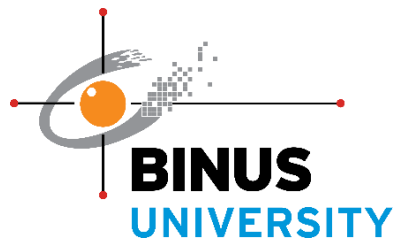


Laporan Menjadi Full Stack Developer dan IT Support Software Engineering
RS St. Carolus Jakarta

Laporan *Enrichment Program*

oleh

Pradah Widjaksono Suherli 2602058573



School Of Computer Science
Teknik Informatika
School Of Computer Science

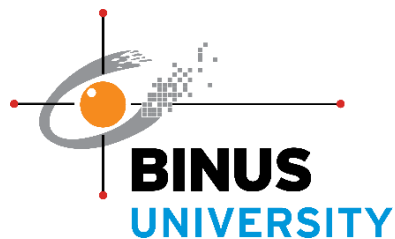
UNIVERSITAS BINA NUSANTARA
JAKARTA
2025

Laporan Menjadi Full Stack Developer dan IT Support Software Engineering
RS St. Carolus

Laporan *Enrichment Program*

oleh

Pradah Widjaksono Suherli 2602058573



School Of Computer Science
Teknik Informatika
School Of Computer Science

UNIVERSITAS BINA NUSANTARA
JAKARTA
2025

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan rahmat-Nya sehingga penulis dapat menyelesaikan laporan magang ini dengan baik. Laporan ini disusun sesuai dengan pemenuhan Enrichment Program yang merupakan salah satu persyaratan akademik pada Program Studi Computer Science, Universitas Bina Nusantara (BINUS).

Selama masa magang di Sistem Informasi Rumah Sakit (SIRS) Santo Carolus, penulis memperoleh banyak pengalaman dan pembelajaran yang sangat berharga terkait proses pengujian sistem dan pengendalian mutu dalam dunia kerja profesional.

Pada kesempatan ini, penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

- Ibu Hanis Amalia Saputri, S.Kom., M.Kom., selaku dosen pembimbing, atas waktu, bimbingan dan evaluasi yang diberikan selama penyusunan laporan ini.
- Bapak Martinus Harry Purwanto, selaku pembimbing lapangan di Rumah Sakit St. Carolus, yang telah membimbing, memberikan arahan, dan kesempatan untuk terlibat langsung dalam berbagai kegiatan di Unit SIRS.
- Seluruh staff dan karyawan di Rumah Sakit St. Carolus yang telah mendukung dan membantu selama proses magang berlangsung.
- Serta semua pihak yang turut membantu secara langsung maupun tidak langsung dalam penyusunan laporan ini.

Penulis menyadari bahwa laporan ini masih memiliki kekurangan. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun demi perbaikan di masa mendatang.

Akhir kata, semoga laporan ini dapat memberikan manfaat bagi pembaca serta menjadi gambaran umum mengenai pengalaman magang di Rumah Sakit St. Carolus.

Jakarta, 24 Agustus 2025



Pradah Widjaksono Suherli

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
BAB 1 PENDAHULUAN.....	1
1.1. Profil Perusahaan/Institusi/Komunitas	1
1.2. Posisi dan Peran Mahasiswa	1
BAB 2 LAPORAN KEGIATAN	6
2.1. Proses Bisnis.....	6
2.2. Kegiatan yang Dilakukan Sesuai <i>Learning Plan</i>	9
2.3. Penuntasan Tugas dan Penanganan Masalah.....	28
BAB 3 KESIMPULAN	33
REFERENSI.....	34

BAB 1

PENDAHULUAN

1.1. Profil Perusahaan/Institusi/Komunitas

Rumah Sakit St. Carolus (RSSC) adalah rumah sakit Katolik pertama di Indonesia, didirikan Keuskupan Agung Jakarta pada 13 Januari 1919 di Salemba, Jakarta. RSSC merupakan rumah sakit tertua ke-2 di Jakarta dan dikelola oleh Yayasan Sint Carolus.

Sejarahnya berawal tahun 1910 ketika sejumlah tokoh Katolik Batavia bersama Mgr. Edmundus Sybrandus Luypens SJ berinisiatif mendirikan rumah sakit. Tahun 1918, sepuluh Suster Carolus Borromeus dari Belanda tiba di Batavia untuk memulai karya misi. Pada, 21 Januari 1919, RSSC resmi dibuka dengan 40 tempat tidur.

Sepanjang sejarahnya, RSSC melewati masa penjajahan Belanda, Jepang, hingga kemerdekaan Indonesia. Beberapa peristiwa penting terjadi di antaranya:

- Rumah Sakit Santo Carolus menjadi tempat perawatan Presiden Soeharto (1980)
- Meraih predikat RS Sayang Ibu dan Bayi (2007)
- Peresmian Gedung Medik St. Carolus Borromeus (GMCB) (2016-2018)
- Lulus akreditasi Paripurna (2022)

Visi dari RSSC Adalah menjadi mitra Kesehatan keluarga terpercaya yang memberikan layanan Kesehatan bertaraf internasional yang didukung teknologi tepat guna.

