mengandung satu atau sejumlah tabel [26].

## 2.8 *Question and Answer*

*Question and Answer* merupakan suatu aktivitas percakapan antara dua orang yang sedang membaca suatu bacaan, yang satu bertindak sebagai penanya dan yang lain sebagai penjawab. Terdapat dua peran yaitu peran penanya dan penjawab, keduanya melihat percakapan yang terjadi. Penjawab tetap berpegang pada kosa kata dalam bagian itu untuk membatasi jumlah kemungkinan jawaban [27].

*Question answering system (QA system)* adalah sistem yang mengijinkan user menyatakan kebutuhan informasinya dalam bentuk *natural language question* (pertanyaan dalam bahasa alami), dan mengembalikan kutipan teks singkat atau bahkan frase sebagai jawaban. Maka *QA system* akan semakin dibutuhkan karena memberikan banyak keuntungan dengan adanya sumber pengetahuan yang luas, dan dapat mengatasi sejumlah data yang tidak berguna [28].

## 2.9 *FlowMap*

*Flowmap* adalah campuran peta dan *flowchart*, yang menunjukkan pergerakan benda dari satu lokasi ke lokasi lain, seperti jumlah barang yang diperdagangkan, atau jumlah paket dalam jaringan. *Flow map* menolong analisis dan *programmer* untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoprasian [29].

*Flowmap* merupakan diagram yang menunjukkan aliran data berupa formulir ataupun keterangan berupa dokumentasi yang mengalir dalam suatu sistem. Adapun fungsi dari *flowmap* adalah untuk mempermudah penggambaran aliran data yang berupa dokumen sistem yang sedang berjalan [30]. *flowmap* adalah menjelaskan mengenai beberapa hak akses yang dimiliki oleh admin. Dalam diagram *flowmap* terdapat kegiatan atau urutan interaksi yang saling berkaitan antara *system* dan *actor* [31].

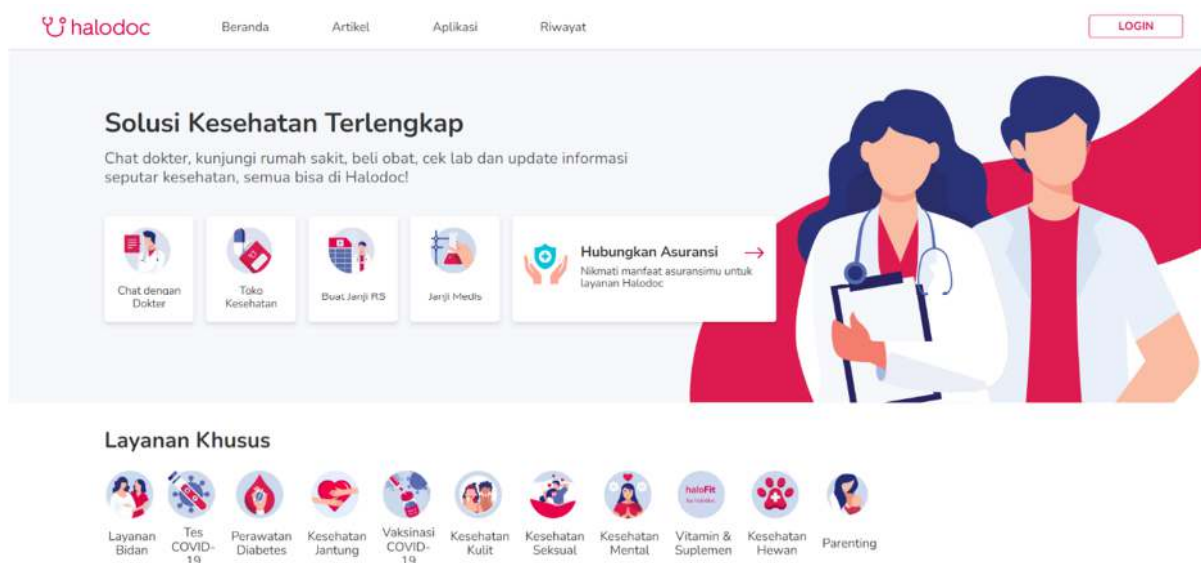
## BAB III

### ANALISIS DAN PERANCANGAN

#### 3.1 Analisa Pelayanan Rumah Sakit

Rumah Sakit merupakan institusi pelayanan kesehatan yang menyediakan pelayanan rawat inap, rawat jalan, dan gawat darurat. Didalam pelayanan terdapat indikator mutu pelayanan keperawatan diantaranya kenyamanan, kepuasan pasien dan keselamatan pasien. Dengan adanya aplikasi yang dibuatkan rumah sakit tersebut sebagai pemenuhan dari mutu pelayanan rumah sakit. Berikut strategi dalam meningkatkan pelayanan rumah sakit diantaranya:

1. Pelayanan cepat dan tanggap
2. Memiliki pelayanan *customer service*
3. Mendengar keluhan pasien
4. Menjaga kesopanan

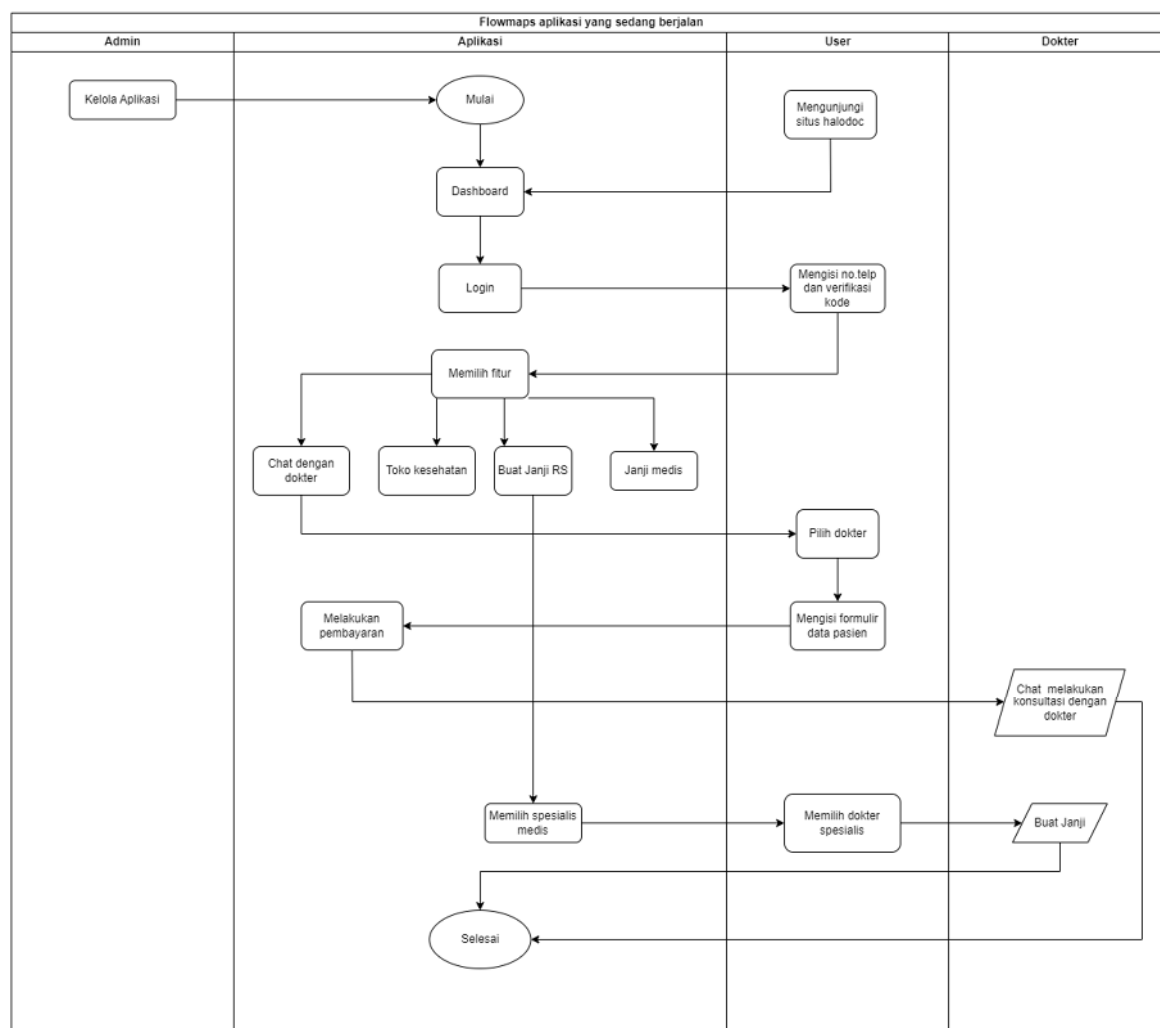


Gambar 3.1 Aplikasi *Halodoc*

Menunjukkan tampilan asli dari aplikasi pelayanan rumah sakit yang didalamnya pasien dapat memilih navigasi yang tersedia Solusi Kesehatan Terlengkap diantaranya *Chat* dengan dokter, Toko kesehatan, Buat janji RS dan Janji Medis. *Website* pelayanan rumah sakit ini digunakan untuk segala keperluan pasien.



### 3.1.1 Flowmap Aplikasi Pelayanan Rumah Sakit Yang Sedang Berjalan



Gambar 3. 2 Flowmap Aplikasi Yang sedang berjalan

### 3.1.2 Alur Kerja Aplikasi Melalui Flowmap Yang Sedang Berjalan

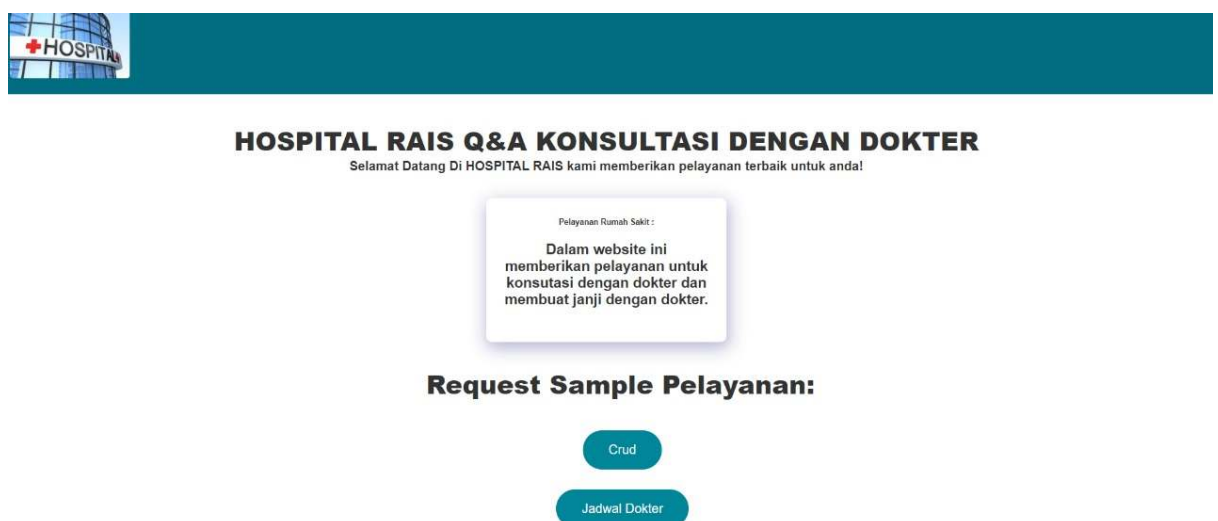
Dalam aplikasi *Halodoc* yang sedang berjalan dibuat sebuah perancangan yaitu Pelayanan rumah sakit. Pada pelayanan rumah sakit di aplikasi *Halodoc* ini berfungsi untuk mempermudah masyarakat (Pasien) dalam melihat kebutuhan yang diperlukan seperti pendaftaran *online*, pencarian RS klinik dan cari dokter. Sehingga dengan adanya aplikasi ini mempermudah pasien untuk mendaftarkan data dirinya tanpa harus ke rumah sakit terlebih dahulu.