Misi dari RSSC Adalah Pertama, Memberikan layanan Kesehatan yang bermutu dan aman dengan sikap belarasa, hormat terhadap kehidupan tanpa membedakan agama, ras, golongan, dan status social dalam Upaya pemulihan Kesehatan yang berfokus pada pasien, dan bersinergi dalam pelaksanaan pelayanan, Pendidikan, penelitian, dan pengembangan sesuai ilmu pengetahuan dan teknologi Kesehatan terkini yang berlandaskan pada penerapan budaya ICARE yang menjadi nilai keutamaan RSSC.



Gambar 1.1 Nilai-nilai keutamaan ICARE Rumah Sakit Santo Carolus Borromeus

Rumah Sakit Santo Carolus Borromeus yang diubah Namanya menjadi Pelayanan Kesehatan St. Carolus (PKSC) memiliki beberapa fasilitas untuk mencukupi pasien maupun staf, maupun murid-murid yang mendalami ilmu Kesehatan di dalamnya seperti.



Gambar 1.2 Gedung GRCB Rumah Sakit Carolus Borromeus

Merupakan Gedung terbaru PKSC, terdiri dari 8 lantai dan 3 lantai basement sebagai peluang membuka meningkatkan cakupan dan kualitas layanan sehingga dapat mencakup seluruh lapisan Masyarakat. Menjadi tempat untuk IGD, dan pendaftaran pasien. 4 lantai untuk Unit Rawat Jalan dan Medical Check Up. 2 lantai untuk Rawat Inap, dan 3 lantai basement untuk parkir kendaraan dengan kapasitas 300 mobil.



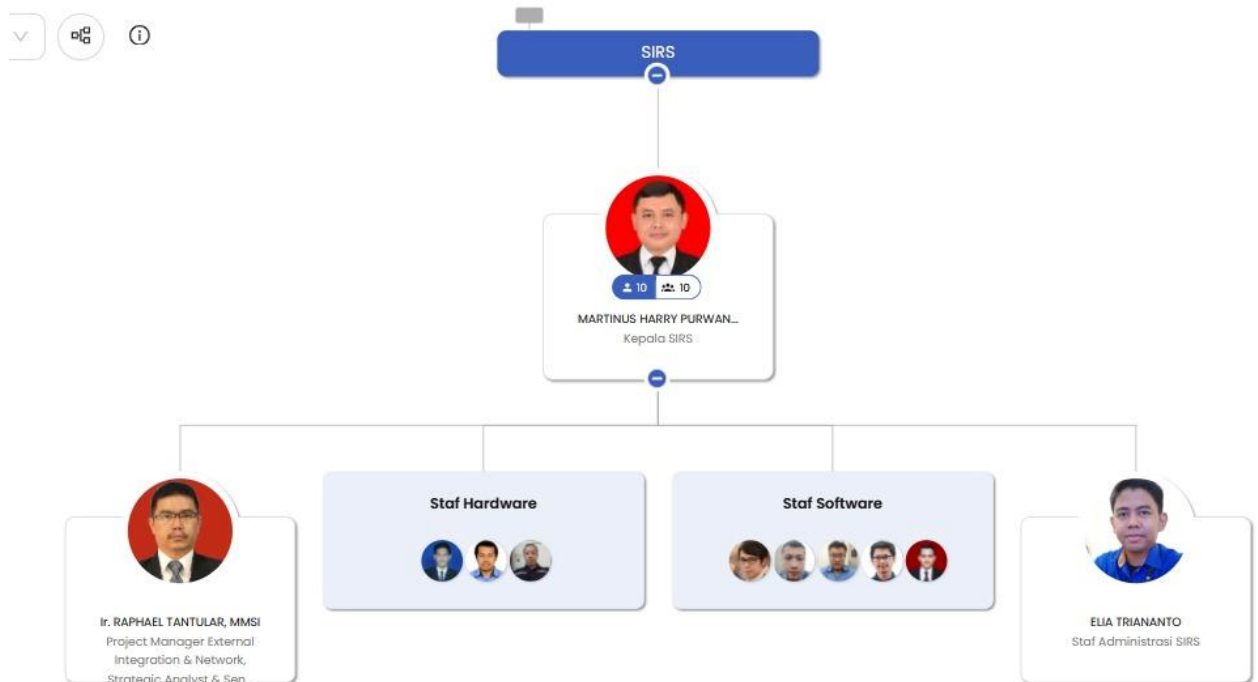
Gambar 1.3 Kantor SIRS

Merupakan kantor tempat kerja SIRS, memiliki ruang rapat yang biasanya dipakai untuk rapat Bersama pihak-pihak lain di PKSC yang di dalamnya ada projector dan TV besar, dan memiliki TV besar di Tengah ruangan yang terhubung ke computer untuk rapat di pagi hari. Ada juga tempat makan di belakang yang terhubung dengan ruangan kerja hardware support SIRS.

1.2. Posisi dan Peran Mahasiswa

Saya berada di SIRS (Sistem Informasi Rumah Sakit) subdivisi Software di PKSC. Tugas kami Adalah tak hanya melayani segala kendala-kendala teknis (IT Support) di Rumah Sakit, tetapi juga sebaga Full Stack Developer di Rumah Sakit. Kami jadi melayani Rumah Sakit secara IT Support dan develop aplikasi/program yang dapat meningkatkan mutu pelayanan PKSC.

1.3 Kepengurusan Kantor SIRS:



Gambar 1.4 Bagan Kepengurusan SIRS

Keterangan:

- Kepala SIRS = Martinus Harry Purwanto
- Project Manajer Eksternal & Integrasi & Network Strategic Analyst = Ir. Raphael Tantular, MM
- Staf Administrasi SIRS = Elia Triananto

Staf Software :

- Aditya Rizki Andriatna, S.Kom
- Wibowo, S.Kom
- Muhamad Kamaludin, S.kom
- Ryan Herdianto, S.Kom
- Albertus Karvel Bith, AMD.Kom

Staf Hardware :

- Andika Prasetyo
- Irfan Augusta
- Ignatius Rosario

Intern Software :

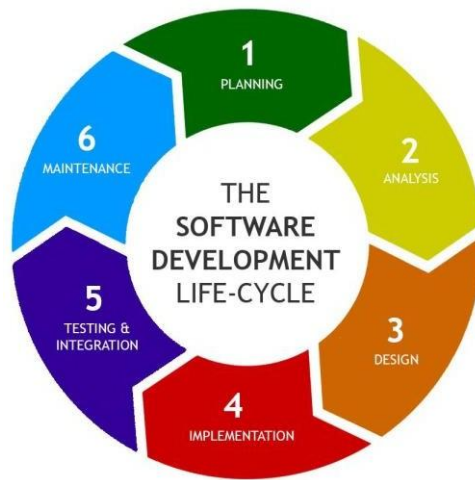
- Pradah Widjaksono Suherli
- Kennard Norbert Sudiarjo

BAB 2

LAPORAN KEGIATAN

2.1. Proses Bisnis

Proses bisnis SIRS dalam develop aplikasi mengikuti Software Development Lifecycle.



Gambar 2.1 Software Development Lifecycle



Gambar 2.2 Flowchart proses bisnis SIRS

Ini dipakai untuk semua proyek yang kami develop di SIRS, jadi kami mulai dengan wawancara dengan user untuk membuat proyek baru, kami tulis requirementsnya di dalam word sambil direkam rapatnya, kemudian setelah kami merapikan dokumennya kami Analisa requirementsnya dan mulai design software kami melalui UML. ERD untuk database, flowchart untuk flow, dan Figma untuk UI/UX. Kemudian kami mulai coding, nanti setelah proyek berada

di fase yang cukup stabil dan mengikuti semua requirements akan ada rapat dengan user untuk menerima saran dan masukan sekalian testing, Selanjutnya akan direvisi dan jika user sudah menerima programnya, kami akan deploy dan menunggu untuk maintenance.

2.2. Kegiatan yang Dilakukan Sesuai *Learning Plan*

2.2.1 Fase Sebelum Development (Planning)

Kegiatan yang saya lakukan saat 6 bulan di RSSC Adalah, mengerjakan proyek-proyek development software untuk membuat pelayan RSSC lebih efisien dengan bantuan teknologi berbasis software atau aplikasi. Mengikuti alur SDLC, dimulai dari interview user, dan mencatat requirements, dilanjutkan dengan visualisasi project dengan diagram. Menggunakan flowchart untuk mapping flow proses aplikasi, ERD untuk mapping pembuatan database dan backend dari aplikasi, terus memilih Bahasa coding apa yang akan dipakai untuk membuat aplikasi biasanya dibagi 2 Front End dan Backend.

2.2.2 Development

Kemudian kami mulai coding proyek kami secara full stack setelah merancang dan memvisualisasikan proyek yang kami buat, biasanya kami coding melalui Visual Studio Code, environment backend kami memakai MySQL dari Laragon dan visualisasi databasenya dari TablePlus. Dari backend kami memakai Model View Controller design pattern, kemudian membuat query-query untuk aplikasi di model, yang di validasi oleh controller dan dijadikan route untuk dipakai di bagian frontend.

Kami juga diberi akses ke servis-servis eRSSC seperti API, dan servis untuk mengirim pesan WhatsApp, validasi kode booking atau NOP, dll.

Setelah melakukan coding dilanjutkan dengan Functional Testing untuk mencari bugs di aplikasi secara local dengan membuat google sheets dengan semua fitur-fitur cara kerjanya dan hasil yang diinginkan. Kemudian kami akan *deploy* aplikasi ke server melalui *docker linux* menggunakan *CLI*. Terakhir kami melakukan Live Testing untuk aplikasi-aplikasi yang memerlukan pelayanan pasien seperti antrian untuk mencari bug dan saran sebagai revisian.