Pada gambar *flowmap* diatas menjelaskan gambaran dari sebuah sistem yang sedang berjalan yaitu *halodoc*. Dimana terdapat tiga *actor* yaitu admin, *user* (pasien) dan dokter. Ketiga *actor* tersebut memiliki tugasnya masing – masing.

Admin bertugas dan bertanggung jawab pada bagian seluruh sistem ketika terjadi kendala, *user* hanya dapat melakukan *login* dengan mengisi no tlp dan melakukan kegiatan sesuai dengan keperluan pada sistem ini, dan dokter dapat masuk dengan melakukan *login* dan bertugas sebagai penjawab dari pertanyaan seputar konsultasi pasien dan membuat janji dengan pasien. Pada sistem ini dapat menampilkan *dashboard* sebagai halaman awal dan berlanjut ke *login* untuk melakukan proses atau langkah selanjutnya. Setelah itu sistem akan menampilkan beberapa fitur yang diantaranya: *chat* dengan dokter, toko kesehatan, buat janji rs., dan janji medis. Dari beberapa fitur yang tersedia penulis membuat *website* serupa pada bagian *chat* dokter dan buat janji rs yang dimana langkah selanjutnya *user* (pasien) dapat memilih dokter yang dibutuhkan dan melengkapi prosedur hingga selesai.

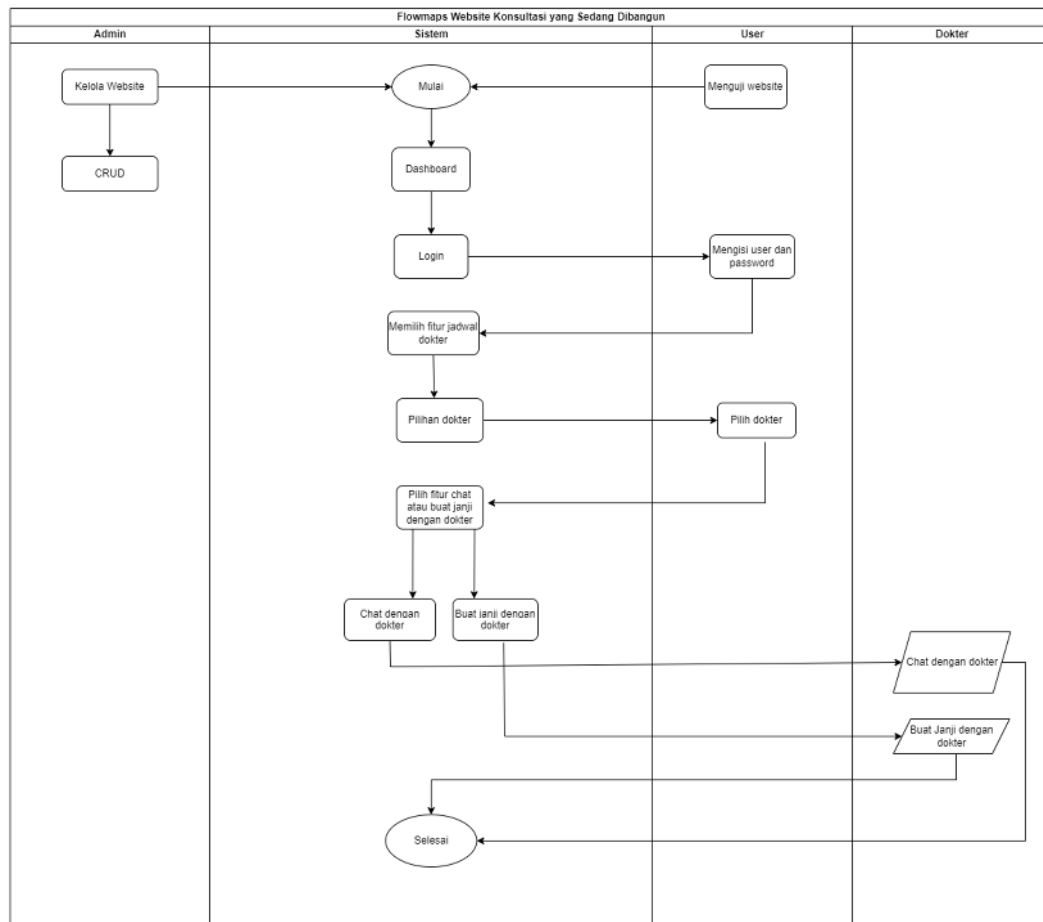
### 3.2 Analisa Website Pelayanan Rumah Sakit Yang Akan Dibangun

*Website Q&A* konsultasi dengan dokter merupakan *website* yang dibuat serupa dengan aplikasi *halodoc* yang dimana dapat berfungsi sebagai media konsultasi *online* antara *user* (pasien) dengan dokter. Konsultasi berbasis *website* ini dilakukan dengan cara mengunjungi *website* tersebut Kemudian, jika belum memiliki akun maka pilih fitur daftar, sedangkan jika sudah memiliki akun maka pasien langsung menuju fitur *login*. Setelah itu memilih fitur jadwal dokter dan memilih sesuai keperluan berkonsultasi melalui *chat* dengan dokter atau membuat janji terlebih dahulu dengan dokter untuk konsultasi secara langsung di rumah sakit



Gambar 3. 3 Website Q&A Konsultasi dengan Dokter

### 3.2.1 Flowmap Website Yang Akan Dibangun



Gambar 3. 4 Flowmap website Q&A konsultasi dengan dokter yang akan dibangun

### 3.2.2 Alur Kerja Website Yang Akan Dibangun

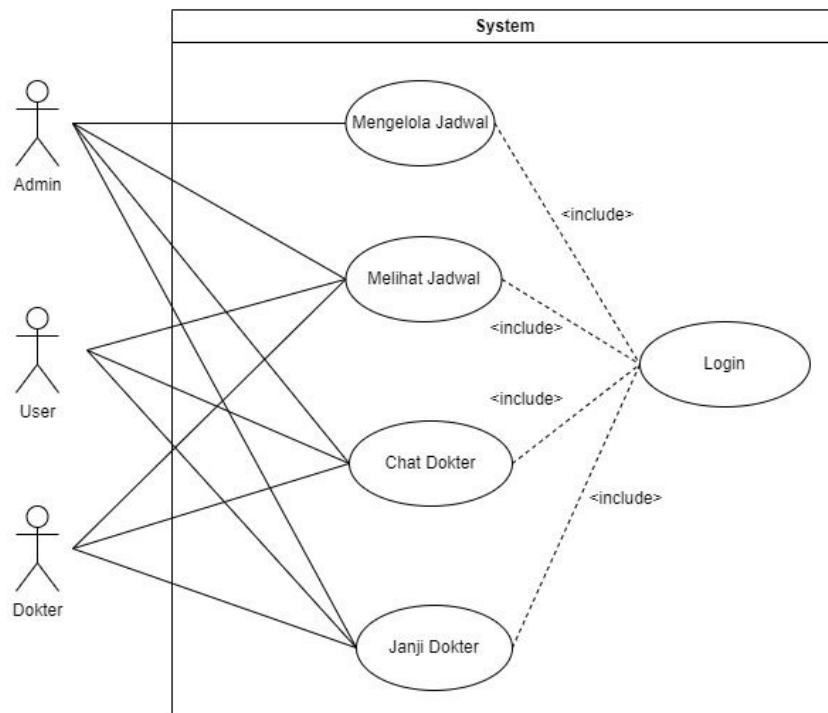
Dalam *website* rumah sakit RAIS yang akan dibangun dibuat sebuah perancangan yaitu pelayanan dalam konsultasi dokter. *Website* ini dibangun bertujuan untuk Q&A konsultasi dokter. Dengan adanya *website* ini diharapkan dapat mempermudah pasien untuk bertanya dalam halnya konsultasi keluhan dengan cara *online* sekaligus dapat membuat janji dengan dokter setelah melakukan kesepakatan.

Pada gambar *flowmap* diatas telah digambarkan alur dari *website* yang dibuat. Dimana dalam *website* tersebut terdapat tiga *actor* yang dapat menggunakan *website* tersebut, diantaranya: Admin, *user* (pasien) dan dokter. Peran dari ketiga *actor* tersebut tentunya berbeda-beda. Admin yang bertugas sebagai pemegang *crud*, *database* dan sekaligus sebagai penanganan ketika terjadi *error* pada *website* tersebut. *User* hanya dapat melakukan *login* dan daftar dengan mengisi data yang harus di isi pada

*website* tersebut. Dokter yang bertugas sebagai penjawab dari pertanyaan seputar konsultasi dan melakukan pertemuan janji dengan pasien yang telah disepakati bersama.

Pada sistem ini dijelaskan terdapat *dashboard* sebagai halaman awal *website* yang tersedia fitur *login* dan *daftar*. Kemudian pada admin di bedakan terdapat fitur *crud* yang dikelola oleh admin. Fitur jadwal dokterlah yang menjadi tujuan untuk konsultasi dan buat janji dengan dokter, caranya pasien setelah memasuki jadwal dokter kemudian pasien akan memilih dokter mana yang pasien butuhkan, selanjutnya pasien akan di pilihkan fitur *chat* dengan dokter atau membuat janji dengan dokter setelah ada kesepakatan bersama untuk pertemuan melakukan pemeriksaan lebih lanjut.

### 3.3 Use Case Diagram Question & Answer Konsultasi Dokter



Gambar 3. 5 Use case Diagram

#### 3.3.1. Penjelasan Alur Use case Diagram Q&A Konsultasi Dokter

- **Admin**

Admin bertugas memegang aplikasi dan bisa kelola login serta berpegang dalam perubahan jadwal dokter di dalam *CRUD*.

- **System**

*System (Website)* yang dibuat untuk mempermudah pasien dapat berkonsultasi dan membuat janji dengan dokter secara *online*, di dalam *system* ini terdapat *login*, *crud*. Jadwal dokter yang berisikan *chat* dengan dokter dan buat janji dengan dokter.

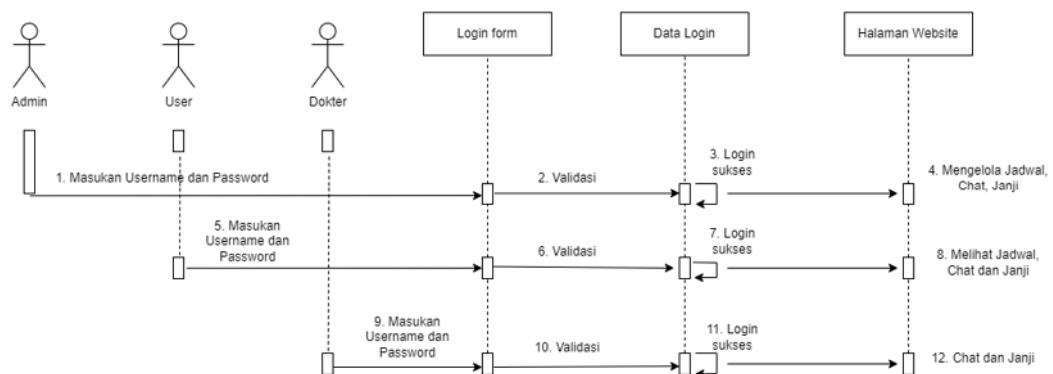
- **User**

*User* berperan sebagai pasien yang dimana dalam *website* dapat melakukan *login* terlebih dahulu kemudian mengunjungi fitur jadwal dokter yang dimana pasien memilih dokter setelah itu pilih fitur selanjutnya sesuai dengan keperluan pasien bisa *chat* dengan dokter dan buat janji dengan dokter.