2.2.1. *Project*

a) **Bulan I**

Untuk bulan pertama kami membantu karyawan magang dengan projeknya untuk bisa tanda tangan pdf Standard Operating Procedure (SPO) secara digital dan di save, melalui pdfJS untuk library pdfnya dan yii2 untuk aplikasinya, kami riset di pdfjs bagaimana untuk menggambar dengan pen, terus modifikasi fitur savenya supaya dapat menyimpan tanda tangannya ke database dan download untuk lockl, untuk nanti ditampilkan. Kemudian disambungkan pdfJS yang sudah dimodifikasi ke aplikasi yii2nya supaya bisa CRUD pdfnya.

Kemudian kami belajar cara membuat RBAC website, saya membuat backendnya via Laravel Scramble, disana kami mempelajari cara memakai JWT untuk autentikasi dan menjaga keamanan aplikasi dan prinsip prinsip RBAC.

b) **Bulan II**

Kami melanjutkan website RBAC kami, dan mendapati projek baru yaitu sebuah form Medical Check Up untuk IGD rumah sakit, disana saya membuat figma dari formulir medical checkup, terus flowchart prosesnya dan mapping databasenya melalui ERD. Formnya bersifat website tetapi bisa dipakai untuk mobile dan tablet, setelah perancangan aplikasinya kami memakai NextJs dan bootstrap sebagai Frontend, dan Laravel PHP untuk backend aplikasi.

Form MCUnya dibagi-bagi menjadi halaman-halaman berbeda berdasarkan berapa banyak halaman di form tertulisnya yang dibagi berdasarkan hal-hal seperti identitas diri, Riwayat Kesehatan pribadi, keluarga dan keluarganya pasien tersendiri, disitu secara frontend pagesnya terdiri dari komponen-komponen terus karena beberapa pertanyaan cara jawabnya sama misalnya iya or tidak, jadi diutuskan membuat sebuah form yang bisa diisi dengan pertanyaan sehingga tinggal mapping satu komponen untuk banyak pertanyaan mengurangi jumlah komponen di aplikasi.

Kemudian memakai google OAuth untuk mencoba login ke form MCU dengan gmail sekaligus menyimpan email dari pasiennya. Selanjutnya, membuat validasi data yang diinput di modul pertanyaanya dan kemudian membuat sebuah page setelah login untuk cek apakah user pernah mengisi formnya melalui pengecekan NIK nama dan tanggal lahir. Bila user sudah pernah mengisi form tersebut nanti semua jawaban dari database untuk user tersebut akan terisi di formnya tinggal update dan Ganti jawaban yang perlu digantikan.

Kemudian, membuat api untuk save jawaban form tersebut. Struktur databasenya dibagikan menjadi berbagai table berdasarkan halaman yang disediakan jadi ada table untuk response form identitas, Riwayat Kesehatan pribadi, dll. Kemudian kami lengkapi dengan API dan bridging ke frontend.

Terakhirnya, kami memulai setup untuk melanjutkan projek-projek dari karyawan magang sebelumnya seperti Tensi Mandiri dan Antrian farmasi.

c) **Bulan III**

Melanjutkan setup antrian farmasi dan tensi mandiri, Carlobot Chatbot WA, dan juga MCU.

Untuk MCU diperlukan fitur save jawaban sebagai pdf jadi memakai library jsPDF untuk membuat pdf sendiri secara custom yang dapat di download secara lokal setelah menjawab semua formulir MCU, kemudian kami melakukan bug fixing dan mempercantik dan optimisasi frontend backend sebisa mungkin untuk disimulasikan di rapat Bersama dokter-dokter.

Kemudian, untuk Antrian Farmasi ada berbagai hal-hal penting yang terjadi selain setup farmasinya supaya bisa run di lokal dari karyawan magang sebelumnya, dilanjutkan dengan bug fixing dan setup websocket supaya tidak perlu timer tiap kali untuk call API mengambil data ke frontend tetapi melalui emitter dari websocket, mengutamakan frontend dinamis dan efisien secara realtime. Kemudian membuat beberapa modifikasi *Quality Of Life* seperti, mengambil data berdasarkan timestamp dia terbuat di database atau diupdate statusnya, membuat queue untuk mengucapkan pasien yang dapat ambil obat, pembuatan antrian baru melalui kode booking baru FAXXXX.

Kemudian, untuk tensi mandiri saya mempelajari RUST karena backendnya memakai rust untuk bisa terhubung ke alat tensi mandirinya, setelah setup dan bugfixing kami langsung test di lantai 2 Klinik Jantung dan Klinik Penyakit Dalam untuk menerima saran dan masukan.