- **Dokter**

Dokter dapat melakukan *login* terlebih dahulu untuk memasuki *website* dan peran dokter dapat melayani pasien dengan menjawab pertanyaan yang disampaikan oleh pasien yang berada di *chat* dokter (*live chat*) dan membuat kesepakatan dokter dengan pasien di fitur buat janji dengan dokter, serta dokter harus memiliki akun *twak.to*.

### 3.4 Sequence Diagram Question & Answer Konsultasi Dokter



Gambar 3. 6 Gambar 3.6 Sequence Diagram

#### 3.4.1 Penjelasan Alur Sequence Diagram Question & Answer Konsultasi Dokter

- **Admin**

Admin dapat kelola fitur yang ada di *website*. Admin melakukan pengisian *username* dan *password* untuk proses *login*, setelah validasi dan *login* sukses admin di arahkan pada halaman utaman yang ber-isikan *Crud* dan Jadwal dokter yang dimana terdapat *chat* dan buat janji dengan dokter. Tugas admin sendiri

memantau dan mengatasi jika *web* tersebut mengalami *error* dan pembaruan data dokter.

- **User**

*User* (pasien) harus mengisi *username* dan *password* untuk mengunjungi *website* tersebut. Setelah itu *website* akan memvalidasi dan *login* sukses, *User* (pasien) di arahkan pada halaman yang berisikan jadwal dokter yang nantinya *user* (pasien) dapat melihat dan memilih dokter mana yang ingin di tanyakan mengenai konsultasi. Pada jadwal dokter terdapat *chat* dan buat janji dengan dokter, *user* akan memilih sesuai dengan keperluan untuk konsultasi.

- **Dokter**

Dokter melakukan pengisian *username* dan *password* untuk mengunjungi *website*. Peran dokter dalam *website* tersebut yaitu membalas *chat* (pesan) mengenai konsultasi dari pasien dan membuat kesepakatan dengan pasien untuk konsultasi langsung di rumah sakit.

### 3.5 Struktur Database

#### 3.5.1. Source code SQL Dokter pada Database

```
-- phpMyAdmin SQL Dump
-- version 5.2.0
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Jan 23, 2023 at 06:45 AM
-- Server version: 10.4.25-MariaDB
-- PHP Version: 7.4.30

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `dokter`
--
```



```
--
-- Table structure for table `produk`
--

CREATE TABLE `produk` (
  `id` int(11) NOT NULL,
  `nama_dokter` varchar(255) DEFAULT NULL,
  `deskripsi` text DEFAULT NULL,
  `jam_operasional` varchar(28) DEFAULT NULL,
  `foto_dokter` varchar(255) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data for table `produk`
--

INSERT INTO `produk` (`id`, `nama_dokter`, `deskripsi`,
`jam_operasional`, `foto_dokter`) VALUES
(17, 'Dr. Nita ,Sp.BA', 'Bedah Anak', '10.00-12.00', '528-4.jpg'),
(18, 'Dr. Ayana,Sp.OG', 'Spesialis Kandungan', '13.30-15.00', '60-5.jpg'),
(19, 'Dr. Dodi,Sp.P', 'Spesialis Paru', '19.00-21.00', '700-1.jpg'),
(20, 'Dr.Pudji,Sp.An', 'Spesialis Anestesi', '20.00-20.45', '431-2.jpg'),
(21, 'Dr.Corla,Sp.M', 'Spesialis Mata', '07.00-09.00', '115-5.png'),
(22, 'Dr.Lesti,Sp.JT', 'Jantung & Pembuluh Darah', '09.00-11.00', '126-6.png'),
(23, 'Dr.Indah,SpKJ', 'Spesialis Kedokteran Jiwa', '12.00-16.00', '678-7.jpg');

-----

--
-- Table structure for table `users`
--

CREATE TABLE `users` (
  `name` varchar(20) NOT NULL,
  `username` varchar(20) NOT NULL,
  `email` varchar(20) NOT NULL,
  `password` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data for table `users`
--
```



```

INSERT INTO `users` (`name`, `username`, `email`, `password`) VALUES
('isnaenti', 'ifah', 'isnaenti16@gmail.com',
'$2y$10$iJyS2lcoSkDPYx7QGvAnw.BDpVQcnMeli5uoUFMqqFYrAWgCq4IF6'),
('isnaenti', 'ifah', 'isnaenti34', '12345'),
('ratu', 'ratu', 'ratu@gmail.com',
'$2y$10$32VwjJAXykmHyHb4AePG.5cRDrridD47/TzTsdFeVq.d01AtDZky'),
('ifah', 'ifah', 'isnaenti16@gmail.com',
'$2y$10$.kXFqxo166wrbA9VXKIRZum7tX7TgVMTA/gaLzQMaJ4sFYX.u1hG2'),
(' ', ' ', ' ',
'$2y$10$b0W/teltSB15u0YE3I2PP09H1gSUASNfoT4SsCNH9tmErKy6QQmKq'),
('lisa', 'lisa', 'lisa@gmail.com',
'$2y$10$1907aXNTFRK81Ld/ZgKvqe/M5yXAIz4kPweWLMFhtNJ3aCzSEFz4i'),
('zidan', 'zidan', 'zidan@gmail.com',
'$2y$10$LPtAU3Z7JHiyy6/gf2YL0uvJKh4z0LDKokBWztQPkiX0ltwCqXyhS'),
('zidan', 'zidan', 'zidan@gmail.com',
'$2y$10$aMfNy50F2CDiSwWE/WmRr.LmTt6mCTqzabZ2HpPWQpGwFVrUstlHq'),
('zidan', 'zidan', 'zidan@gmail.com',
'$2y$10$2Shw2Va5e8Che.fl9miwU0jr9pJPQtEuiWqmlCTGait.YkV0Vohki'),
('zidan', 'zidan', 'zidan@gmail.com',
'$2y$10$QGQiG2alAbxYYinWr4HzfeN102hAjncBPk.Ko4vcNyiW.azSgLBp.'),
('zidan', 'zidan', 'zidan@gmail.com',
'$2y$10$TY1qlA8QIY1fJREVDU3MVeUqYpbir4KoE3P1xwxtJJQNGCe1ZxU.yC'),
('zidan', 'zidan', 'zidan@gmail.com',
'$2y$10$W.MS9cviNH.Hu6rX2Z380Z5oT6HNpE9nckAPmSEs.SkPGNWtibTm');

--
-- Indexes for dumped tables
--

--
-- Indexes for table `produk`
--
ALTER TABLE `produk`
  ADD PRIMARY KEY (`id`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `produk`
--
ALTER TABLE `produk`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=28;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

```

```

/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

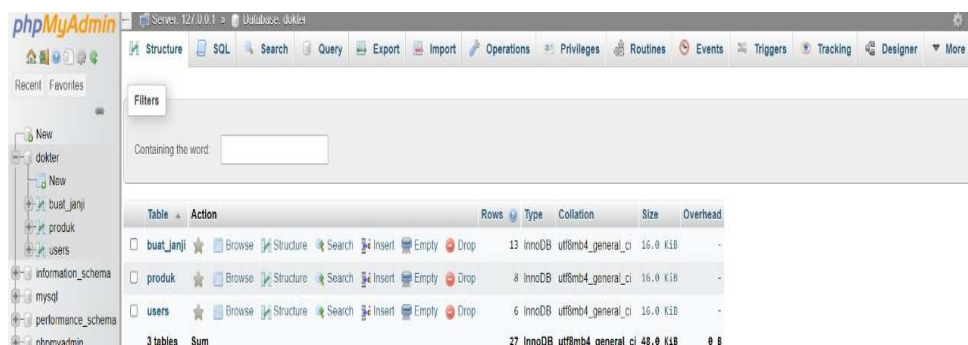
```

### 3.5.2 Struktur

Pada gambar dibawah ini menunjukan tampilan *structure database* dari data aplikasi yang akan dibangun yaitu *Question & Answer* Konsultasi dengan Dokter. *Database* yang dibuat diberi nama “dokter”.



Gambar 3. 7 Database dokter



Gambar 3. 8 Stuktur database

```

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| dokter   |
| information_schema |
| mysql    |
| performance_schema |
| phpmyadmin |
| test     |
+-----+
6 rows in set (0.005 sec)

MariaDB [(none)]> use dokter;
Database changed
MariaDB [dokter]> show tables;
+-----+
| Tables_in_dokter |
+-----+
| buat_janji       |
| produk           |
| users            |
+-----+
3 rows in set (0.001 sec)

MariaDB [dokter]> describe buat_janji;
+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+
| nama_depan | varchar(20) | NO   |     | NULL    |       |
| nama_belakang | varchar(20) | NO   |     | NULL    |       |
| umur       | int(4)      | NO   |     | NULL    |       |
| alamat     | text       | NO   |     | NULL    |       |
| hari       | varchar(10) | NO   |     | NULL    |       |
| dokter     | varchar(25) | NO   |     | NULL    |       |
| keluhan    | varchar(50) | NO   |     | NULL    |       |
+-----+
7 rows in set (0.025 sec)

MariaDB [dokter]> describe users;
+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+
| name       | varchar(20) | NO   |     | NULL    |       |
| username   | varchar(20) | NO   |     | NULL    |       |
| email      | varchar(20) | NO   |     | NULL    |       |
| password   | text       | NO   |     | NULL    |       |
+-----+

```

Gambar 3. 9 Struktur tabel pada database dokter

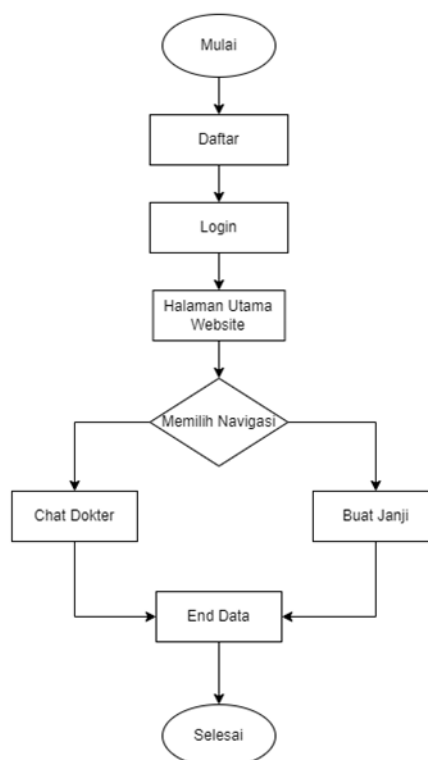


### 3.6 Logika Fungsi atau Algoritma Fungsi

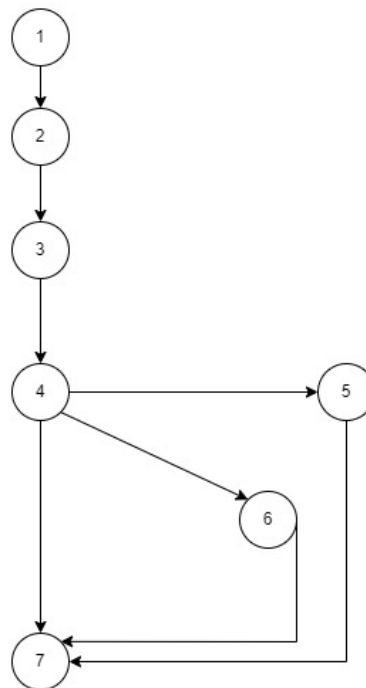
Metode yang digunakan adalah metode *whitebox* dan teknik yang digunakan adalah teknik jalur dasar atau *basis path*. *Whitebox testing* adalah pengujian perangkat lunak pada tingkat alur kode program, apakah masukan dan keluaran yang sesuai dengan spesifikasi yang dibutuhkan [32]. Metode jalur dasar adalah salah satu metode *white box testing*, di mana dalam proses pengujian diperlukan untuk membuat *flowgraph* dari program skrip dan juga menentukan nilai kompleksitas siklomatik [33]. Proses pengujian dilakukan dengan tahapan-tahapan:

1. Mengubah *source code function* cari ke dalam bentuk *flowchart*
2. Mengubah *flowchart* ke dalam bentuk *flowgraph*
3. Penentuan Independen *Path*

Teknik *Basis Path* merupakan salah satu teknik Pengujian *White Box* yang pertama kali dikemukakan oleh Tom McCabe. Teknik ini memungkinkan penguji untuk mengukur kompleksitas logika dari rancangan prosedural. Skenario uji coba yang dibuat untuk menguji teknik *basis path* ini dijamin akan mengeksekusi setiap *statement* dari aplikasi yang diujikan setidaknya satu kali saat tahap pengujian. Berikut dibawah ini adalah gambar dari *flowchart* dan *flowgraph*.