Terakhirnya, untuk Carlobot, saya mempelajari cara memakai *docker* CLI, cara *deploy* dan build aplikasi di docker serta mempelajari RUST untuk backend carlobot.

d) **Bulan IV**

Bulan ini saya lebih fokus ke proyek Antrian Farmasi dan sedikit Carlobot.

Untuk farmasi kami melakukan testing secara langsung di Farmasi BPJS, membuat halaman baru disebut Logs untuk melihat data heuristik Antrian Farmasi dan yang paling penting. Membuat halaman kumpulan semua logging data yang masuk ke antrian farmasi dari verifikasi ke pembuatan ke pemanggilan obat ke pasien, kemudian di verifikasi mulai mengganti sistemnya dari servis menjadi proses sehingga tidak memerlukan pasien datang ke loket untuk verifikasi tetapi tinggal nunggu diverifikasi.

Caranya Adalah mengubah sistem Antrian Farmasi yang tadi berdasarkan kiosk BPJS menjadi seperti sebuah servis ticketing, dimana setiap antrian BPJS yang masuk akan di print keluar tracer, kemudian dapat di scan melalui barcode scanner setelah verifikasi untuk nanti disiapkan obatnya sesuai resep. Disini mulai munculnya sistem notifikasi WA farmasi yang memberi notifikasi ke pasien tentang status obat dan juga nomor antriannya yang sekarang diganti berdasarkan tipe obat yakni RC untuk Racikan dan NR non racikan.

Kemudian kami melakukan bugfixing, optimisasi backend dan frontend communication aplikasi, mengubah display admin dan display untuk pasien sesuai dengan revisi farmasi.

Terakhirnya, untuk carlobot kami hanya menambahkan hal-hal seperti pertanyaan baru atau teks deskripsi baru dan up ke docker.

e) **Bulan V**

Bulan ini masih melanjutkan farmasi berdasarkan revisi yang diberikan, dan fixing Kiosk BPJS.

Untuk farmasi kami mengubah booking id menjadi NOP baik di database, backend dan frontend jadi primary key barunya Adalah NOP bukan booking id. Menambahkan berbagai visualisasi data untuk table logs, kemudian menambahkan fitur-fitur Quality Of Life di admin verif untuk mempermudah mencari antrian atau data yang diperlukan seperti search bar, filter berdasarkan tanggal untuk page verif, proses obat dan juga admin obat untuk pemanggilan pasien mengambil obatnya.

Kemudian lanjut menggantikan display farmasi supaya lebih simple dan mudah dibaca, dan juga menambahkan data-data penting seperti status obat beserta nama pemilik antrian, memberi warna berdasarkan statusnya, misalnya hijau menunggu/dipanggil, dan merah untuk dilewatkan. Dan menambahkan fitur di admin verif untuk insert antrian baru berdasarkan NOP pasien, yang nantinya dapat pilih tipe obat dan send WA ke pasien untuk notifikasi beserta print di tracer BPJS.

Dilanjutkan dengan simulasi antrian farmasi lagi untuk mencari revisi dan masukan, setelah simulasi melakukan revisi yakni membuat website lebih realtime dan sinkron datanya saat pindah pindah halaman dan status melalui SocketJS, mempermudah UI baik untuk admin dan pasien, menambah fitur edit nomor telepon di admin verif sekaligus send WA ke nomor yang baru diupdate, retry send WA dan print jika terjadi error saat eksekusi functionnya, dan membuat dokumentasi progress farmasinya dengan screenshot UI dan flow successnya di word.

Kemudian untuk Kiosk BPJS kami mencari tau kenapa terjadi error crash atau force close setiap kali mau fingerprint scan.

Terakhir, dari tanggal 30 Juni untuk 1 minggu saya tidak bisa masuk kerja karena sakit penyakit dalam di bagian lambung dan pencernaan.

f) **Bulan VI**

Kembali bekerja pada minggu ke 2 Juli di tanggal 6 Juni. Menyelesaikan farmasi BPJS untuk simulasi terakhir.

Dimulai dengan menyamakan semua komponen dari Antdesign karena berbagai komponen seperti datepicker diambil dari library lain atau bikin sendiri, dan mempelajari slug di nextjs untuk mempersiapkan pemakaian antrian farmasi di tempat lain seperti LT3 dan LT1 GMCB RSSC, sehingga diperlukan sebuah page setelah login untuk pilih mau ke farmasi bagian mana, dan menambahkan back button di sidebar untuk balik ke page Lokasi farmasi tersebut.

Kemudian melengkapi dokumentasi farmasi dengan update ERD, flowchart dan dokumentasi API beserta functional testing. Lanjut bugfixing, optimisasi API call farmasi dan SocketJS, membuat fungsi backend untuk ambil semua data atau data-data tertentu per hari atau berdasarkan tanggal tertentu untuk ditampilkan di frontend

Kemudian menambahkan fitur untuk mengirim pesan ke WA pasien atau mengirim ke WA sendiri di sidebar dalam bentuk toggle switch untuk safe testing, kemudian kami mempelajari docker dan cara deploy aplikasi ke linux docker melalui CLI, setelah itu kami melakukan Simulasi Antrian Farmasi dan sampai sekarang berjalan lancar sambil revisi revisi hal-hal minor.