Gambar 3. 13 *Flowchart*

Gambar 3.14 *Flowgraph*

Jalur Independen yang di dapat yaitu :

Jalur 1: 1,2,3,4,5,7

Jalur 2: 1,2,3,4,6,7

Jalur 3: 1,2,3,4,7

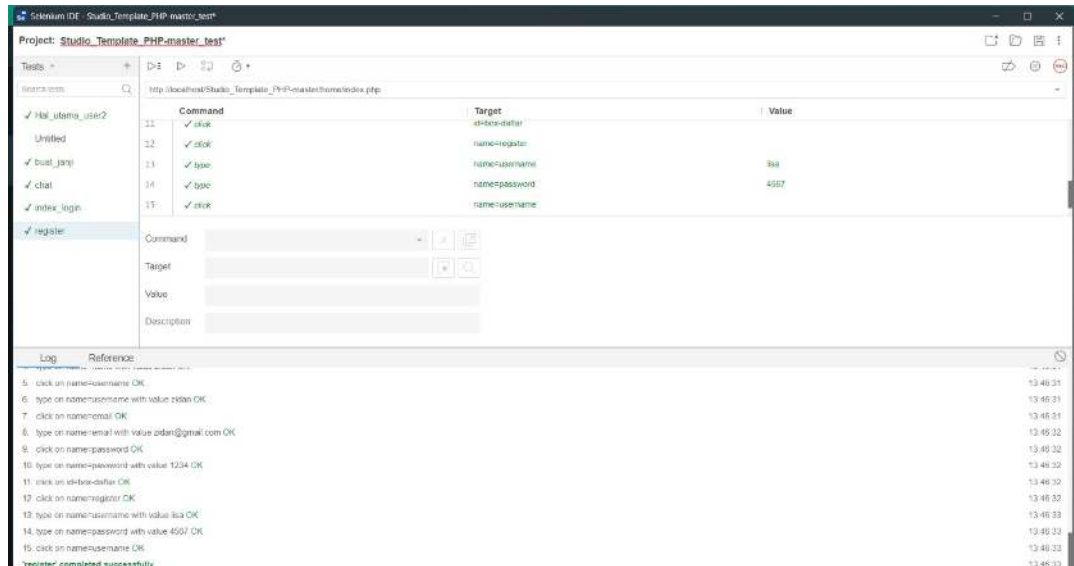
### 3.7 Code Coverage

*Code coverage* adalah salah satu alat ukur utama dalam proses *testing* sebuah *software* yang mengukur seberapa banyak *code* yang telah dibuat dan sudah dilakukan proses *testing*. semakin tinggi *code coverage* dari *software* tersebut, maka menunjukkan proses *testing* yang kita lakukan telah menyeluruh sehingga klien akan menilai bahwa *software* kita telah di-*develope* dan di-*maintain* secara baik serta dapat dipastikan performa *software* kita akan bagus. Penulis menguji *source code* menggunakan *selenium IDE* dan Pengujian *code coverage* dengan *phpunit* pada *visual studio code*.

#### 3.7.1 Pada Tampilan Website

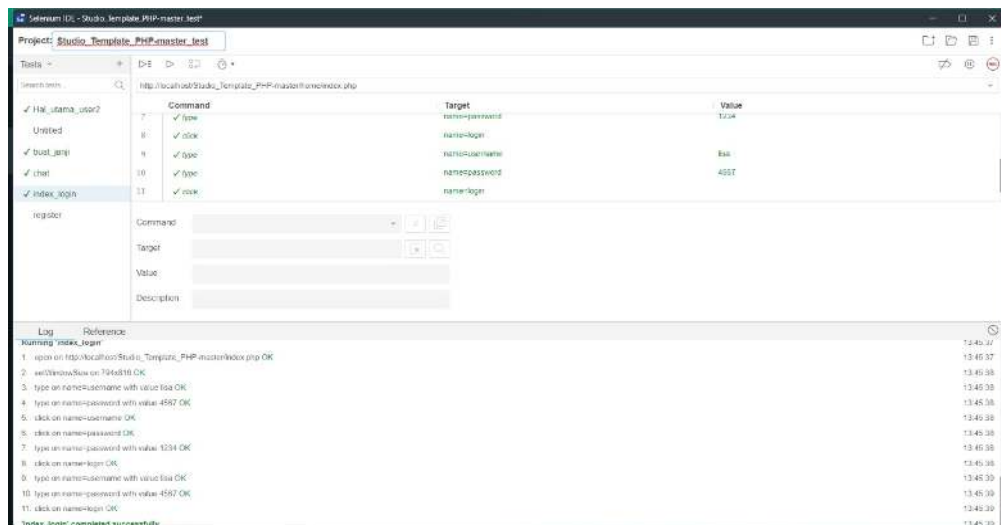
Pada sub Bab ini penulis menunjukan hasil dari pengujian menggunakan *selenium IDE* dan Pengujian *code coverage* dengan *phpunit* pada *visual studio code*.

Penjelasan pengujian pada *register* menggunakan *selenium IDE* yang dimana dapat diperoleh pengecekan bahwa ditandai dengan ceklis berwarna hijau menandakan *source code* tidak terdapat *error*. Pada *source code* daftar (*register*) berisikan nama lengkap, *username*, *email*, dan *password*. Jika diisi sesuai dengan *type* yang ditentukan maka proses daftar berhasil.



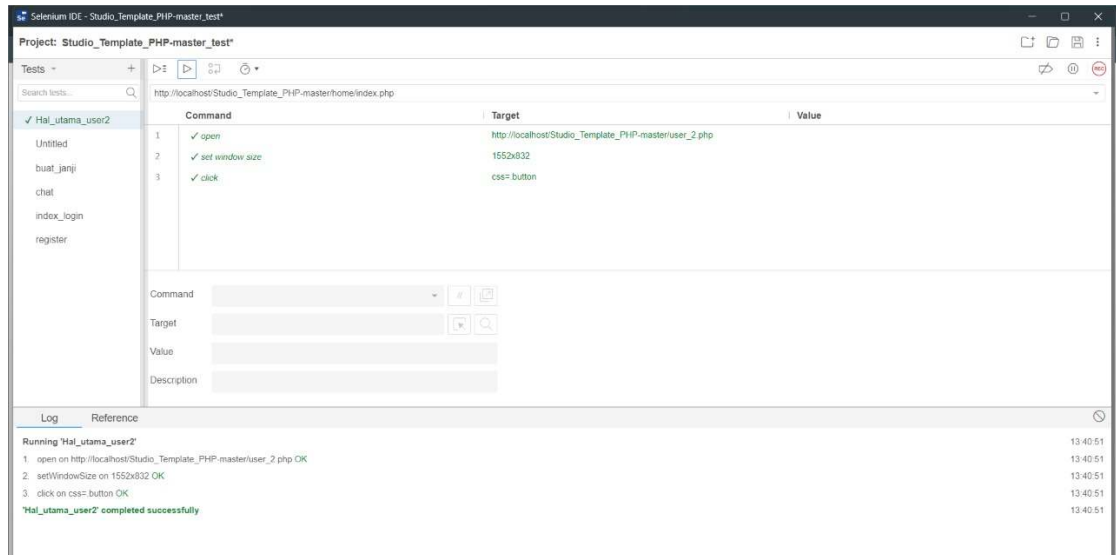
Gambar 3.15 Pengujian daftar (*register*)

Penjelasan pengujian pada *login* menggunakan *selenium IDE* yang dimana dapat diperoleh pengecekan bahwa ditandai dengan ceklis berwarna hijau menandakan *source code* tidak terdapat *error*. User dapat mengisi *username* dan *password* yang sudah di buat pada *form register*.



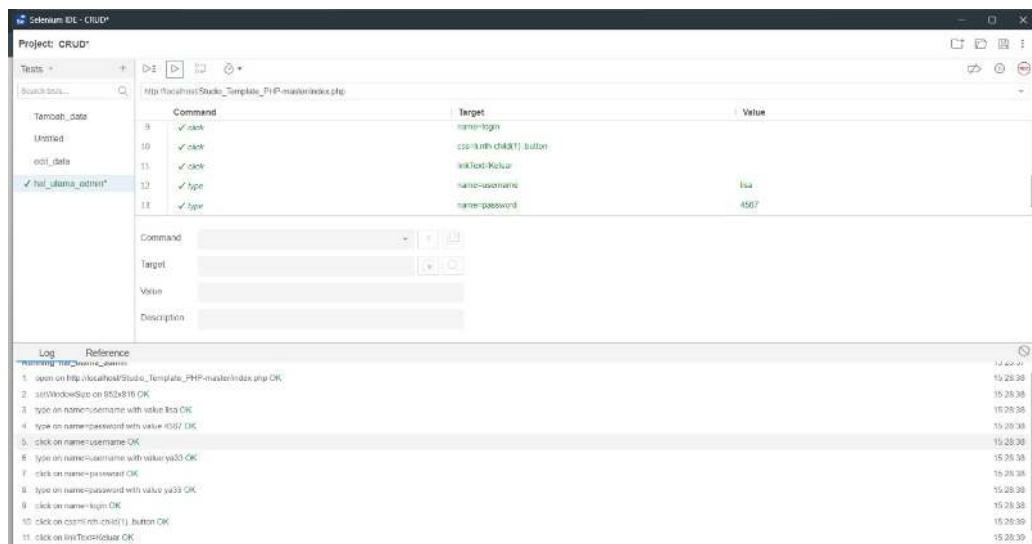
Gambar 3.16 Pengujian *login*

Penjelasan pengujian pada halaman utama *\_user2* menggunakan *selenium IDE* yang dimana dapat diperoleh pengecekan bahwa ditandai dengan ceklis berwarna hijau menandakan *source code* tidak terdapat *error*. Pada *source code* ini menampilkan halaman utama yang terdapat jadwal dokter.