Terakhirnya, Kami juga mempelajari senam untuk pasien pasien di healing garden dan update deskripsi dan gambar di carlobot.

2.2.2. *Technical Competency (TC)*

a) **Bulan I**

Untuk proyek penandatanganan SPO secara digital hubungannya terhadap kompetensi teknis saya tidak hanya dalam pengertian saya dalam Bahasa coding Javascript atau Laravel tetapi juga dalam analisa kode dan refactoring supaya sesuai dengan yang diperlukan oleh proyek kami. Umumnya kami memakai dokumentasi dan bantuan AI untuk mengertikan code dan mengubah cara kerjanya.

Untuk RBAC hubungannya terhadap skill kami dalam coding Full Stack dengan prinsip-prinsip coding yang telah dipelajari. Untuk backend kami memakai Laravel dengan memakai Scramble dan kemudian untuk proyek-proyek lainnya Postman untuk dokumentasi API yang merupakan hal yang penting, sebuah jurnal dari 2019 menjelaskan dampak-dampak positif API Documentation yakni, untuk mempraktikan langsung kerja dan hasil dari API yang terkumpul di dokumentasi, dan mempermudah developer untuk membaca, mengumpul, mengertikan, dan testing API yang dipakai di sebuah proyek [1]. Dengan kompetensi teknis API Documentation kami bisa mengumpul dan juga mengertikan segala koneksi API di proyek kami, dan juga mempermudah mempelajari segala kode error saat testing API seperti 404, 500, 400, dll.

Kemudian saya mendalami konsep Model View Controller design pattern yang menjadi dapat diambil dari jurnal ini dari tahun 2022, karena sangat mudah untuk dimengertikan, memecahkan aplikasi menjadi Model, View, dan Controller, setiap bagian di MVC mempunyai tujuannya sendiri dan tidak saling campur tetapi terhubung

untuk melakukan tujuan tertentu. Sehingga aplikasi mudah untuk di modifikasi, dan menjaga keamanan dari backend aplikasi dengan abstraction. [2]

Dari segi frontend hubungannya terhadap kompetensi teknis yang saya miliki lebih cenderung ke menghubungkan Frontend dengan Backend melalui API route, karena saya di projek ini lebih cenderung menjadi backend developer.

Saya memakai Axios untuk membantu hal tersebut, jadinya ada sebuah folder yang mengumpulkan semua API routes dari backend ke frontend, nanti jikalau sebuah web atau komponen memerlukannya tinggal panggil function yang diperlukan untuk trigger Axios call. Alasan kami memakai Axios karena dia mempermudah komunikasi dengan REST API yang telah kami buat dan dokumentasi, dan bisa dipakai di backend atau frontend, axios juga bisa membuat operasi CRUD, menerima dan merubah request maupun response ke dalam bentuk JSON untuk mempermudah pembacaan data, disini kami mempelajari cara-cara lebih efisien untuk melakukan API call di frontend. [3]

Projek RBAC juga memiliki hubungan erat dengan Software Design melalui UML. Merupakan salah satu tahap dari SDLC yakni design kami memvisualisasikan aplikasi kami melalui diagram seperti UML. Tahap ini sangat penting karena dapat memvisualisasikan requirement dari aplikasi kami untuk mempermudah development, kami biasanya mapping flow dari aplikasinya lewat flowchart, UI dari Figma dan database backend melalui ERD, disini kami diajari oleh karyawan magang system analyst cara mudah untuk proses requirements jadi UML, dan cara efektif mapping database ke dalam ERD, dll [4].

b) **Bulan II**

Untuk bulan kedua, kompetensi teknis yang terhubung dengan proyek baru saya di MCU ada pembuatan UI/UX, dan visualisasi software memakai UML untuk bagian perancangan.

Untuk pembuatan UI/UX Formulir MCU, saya mengutamakan membuat sebuah form yang mudah dibaca dan dimengertikan pasien, dan yang paling penting Adalah menjaga supaya pasien tidak melewati pertanyaan-pertanyaan penting dan dapat mencari tau di pertanyaan mana yang perlu diganti, alias formnya harus responsif dan mudah dibaca, mengikuti prinsip-prinsip utama dari UI/UX design, peningkatan yang saya alami Adalah lebih ke pergantian prinsip dari membuat website yang bagus dan eye catching ke website yang simple dan mudah dibaca, karena user-user kami tidak terlalu paham teknologi, dan tidak terlalu memikirkan design tetapi yang penting Adalah mudah untuk dimengertikan. Jadinya kami mengadaptasi cara kami membuat aplikasi mengutamakan UX dibanding UI yang unik dan cakep. [5].