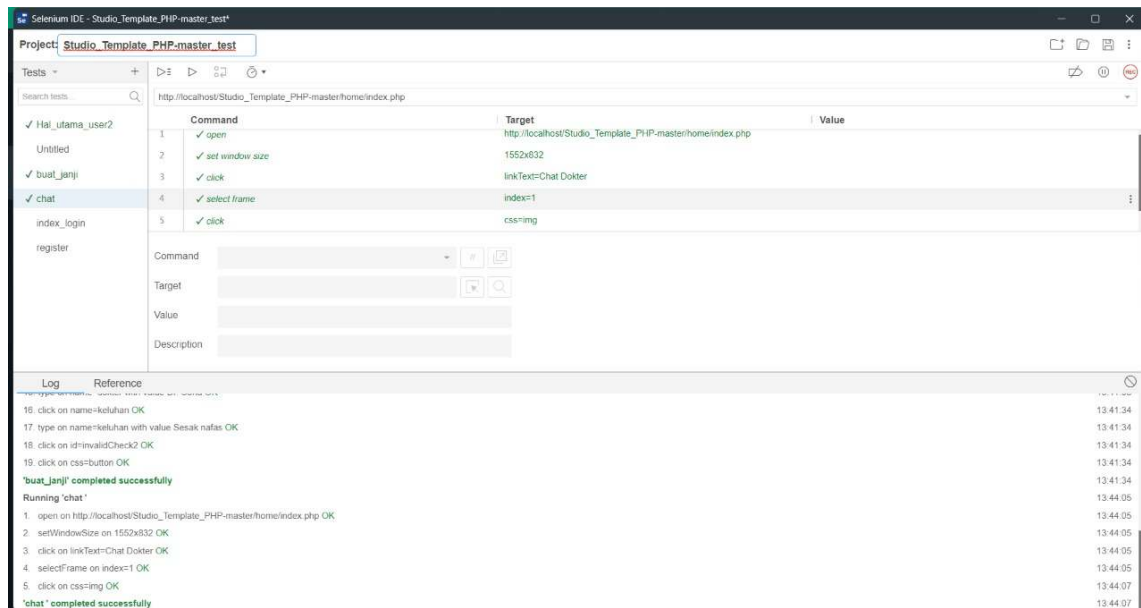
Gambar 3.17 Pengujian halaman utama *\_user2*

Penjelasan pengujian pada halaman utama admin menggunakan *selenium IDE* yang dimana dapat diperoleh pengecekan bahwa ditandai dengan ceklis berwarna hijau menandakan *source code* tidak terdapat *error*. Pada *source code* ini menampilkan halaman utama yang terdapat *CRUD* dan jadwal dokter.



Gambar 3.18 Pengujian halaman utama admin

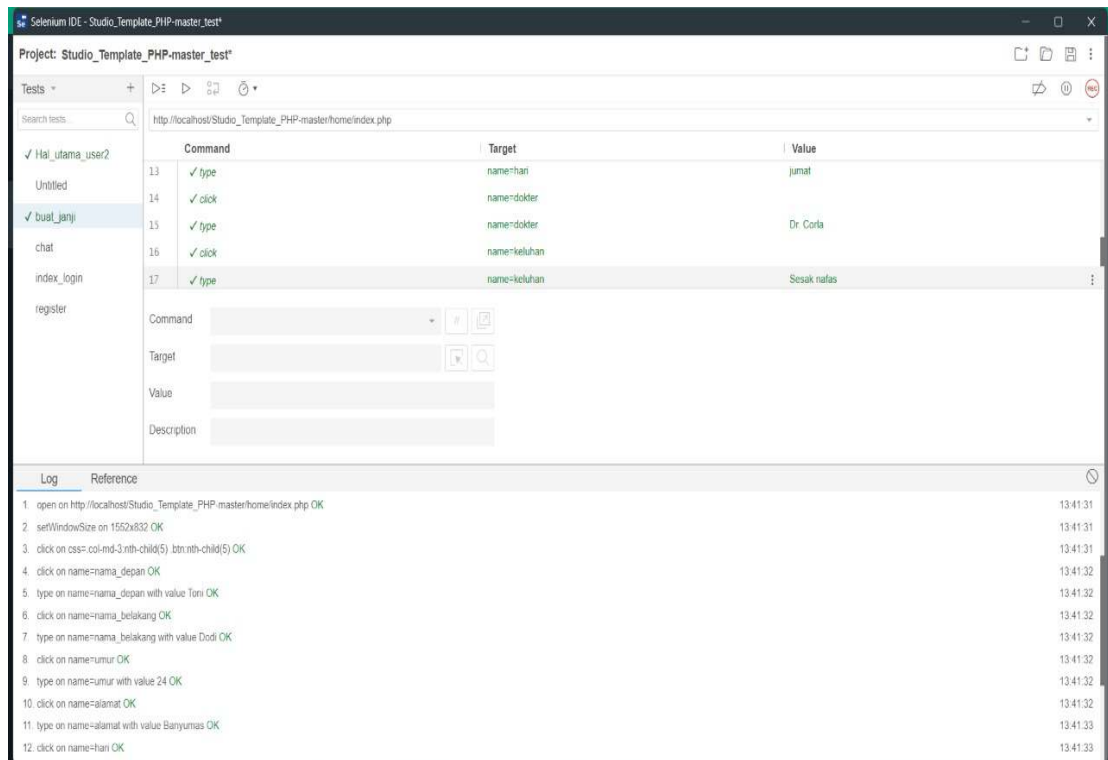
Penjelasan pengujian pada pengujian *chat* menggunakan *selenium IDE* yang dimana dapat diperoleh pengecekan bahwa ditandai dengan ceklis berwarna hijau menandakan *source code* tidak terdapat *error*. Pada *source code* chat terdapat *source code live chat* yang dibuat dari *live chat twak.to* yang digunakan sebagai media komunikasi pada *website* antara *user* (pasien) dan dokter.



Gambar 3.19 Pengujian Chat

Penjelasan pengujian pada *buat\_janji* menggunakan *selenium IDE* yang dimana dapat diperoleh pengecekan bahwa ditandai dengan ceklis berwarna hijau menandakan *source code* tidak terdapat *error*. Pada *source code* *buat janji* terdapat *source code live chat* yang dibuat dengan *live chat twak.to* yang digunakan sebagai media komunikasi dan *buat janji* pada *website* antara *user* (pasien) dan dokter.

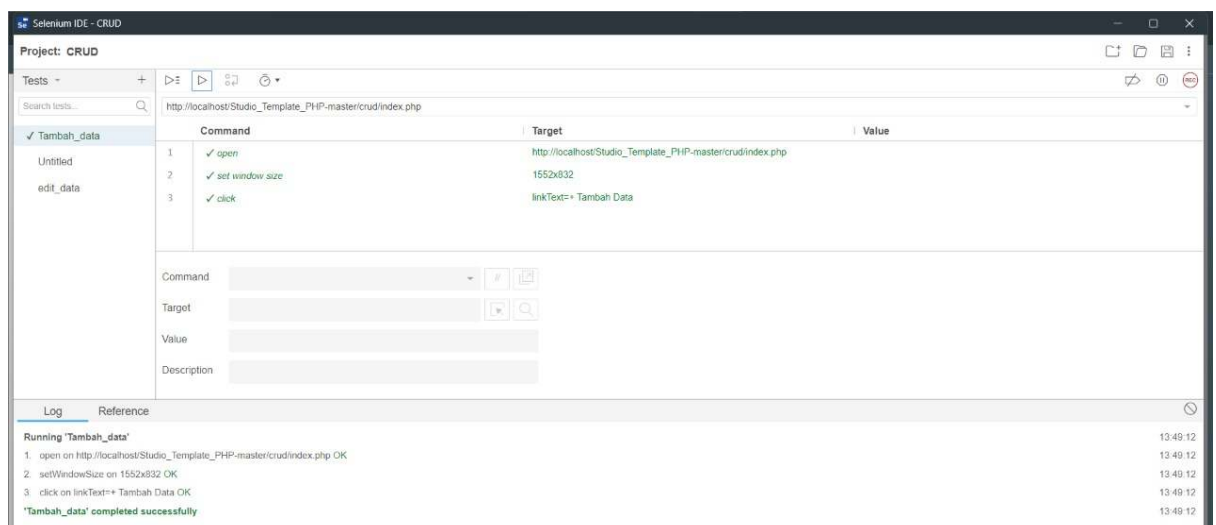




Gambar 3. 20 Pengujian buat\_janji

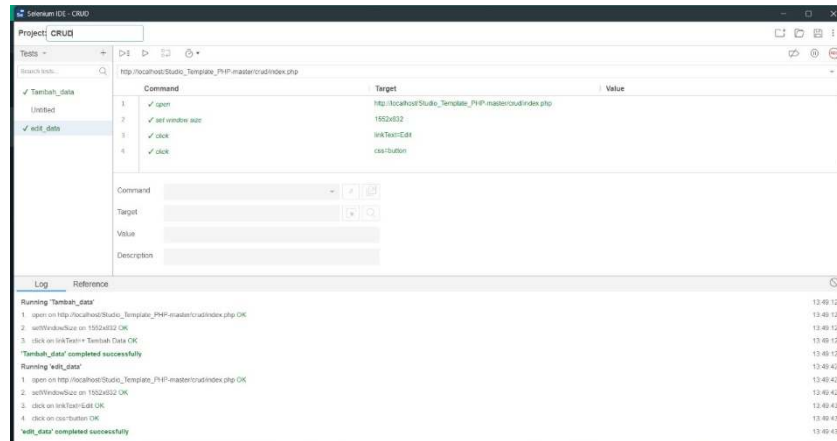
### 3.7.2 Tampilan pada *CRUD*

Penjelasan pengujian pada tambah\_data menggunakan *selenium IDE* yang dimana dapat diperoleh pengecekan bahwa ditandai dengan ceklis berwarna hijau menandakan *source code* tidak terdapat *error*. Pada *source code* tambah data berfungsi untuk menambahkan data dokter seperti nama, jam praktek dokter dan foto dokter.



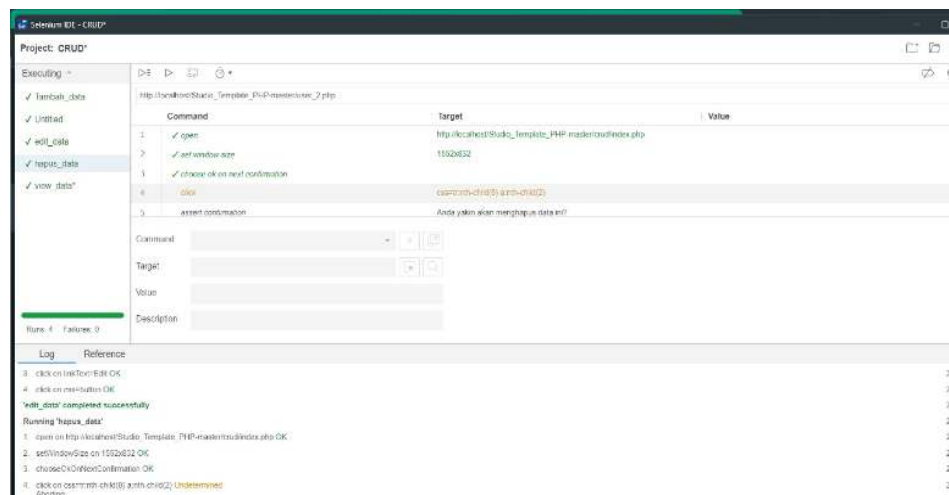
Gambar 3.21 pengujian tambah\_data

Penjelasan pengujian pada `edit_data` menggunakan *selenium IDE* yang dimana dapat diperoleh pengecekan bahwa ditandai dengan ceklis berwarna hijau menandakan *source code* tidak terdapat *error*. Pada *source code* edit data berfungsi untuk mengedit (mengubah) data dokter seperti nama, jam praktek dokter dan foto dokter.



Gambar 3.22 pengujian `edit_data`

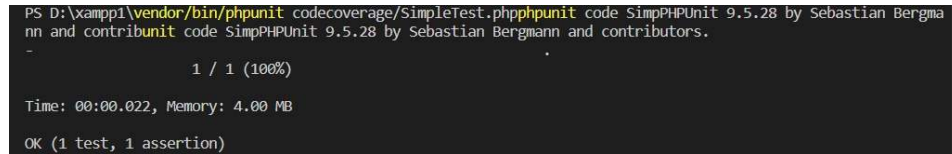
Penjelasan pengujian pada `hapus_data` menggunakan *selenium IDE* yang dimana dapat diperoleh pengecekan bahwa ditandai dengan ceklis berwarna hijau menandakan *source code* tidak terdapat *error*. Pada *source code* delete data berfungsi untuk menghapus data dokter seperti nama, jam praktek dokter dan foto dokter.