Dari segi development salah satu kompetensi teknis yang terlatih adalah Component Driven Design. Alasannya saya memakai Component Driven Design untuk MCU karena formnya memiliki banyak pertanyaan dan jikalau tidak dibagi dikomponen maka kodenya akan menjadi “bloated” jadinya kami memakai komponen yang bersifat “reusable” untuk menjadi modul pertanyaanya formnya, memudahkan efisiensi dan implementasi karena fungsinya tertera di komponennya sendiri. Peningkatan yang saya lihat adalah lebih pintarnya saya memakai komponen secara efektif dan jumlah dikit untuk membuat aplikasinya [6].

c) **Bulan III**

Untuk bulan ketiga, saya sudah mulai develop proyek-proyek banyak seperti MCU, Antrian Farmasi, Tensi Mandiri dan Carlobot, yang setiap proyek memiliki hubungan dengan kompetensi teknis tertentu.

Untuk Antrian Farmasi seperti biasa mengasah ke kompetensi teknis saya dalam Component Driven Development, MVC design pattern dan juga satu hal penting yaitu cara membuat website kami dinamis atau dapat mengupdate data secara realtime.

Kenapa kami memilih Antrian Farmasi kami bersifat realtime dengan Websocket? Supaya petugas dan Masyarakat dapat memantau dan mengelola antrian secara langsung tanpa latensi atau jeda waktu lama antara informasi baru dengan informasi lama. Sehingga, system ini memungkinkan pengguna mendapatkan antrian dan melihat perkembangan secara langsung, dan mempermudah petugas dalam memanggil, dan memproses obat meningkatkan efisiensi layanan, peningkatan saya di kompetensi ini menurut saya sangat drastic, tak hanya saya mempelajari prinsip baru yaitu memakai Websocket untuk update data secara realtime, tetapi saya mempelajari banyak tentang Websocket, dari cara efektif memakainya dan pula ke setupnya [7].

Cara kami melakukan hal tersebut Adalah memakai Websocket, alasannya kami memakai websocket karena “Websocket, dengan koneksi mereka yang bersifat 2 arah, selalu menyala, lebih sesuai untuk menjaga respon yang konsisten dan low latency setelah koneksi terbuka,

membuat mereka pas untuk aplikasi real time yang mengutamakan daya tanggap.” Mengutip dari jurnal membandingkan Websocket dengan HTTP polling dari Ibom State Polytechnic. Jadinya kami mempelajari cara setup Websocket di frontend dan backend, dan cara mereka melakukan realtime pollingnya, yakni dengan 2 tahap penting, emit dan receive. Jikalau dari backend sebuah socket sedang emit data atau pesan, dari frontend dapat menerima emit tersebut dengan socket.on dan mematikan dengan socket.off jadinya tidak perlu timer untuk cek dan ambil data terus dari API tetapi hanya taruh emitter socket js di tempat yang tertentu dan di receive di tempat tertentu [7].

Untuk Tensi Mandiri, ada hubungan dengan kompetensi teknis IoT Integration, ini karena saya develop software menggunakan RUST dan React untuk membaca tensi pasien saat diukur dan disimpan di dalam database.

Alasan kenapa kami memakai RUST untuk backend ini adalah karena RUST tidak memakai banyak memory CPU, ini sangat penting karena computer-computer untuk pelayanan rumah sakit tidak bersifat “high-end”. Sebuah riset yang membandingkan berbagai Bahasa coding yang dipakai untuk Backend development dari Tampere University Of Applied Sciences di Finlandia, menyatakan bahwa RUST memakai hanya 0.5% CPU dan performanya sangat efisien dan bagus meskipun pembuatnya tidak mengerti RUST dan menurut dia dapat ditingkatkan lagi dengan pengalaman lebih banyak dalam RUST, sehingga RUST disarankan jikalau perlu melestarikan memory computer, tetapi susah untuk dipelajari dan perlu pengalaman yang dalam terhadap RUST untuk mendapatkan hasil yang sangat efisien, peningkatan kompetensi teknis yang saya alami adalah di cara

menghubungkan alat tensi mandiri Omron ke software dengan cara memakai USB-B atau usb untuk menghubungkan printer ke PC dan RUST backend mengambil port device dan membuat fungsi untuk konek dan mengambil kode error dan hasil tensinya [8].

Untuk projek Carlobot, kompetensi teknis yang memiliki hubungan dengan projeknya adalah penggunaan Linux dan Docker untuk deploy aplikasi, tetapi hanya untuk update projek karena di projek ini saya hanya mengubah deskripsi, tidak deploy projek atau membuat projeknya, sehingga teori tentang kompetensi teknisnya saya jelaskan di bulan ke 6.