Gambar 3.23 pengujian `hapus_data`

Berikut hasil *code coverage* pada *website* penulis:

Setelah pembuatan *Website Question & Answer Konsultasi Dokter* dibangun. Maka selanjutnya penulis melakukan pengujian terhadap *source code* tersebut. Berikut di bawah ini penulis menunjukkan hasil pengujian *code coverage*.



```
PS D:\xampp1\vendor\bin\phpunit codecoverage\SimpleTest.phpphpunit code SimpHPUnit 9.5.28 by Sebastian Bergma
nn and contribunit code SimpHPUnit 9.5.28 by Sebastian Bergmann and contributors.
-
1 / 1 (100%)
Time: 00:00.022, Memory: 4.00 MB
OK (1 test, 1 assertion)
```

Gambar 3.24 Pengujian Code Coverage

Dapat dilihat bahwa *folder code coverage* dengan *file SimpleTest.php* menunjukkan bahwa hasil pengujian diperoleh *presentase* 100%.

## BAB IV IMPLEMENTASI DAN PENGUJIAN

### 4.1 Pembahasan Hasil Implementasi

Lingkungan implementasi berisi *CRUD* yang dikelola admin dapat berfungsi mengatur proses pada “Jadwal Dokter” dan terdapat fitur “*chat* atau buat janji” untuk mempermudah *user* (pasien) konsultasi secara *online*.

Bab ini merupakan implementasi dari “MEMBANGUN APLIKASI *HALODOC*”. Implementasi merupakan tahapan pembuatan program sesuai rancangan yang dilakukan dan yang telah ditemukan. Tahap pengujian dilakukan setelah semua proses pembuatan *website* selesai dilakukan hasil pengujian dapat dijadikan sebagai bukti atau bentuk keberhasilan dan mengimplementasikan hasil perancangan dengan mengidentifikasi dan melakukan analisis terhadap hasil pengujian. Hasil pengujian dapat digunakan untuk acuan dalam melakukan pengembangan di masa yang akan datang. *Website* yang akan penulis bangun yaitu *Question & Answer* Konsultasi Dokter. Penulis membangun *website* ini terinspirasi dari *Halodoc*. Langkah selanjutnya penulis akan menganalisis *source code* pada *website* yang penulis buat.

#### 4.1.1 Analisis Daftar (Register)

Pada analisis *Source code* dibawah ini yaitu Daftar (*register*)

```
<?php
require("koneksi.php");
if (isset($_POST['register'])) {
    // filter data yang diinputkan
    $name = filter_input(INPUT_POST, 'name', FILTER_SANITIZE_STRING);
    $username = filter_input(INPUT_POST, 'username',
FILTER_SANITIZE_STRING);
    // enkripsi password
    $password = password_hash($_POST["password"], PASSWORD_DEFAULT);
    $email = filter_input(INPUT_POST, 'email', FILTER_VALIDATE_EMAIL);
    // menyiapkan query
    $sql = "INSERT INTO users (name, username, email, password)
VALUES ('$name', '$username', '$email', '$password')";
    $query = mysqli_query($koneksi, $sql);
    if ($query) {
        // jika berhasil tampilkan pesan berhasil insert
        header("Location: index.php");
    } else {
```

```

        // jika gagal tampilkan pesan kesalahan
        die("Gagal menyimpan perubahan...");
    }
}

?>
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Halaman Daftar</title>

    <link rel="stylesheet" href="css/bootstrap.min.css" />
</head>
<body class="bg-light">
    <div class="container mt-5">
        <p>&larr; <a href="index.php">Keluar</a>
        <div class="row">
            <div class="col-md-6">
                <form action="" method="POST">
                    <br>
                    <br>
                    <div id="box-daftar">
                        <center>
                            <h1>Daftar</h1>
                        </center>
                        <br>
                        <div class="form-group">
                            <label for="name">Nama Lengkap</label>
                            <input class="form-control" type="text"
name="name" placeholder="Nama kamu" />
                        </div>

                        <div class="form-group">
                            <label for="username">Username</label>
                            <input class="form-control" type="text"
name="username" placeholder="Username" />
                        </div>

                        <div class="form-group">
                            <label for="email">Email</label>

```

```

        <input class="form-control" type="email"
name="email" placeholder="Alamat Email" />
    </div>

    <div class="form-group">
        <label for="password">Password</label>
        <input class="form-control" type="password"
name="password" placeholder="Password" />
    </div>
    <br>
    <center>

        <input type="submit" class="btn btn-success btn-
block" name="register" value="Selesai" />
    </center>
</form>
</div>
</div>
</div>
</div>
</body>

<style>
    * {
        padding: 0;
        margin: 0;
        font-family: 'Times New Roman', Times, serif;
    }
    body {
        background-color: whitesmoke;
    }
    .row {
        display: flex;
        height: 100px;
        justify-content: center;
        align-items: center;
    }
    #box-daftar {
        justify-content: center;
        width: 300px;
        min-height: 200px;
        border: 5px solid;
        background-color: lightblue;
        padding: 50px;
        box-sizing: border-box;
    }
</style>

```

```
</html>
```

### Penjelasan Analisis *Source Code* Daftar (*Register*)

Diatas adalah *source code* dari daftar dalam *source code* ini bertugas untuk membuat akun *user* yang ingin mengunjungi *website*.

#### 4.1.2 Analisis *Login*

Pada analisis *Source code* dibawah ini yaitu *Login*

```
<?php
require("koneksi.php");

if (isset($_POST['login'])) {

    $username = filter_input(INPUT_POST, 'username',
FILTER_SANITIZE_STRING);
    $password = filter_input(INPUT_POST, 'password',
FILTER_SANITIZE_STRING);

    $sql = "SELECT * FROM users WHERE username='$username' OR
email='$username'";
    $user = mysqli_query($koneksi, $sql);
    $row = mysqli_fetch_assoc($user);

    // jika user terdaftar
    if ($row) {
        // verifikasi password
        if (password_verify($password, $row["password"])) {
            // buat Session
            session_start();
            $_SESSION["user"] = $row["username"];
            $_SESSION["email"] = $row["email"];

            // login sukses, alihkan ke halaman timeline
            header("Location: user_2.php");
        }
    }

    else{
        $list_user = [
            [
                'username' => 'john',
                'password' => '123456'
            ],
            [
```

```

        'username' => 'ya33',
        'password' => 'ya33'
    ]
];

//dapatkan data user dari form
$user = [
    'username' => $_POST['username'],
    'password' => $_POST['password'],
];

// login sukses, alihkan ke halaman timeline
header("Location: indexv2.php");
}
}
?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Login Rumah Sakit RAIS</title>

    <link rel="stylesheet" href="css/bootstrap.min.css" />
</head>

<body class="bg-light">

    <div class="container mt-5">
        <div class="row">
            <div class="col-md-6">

<center>
        <form action="" method="POST">

```



```

        <br>
        <br>
        <br>
        <br>
        <br>
        <br>
        <br>
        <br>
        <div class="bg-login">
            <div id="box-login">
                <h1>Login</h1>
                <br>
                <div class="form-group">
                    <label for="username">Username</label>
                    <input class="form-control" type="text"
name="username" placeholder="Username atau email" />
                </div>
                <div class="form-group">
                    <label for="password">Password</label>
                    <input class="form-control"
type="password" name="password" placeholder="Password" />
                </div>
                <br>
                <input type="submit" class="btn btn-success
btn-block" name="login" value="Masuk" />
                <p>Jika Belum Punya Akun Daftar Terlebih
Dahulu</p>
                <a href="register.php">Daftar di sini</a>
            </div>
        </div>
    </div>
    <div class="col-md-6">
    </div>
    </div>
    </div>
</body>
</html>
<style>
    * {
        padding: 0;
        margin: 0;
        font-family: 'Times New Roman', Times, serif;
    }
    body {

```

```

        background-color: whitesmoke;
    }
    .bg-login {
        display: flex;
        height: 100px;
        justify-content: center;
        align-items: center;
    }
    #box-login {
        justify-content: center;
        width: 300px;
        min-height: 200px;
        border: 5px solid;
        background-color: lightblue;
        padding: 15px;
        box-sizing: border-box;
    }
</style>

```

#### 4.1.3 Analisis Jadwal Dokter

Pada analisis *Source code* dibawah ini yaitu Jadwal Dokter

```

<?php
require('../koneksi.php');

$sql = 'SELECT * FROM produk';
$query = mysqli_query($koneksi, $sql);
$products = mysqli_fetch_all($query, MYSQLI_ASSOC);

?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <link rel="stylesheet" href="../assets/css/bootstrap.min.css"
crossorigin="anonymous">
    <title>Tampilan_Dokter</title>
</head>

<body>
    <nav class="navbar navbar-expand-lg navbar-light bg-light">

```

```

        <a class="navbar-brand" href="#">Konsultasi dengan Dokter dan
membuat janji dengan dokter </a>
        <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarSupportedContent" aria-
controls="navbarSupportedContent" aria-expanded="false" aria-
label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>

        <div class="collapse navbar-collapse"
id="navbarSupportedContent">
            <ul class="navbar-nav mr-auto">
                <!-- <li class="nav-item active">
                    <a class="nav-link" href="#">Home <span class="sr-
only">(current)</span></a>
                </li>-->
                <!--<li class="nav-item">
                    <a class="nav-link" href="#">Link</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link disabled">Disabled</a>
                </li>-->
            </ul>
            <form class="form-inline my-2 my-lg-0">
                <button class="btn btn-outline-success my-2 my-sm-0"
type="submit"><a href="../index.php">keluar</a></button>
            </form>
        </div>
    </nav>

    <div class="container">
        <div class="row">
            <div class="col">
                <hr>
                <h3 class="text-center">Jadwal Dokter</h1>
                <hr>
            </div>
        </div>
        <div class="row">
            <?php foreach ($products as $product) : ?>
                <div class="col-md-3">
                    <div class="card" style="width: auto;">
                        
                        <div class="card-body">
                            <h5 class="card-title"><?php echo
$product['nama_produk'] ?></h5>

```

```

                                <p class="card-text"><?php echo
$product['deskripsi'] ?></p>
                                <p class="card-text"><?php
echo($product['harga_jual']);?></p>
                                <a class="btn btn-primary" href="chat.php"
role="button">Chat Dokter</a>
                                <a class="btn btn-primary" href="janji.php"
role="button">Buat Janji</a>

                                </div>
                            </div>
                        </div>
                    <?php endforeach; ?>
                </div>
            </div>

            <script src="../assets/js/jquery.min.js"></script>
            <script src="../assets/js/bootstrap.bundle.js"
crossorigin="anonymous"></script>
        </body>
    </html>

```

Penjelasan *Source Code* Jadwal Dokter yaitu tampilan mengenai jadwal dokter dan nama dokter. Pada *Source Code* jadwal dokter dibuatkan dari referensi melalui *bootstrap* yang dimana terdapat *navbar* dan *button* yang dibuatkan. `<a class="btn btn-primary" href="chat.php" role="button">Chat Dokter</a>` `<a class="btn btn-primary" href="janji.php" role="button">Buat Janji</a>`. *Button* pada *chat* dokter dan buat janji diberikan warna biru (*Primary*).

#### 4.1.4 Analisis *Chat* Dengan Dokter

Pada analisis *Source code* dibawah ini yaitu *Chat* dengan Dokter

```

<!doctype html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-
scale=1">
        <title>Konsultasi</title>
    </head>

    <!--Start of Tawk.to Script-->
    <script type="text/javascript">

```

```

var Tawk_API=Tawk_API||{}, Tawk_LoadStart=new Date();
(function(){
var
s1=document.createElement("script"),s0=document.getElementsByTagName("s
cript")[0];
s1.async=true;
s1.src='https://embed.tawk.to/63c256e8c2f1ac1e202d62a7/1gmni70p3';
s1.charset='UTF-8';
s1.setAttribute('crossorigin','*');
s0.parentNode.insertBefore(s1,s0);
})();
</script>
<!--End of Tawk.to Script-->

<body>
  <h1>Silakan Anda Bisa Melakukan Konsultasi Dengan Dokter</h1>
  
  <p>Anda bisa melakukan konsultasi mengenai keluhan penyakit anda
pada fitur live chat disebelah kanan</p>

</body>

</html>

```

Penjelasan *Source Code Chat* dengan Dokter yaitu tampilan dimana *user* (pasien) dapat berkonsultasi melalui *online* dengan menggunakan media *live chat* dari *twak.to*.

#### 4.1.5 Analisis Buat Janji Dengan Dokter

Pada analisis *Source code* dibawah ini yaitu Buat janji dengan dokter

```

<?php
require('../koneksi.php');
?><!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
    <title>Janji Dokter</title>
  </head>

  <!--Start of Tawk.to Script-->
  <script type="text/javascript">
var Tawk_API=Tawk_API||{}, Tawk_LoadStart=new Date();
(function(){

```

```

var
s1=document.createElement("script"),s0=document.getElementsByTagName("s
cript")[0];
s1.async=true;
s1.src='https://embed.tawk.to/63c256e8c2f1ac1e202d62a7/1gmni70p3';
s1.charset='UTF-8';
s1.setAttribute('crossorigin','*');
s0.parentNode.insertBefore(s1,s0);
})();
</script>
<!--End of Tawk.to Script-->

<body>
  <h1>Silakan Anda Bisa Membuat Janji dengan dokter setelah melakukan
kesepakatan dengan dokter</h1>
  
  <p>Anda bisa membuat janji dengan dokter pada fitur live chat
disebelah kanan </p>
  <br>
<center>
  <form method="POST" action="crud_janji/proses_janji.php"
enctype="multipart/form-data" >
    <section class="base">
      <div id="box-login">
        <div class="form-group">
          <label>Nama Depan</label>
          <input type="varchar" name="nama_depan" />
        </div>
        <div>
          <div class="form-group">
            <label>Nama Belakang</label>
            <input type="varchar" name="nama_belakang" />
          </div>
          <div class="form-group">
            <label>Umur</label>
            <input type="int" name="umur" />
          </div>
          <div class="form-group">
            <label>Alamat</label>
            <input type="text" name="alamat" />
          </div>
          <div class="form-group">
            <label>hari</label>
            <input type="varchar" name="hari" required="" />
          </div>
          <div class="form-group">
            <label>Dokter</label>

```

```

        <input type="varchar" name="dokter" required="" />
    </div>
    <div class="form-group">
        <label>Keluhan</label>
        <input type="varchar" name="keluhan" required="" />
    </div>
    <div class="col-12">
    <div class="form-check">
        <input class="form-check-input" type="checkbox" value=""
id="invalidCheck2" required>
        <label class="form-check-label" for="invalidCheck2">
            Ceklis jika data anda sudah lengkap
        </label>
    </div>
    <div>
        <button type="submit">Selesai</button>
    </div>
    </div>
    </section>
</form>
</center>
</body>
<style>
    #box-login {
        justify-content: center;
        width: 500px;
        min-height: 200px;
        border: 5px solid;
        background-color: lightblue;
        padding: 30px;
        box-sizing: border-box;
    }
</style>
</html>

```

Penjelasan *source code* buat janji dengan dokter yaitu tampilan dimana *user* setelah mendapatkan kesepakatan untuk konsultasi secara langsung dengan dokter maka *user* (pasien) dapat mengisi data diri pada kolom yang tersedia. Setelah itu *user* (pasien) dapat mengirimkan pesan kepada dokter untuk penjelasan lebih lanjut melalui *live chat* dari *twak.to*.

## 4.2 Pengujian dan Hasil Pengujian

### 4.2.1 Hasil Tampilan Pada Daftar (Register)

Terlihat tampilan dari daftar (*register*) yaitu dimana *user* dapat mengisi data pada *form* tabel untuk dapat memiliki akun agar dapat mengunjungi *website*.



**Daftar**

Nama Lengkap  
ratu

Username  
ratu

Email  
ratu@gmail.com

Password  
Password

Selesai

Gambar 4. 1 Daftar (*register*)

### 4.2.2 Hasil Tampilan Pada Login

Terlihat tampilan dari *login* yaitu dimana *user* setelah membuat akun pada halaman daftar, Kemudian memasukan *username* dan *password* pada halaman *login*. Jikan *user* melakukan kesalahan saat mengisi *username* dan *password* tidak akan bisa mengunjungi *website* tersebut.



**Login**

Username ratu

Password Password

Masuk

Jika Belum Punya Akun Daftar Terlebih Dahulu  
[Daftar di sini](#)

Gambar 4.2 Login user

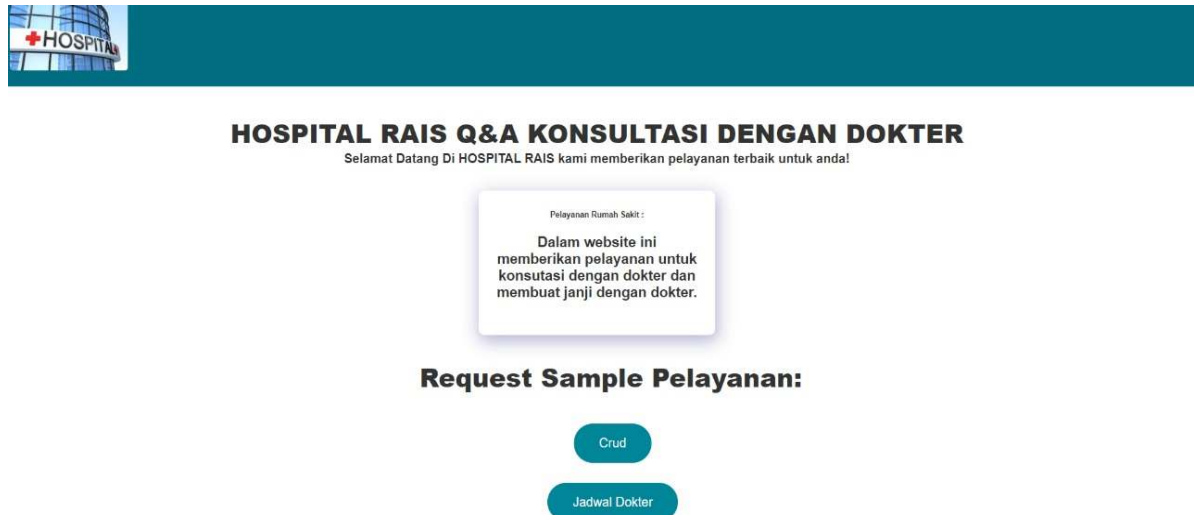



Gambar 4.3 *Login* admin

#### 4.2.3 Hasil Tampilan Pada Halaman Utama Admin

Terlihat tampilan dari halaman utama admin yaitu terdapat fitur *CRUD* dan jadwal dokter. Dimana admin dapat mengelola data dokter melalui fitur *CRUD*, serta admin dapat melihat jadwal dokter yang di dalamnya terdapat fitur *chat* dan buat janji dengan dokter.

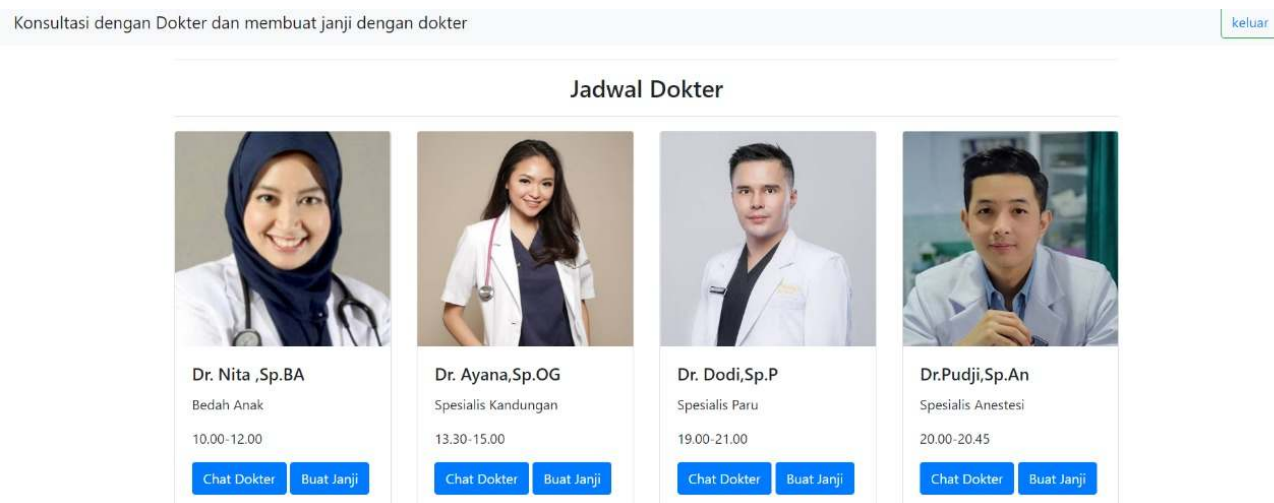
Gambar 4.4 Halaman utama *\_user* (pasien)



Gambar 4.5 Halaman utama admin

#### 4.2.4 Hasil Tampilan Pada Jadwal Dokter

Terlihat tampilan dari jadwal dokter yaitu dimana admin, *user*, dan dokter dapat melihat jadwal dokter dan memilih dokter sesuai tujuan untuk konsultasi, selanjutnya memilih fitur *chat* atau buat janji kesepakatan dengan dokter.



Gambar 4.6 Jadwal dokter

#### 4.2.5 Hasil Tampilan Pada Chat Dokter

Terlihat tampilan dari *chat* dokter yaitu dimana *user* dapat berkonsultasi mengenai keluhan atau rawat jalan dengan dokter melalui *website* ini dengan memilih fitur *chat* dokter. Kemudian dokter akan membalas pesan yang di sampaikan oleh *user* pada fitur yang sama.

Silakan Anda Bisa Melakukan Konsultasi Dengan Dokter



Anda bisa melakukan konsultasi mengenai keluhan penyakit anda pada fitur live chat di sebelah kanan

Mohon isi formulir di bawah dan kami akan membalasnya sesegera mungkin.

\* Nama

\* Email

\* Pesan

Add live chat to your site

Gambar 4.7 Chat dokter

#### 4.2.6 Hasil Tampilan Pada Buat Janji Dengan Dokter

Terlihat tampilan dari buat janji dengan dokter yaitu dimana *user* dapat membuat janji atau kesepakatan dengan dokter yang telah dibicarakan sebelumnya. Setelah itu *user* dapat mengisi data diri pasien pada kolom yang tersedia.

Silakan Anda Bisa Membuat Janji dengan dokter setelah melakukan kesepakatan dengan dokter



Anda bisa membuat janji dengan dokter pada fitur live chat di sebelah kanan

Mohon isi formulir di bawah dan kami akan membalasnya sesegera mungkin.

\* Nama

\* Email

\* Pesan

Gambar 4.8 Buat janji dengan dokter

Nama Depan

Nama Belakang

Umur

Alamat

hari

Dokter

Keluhan

☐ Ceklis jika data anda sudah lengkap

Gambar 4. 9 Form data diri pasien

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan dari aplikasi *halodoc* konsultasi kesehatan sebelumnya dapat dilihat dari tampilan *chat* dokter dan buat janji dapat dibedakan dengan *website Q&A* konsultasi dengan dokter yang penulis buat masih sangat sederhana. Maka kesimpulan yang dapat ditarik untuk diterapkan pada laporan ini adalah:

1. Website yang penulis bangun yaitu *Q&A* konsultasi dokter sudah dapat digunakan masyarakat untuk konsultasi dengan dokter secara online.
2. Proses *Q&A* konsultasi dan membuat janji yang penulis bangun, yaitu menggunakan fitur *live chat twak.to*. Dengan ini masyarakat dapat menghubungi atau bertanya kepada dokter mengenai keluhan dan bisa membuat janji dengan dokter tersebut untuk melakukan konsultasi secara langsung.

#### **5.2 Saran**

Proyek ini tidak luput dari kesalahan maka saran akan dapat diberikan berdasarkan masukan penguji dan pembimbing.

## DAFTAR PUSTAKA

- [1] C. S. Putra, "PERANAN TEKNOLOGI INFORMASI DALAM PELAYANAN," *Jurnal SIMTIKA*, vol. Volume 2, pp. 28-31, 2019.
- [2] S. Siyen, "Faktor Yang Berhubungan Dengan Mutu Pelayanan Rumah Sakit Bhayangkara," *MPPKI*, vol. vol 3, pp. 267-274, 2020.
- [3] R. H. d. A. Ashari, "Penerapan Teknologi Web Service," *Penerapan Teknologi Web Service*, vol. vol 23, pp. 64-77, 2017.
- [4] R. Somyal and T. M. E. Nathanael, "PENGEMBANGAN SISTEM INFORMASI PELATIHAN BERBASIS WEB," *Jurnal TECHNO Nusa Mandiri*, vol. vol 16, pp. 51-58, 2019.
- [5] D. P. Evert, "KOMUNIKASI INTERPERSONAL DALAM," *Jurnal Penelitian dan Studi Ilmu Komunikasi*, vol. vo 1, pp. 127-136, 2020.
- [6] M. D. Ria, "Jurnal Informatika dan Rekayasa Perangkat Lunak (JATIKA)," *PERANCANGAN SISTEM INFORMASI TATA KELOLA TEKNOLOGI*, vol. vol 2, pp. 122-133, 2021.
- [7] T. Y. Ka, "DECISION SUPPORT SYSTEM FOR SELECTING DOCTORS IN APPLICATION X," *Jurnal Teknik Informatika (JUTIF)*, vol. vol 3, pp. 857-862, 2022.
- [8] M. D. Ria1, "PERANCANGAN SISTEM INFORMASI TATA KELOLA TEKNOLOGI," *Jurnal Informatika dan Rekayasa Perangkat Lunak (JATIKA)*, vol. vol 2, pp. 122-133, 2021.
- [9] R. Sidiq, "KAJIAN EFISIENSI PELAYANAN RUMAH SAKIT," *Idea Nursing Journal*, vol. vol 8, pp. 29-34, 2017.
- [10] Khainuddin, "Analisis Kualitas Pelayanan Publik Rawat Inap Pada Badan," *Jurnal Ilmiah Magister Administrasi Publik*, vol. vol 1, pp. 22-32, 2020.
- [11] Khainuddin, "Analisis Kualitas Pelayanan Publik Rawat Inap Pada Badan," *Jurnal Ilmiah Magister Administrasi Publik*, Vols. 22-32, p. vol 1, 2020.
- [12] M. Baharuddin, "Tinjauan Etik Regulasi Jam Kerja Dokter," *Jurnal Etika Kedokteran Indonesia*, vol. vol 1, pp. 25-29, 2017.
- [13] I. M. I. Wirayuda, "Aplikasi Tawk.to Sebagai Pendukung," *Media Informasi*, vol. vol 31, pp. 9-20, 2022.
- [14] E. Febriyanto, "Penerapan Aplikasi Tawk To IduHelp! Sebagai Sarana Pelayanan," *Jurusan Sistem Informasi*, vol. vol 3, pp. 246-259, 2019.

- [15] W. Puiji1, "Aplikasi Peramalan Persediaan Bahan Baku Kain," *JURNAL ALGOR*, vol. vol 1, pp. 37-43, 2020.
- [16] F. Ayu, "PERANCANGAN SISTEM INFORMASI PENGOLAHAN DATA PRAKTEK KERJA," *Jurnal Intra-Tech*, vol. vol 2, pp. 12-26, 2018.
- [17] A. Wibawanto, "PENGUNAAN INTERNET DALAM PERPUSTAKAAN," *penggunaan internet dalam perpustakaan*, vol. vol 10, pp. 192-203, 2018.
- [18] P. S. P. d. Rahadi, "PERANCANGAN SISTEM INFORMASI TOKO ONLINE BERBASIS WEB," *Jurnal Ilmiah Indonesia p*, vol. vol 5, pp. 1657-1668, 2020.
- [19] A. Y. Permana, "PERANCANGAN SISTEM INFORMASI PENJUALAN PERUMAHAN MENGGUNAKAN METODE," *SIGMA – Jurnal Teknologi Pelita Bangsa*, vol. vol 10, pp. 153-167, 2019.
- [20] N. A. Ramdhan, "RANCANG BANGUN DAN IMPLEMENTASI SISTEM," *Jurnal Ilmiah Intech : Information Technology Journal of UMUS*, vol. vol 1, pp. 1-12, 2019.
- [21] N. Hasanah, "IMPLEMENTATION OF DECISION SUPPORT SYSTEM WITH SMART METHOD IN," *Jurnal Teknik Informatika (JUTIF)*, vol. vol 3, pp. 611-618, 2022.
- [22] M. Rapita, "PERANCANGAN SISTEM INFORMASI DATA," *Teknologiterkini.org*, vol. vol 2, pp. 1-13, 2022.
- [23] R. Nirsal, "DESAIN DAN IMPLEMENTASI SISTEM PEMBELAJARAN BERBASIS E-LEARNING," *Jurnal Ilmiah d'Computare V*, vol. vol 10, pp. 30-37, 2020.
- [24] U. J. Agustiani, "SISTEM INFORMASI PENDAFTARAN DI RSIA," *JurnalInformatika dan Komputer* , vol. vol 12 , pp. 146-154, 2021.
- [25] A. Josi, "PENERAPAN METODE PROTOTIPING DALAM PEMBANGUNAN WEBSITE," *JTI*, vol. vol 9, pp. 50-57, 2017.
- [26] H. Maulana, "ANALISIS DAN PERANCANGAN SISTEM REPLIKASI DATABASE," (*Jurnal Nasional Informatika dan Teknologi Jaringan*), vol. vol 1, pp. 32-37, 2017.
- [27] S. Reddy, "CoQA: A Conversational Question Answering Challenge," *Transactions of the Association for Computational Linguistics*, vol. vol 7, pp. 249-266, 2019.

- [28] G. d. G. Lovina, "QUESTION ANSWERING SYSTEM DAN PENERAPANNYA PADA," *Jurusan Teknik Informatika*, vol. vol 7, pp. 1-9, 2017.
- [29] L. A. Faisal, "RANCANG BANGUN APLIKASI PENGGAJIAN KARYAWAN," *Jurnal MediaTIK*, vol. vol 2, pp. 1-8, 2020.
- [30] M. Y. Simargolang, "ANALISIS SISTEM PENGOLAHAN ABSENSI KARYAWAN," (*Jurnal Teknologi Informasi*), vol. vol 1, pp. 144-123, 2017.
- [31] Chamelia Sri Sejati, "Aplikasi Kearsipan Berbasis Web Di Dinas Pekerjaan Umum Dan," *JURNAL DATA SCIENCE & INFORMATIKA* , vol. vol 2, pp. 31-35, 2022.
- [32] Mardinawati, "PENGEMBANGAN SISTEM INFORMASI PENJADWALAN MATA KULIAH," *Jurusan Akuntansi, Politeknik Negeri Semarang*, vol. 10, pp. 1127-1137, 2020.
- [33] C. T. Pratata, "Pengujian White Box pada Aplikasi Cash Flow Berbasis Android Menggunakan," *Jurnal Informatika Universitas Pamulang*, vol. 5, pp. 111-119, 2020.

# LAMPIRAN-LAMPIRAN

## Lampiran 1 Formulir Kegiatan Ratu



POLITEKNIK POS INDONESIA  
PROGRAM STUDI D4 TEKNIK INFORMATIKA  
Jl. SARIASIH NO. 54 BANDUNG 40131  
Telp. 022-2008562, 2008570  
Fax. 022-2009168

### FORMULIR KEGIATAN PROYEK II TA. 2022/2023

Nama : Ismaenti Nur Latifah  
Npm : 1214033  
Judul : Transaksi Top Up Diamond Pada Game Online Mobile Legends  
Pembimbing : Rani Andaryyah, ST., M.KOM., S.FPC



Pertemuan	Tanggal	Sudah Dikerjakan	Pekerjaan Selanjutnya	Nilai
1	-	-	-	-
2	11-11-2022	laporan bab I	menyusun laporan bab I	100
3	17-11-2022	laporan bab I	menyusun laporan bab I dan bab II	100
4	-	-	-	-
5	28-11-2022	laporan bab II	menyusun laporan bab II	100
6	09-12-2022	laporan bab II	menyusun laporan bab II bagian aplikasi	100
7	13-12-2022	laporan revisi bab II dan bab II	melanjutkan membuat penjelasan dari alir flowmap	100
8	23-12-2022	laporan revisi bab II membuat penjelasan dari alir flowmap	melanjutkan pengujian source code code coverage	100
9	27-12-2022	laporan revisi bab II membuat skur diagram uml	melanjutkan pengujian source code code coverage dan membuat sequence diagram	100
10	10-01-2023	laporan revisi bab II membuat sequence diagram	melanjutkan pengujian source code code coverage dan revisi flowmap	100
			Rata-Rata :	100.00

Bandung, 17 Januari 2023

Pembimbing,



Rani Andaryyah, ST., M.KOM., S.FPC  
NIDN. 0420058801



## Lampiran 2 Formulir Kegiatan Isnaenti



POLITEKNIK PIS INDONESIA  
PROGRAM STUDI DI TEKNIK INFORMATIKA  
Jl. SARIASIH NO. 54 BANDUNG 40131  
Telp. 022-269562, 269579  
Fax. 022-269568

### FORMULIR KEGIATAN PROYEK II T.A. 2022/2023

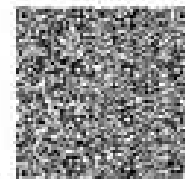
Nama : Isnaenti Nur Latifah  
Npm : 1214033  
Judul : Transaksi Top Up Diamond Pada Game Online Mobile Legends  
Pembimbing : Rini Andaryah, ST., M.KOM.,SFPK



Paralel	Tanggal	Sudah Dikerjakan	Pekerjaan Selanjutnya	Nilai
1	-	-	-	-
2	11-11-2022	laporan bab I	menyusun laporan bab II	100
3	17-11-2022	laporan bab I	menyusun laporan bab II dan bab III	100
4	-	-	-	-
5	26-11-2022	laporan bab II	menyusun laporan bab III	100
6	09-12-2022	laporan bab II	menyusun laporan bab III bagian aplikasi	100
7	13-12-2022	laporan revisi bab I dan bab II	melanjutkan membuat penjelasan dari alur flowmap	100
8	22-12-2022	laporan revisi bab II membuat penjelasan dari alur flowmap	melanjutkan pengujian source code code coverage	100
9	27-12-2022	laporan revisi bab II membuat alur diagram unit	melanjutkan pengujian source code code coverage dan membuat sequence diagram	100
10	10-01-2023	laporan revisi bab I membuat sequence diagram	melanjutkan pengujian source code code coverage dan revisi flowmap	100
			Rata-Rata:	100.00

Bandung, 17 Januari 2023

Pembimbing,



Rini Andaryah, ST., M.KOM.,SFPK  
NIDN. 0420058801

### Lampiran 3 BUKTI CODE COVERAGE

#### Bukti *Code coverage*

```
PS D:\xampp1\vendor\bin\phpunit codecoverage\SimpleTest.phpphpunit code SimpPHPUnit 9.5.28 by Sebastian Bergmann and contributors.
```

```
-
```

```
1 / 1 (100%)
```

```
Time: 00:00.022, Memory: 4.00 MB
```

```
OK (1 test, 1 assertion)
```