Untuk MCU kompetensi masih tetap sama seperti bulan sebelumnya, hanya sekarang menambahkan library pdfJS untuk dokumentasikan jawaban dari formnya menjadi sebuah PDF. Alasan kami memakai pdfJS adalah karena bersifat flexible dan modular, PDFnya bisa diganti sesuai dengan yang diinginkan dan memilki metode-metode untuk integrasi dengan aplikasi web di browser-browser seperti chrome, safari, edge, dll. pdfJS juga mudah untuk melihat dokumen pdf tanpa software external, dapat edit dan anotasi pdfnya dan bersifat efisien dalam rendering pdfnya. Kompetensi teknis baru yang saya pelajari Adalah cara customize PDF di pdfJS [9].

d) **Bulan IV**

Untuk kompetensi teknis bulan keempat saya mulai lebih fokus ke Antrian Farmasi, disini kami melakukan testing dan juga development seperti biasa. Dari segi development ada tambahan baru dalam kompetensi teknis yaitu pembuatan halaman untuk display dan pengumpulan data heuristik atau disebut Data Visualization.

Data visualization ini bersifat penting karena dapat menarik informasi dari data yang mentahnya adalah angka-angka menjadi sesuatu yang kami bisa mengerti dengan mudah, mencari anomali, melihat tren-tren seperti rata-rata jam kerja layanan, total antrian per hari dan dapat membuat solusi atau rencana untuk meningkatkan layanan secara heuristik, sehingga kami membuat dashboard yang isinya merupakan visualisasi data-data penting terhadap layanan farmasi dan data semua antrian yang masuk ke dalam antrian per hari, dan dapat melihat data dari jangka waktu tertentu., disini saya mempelajari cara efektif membuat dashboard untuk menampilkan data secara efektif dan cara membuat query database dan menghubungkan frontend dengan backend untuk mengambil data berdasarkan tanggal tertentu atau jangka waktu tertentu [10-12].

Kemudian, ada kompetensi teknis dalam integrasi print otomatis saat masuknya antrian baru yang disebut Event Driven Architecture, ini karena aplikasi kami yang terdiri dari berbagai komponent memerlukan event-event tertentu untuk beraksi seperti pergantian status ke verifikasi obat ke penyiapan obat yang nanti disertakan WhatsApp yang menunjukkan bahwa obat sedang diproses, event dimana obat ganti status menjadi terpanggil dan log waktu dipanggilnya beserta diumumkan melalui text to speech, dll. Disini event driven architecture dipakai, karena EDA merupakan prinsip dimana komponen-komponen sistem dapat mengeluarkan event dan memproses event sebagai signal perubahan di sistem, sistem ini mengutamakan dapat memproses event secara bersamaan mengurangi bottleneck dan respons realtime terhadap perubahan dan, akhirnya berkaitan dengan pentingnya real time updates melalui Websocket di bulan kemarin [13].

e) **Bulan V**

Kompetensi teknis yang terhubung dengan proyek saya melanjutkan Antrian Farmasi lebih ke development fitur Quality Of Life atau Human Computer Interaction, seperti search, filter, retry print, menambah antrian berdasarkan NOP, mengedit nomor telpon secara langsung, dll. Peningkatan yang saya rasakan Adalah cara untuk membuat fitur-fitur tersebut dengan NodeJS dan NextJS dan diintegrasikan melalui websocket bila ada perubahan terhadap data seperti menambah antrian, update nomor telpon supaya bisa update real time.

Alasan kami menambah fitur-fitur tersebut Adalah berdasarkan revisi dari testing yang kami lakukan, yakni diperlukan cara untuk memudahkan mencari pasien di admin, cara untuk melihat data di hari-hari sebelumnya, sebagai solusi terhadap masalah nomor telepon pasien yang tidak sesuai dengan database, dan mengatasi antrian yang belum masuk atau tidak memakai BPJS.

Pentingnya penambahan fitur-fitur ini bisa diambil dari prinsip-prinsipnya Human Computer Interaction dan UI/UX design. Yakni, software anda harus focus terhadap keperluan user, dan kebiasaan user, design-designnya harus konsisten dan interaksinya harus jelas, respons terhadap input user harus jelas dan langsung, fitur-fiturnya harus jelas dan membuat user lebih efisien dalam pengerjaan dan bisa dipakai siapapun, dan yang paling penting yakni untuk mengatasi error dan cari cara untuk mengembalikan user dari fase error ke normal, makannya ada print lagi, edit nomor telepon dan tambah antrian yang berguna sebagai Solusi untuk masalah-masalah dari revisi sebelumnya [14].

f) **Bulan VI**

Pada bulan keenam sudah mulai deploy Antrian Farmasi sdi Docker, disini kompetensi teknis baru yang dipelajari adalah cara deploy aplikasi dengan docker melalui linux, saya mempelajari linux, syntax Docker dan cara deploy lewat Docker. Disini saya memakai Putty menghubungkan ke server dan lewat github CI/CD clone dan pull projek ke server, dan setelah konfigurasi projek dengan Docker yaml, dockerfile, dll. Saya berhasil membuat container untuk backend dan frontendnya melalui build compose up dan aplikasinya berjalan dengan sangat cepat dibanding lewat lokal machine saya.

Alasan kami memakai Docker untuk deploy software kami adalah, karena Docker dapat menjalankan aplikasi secara konsisten di environment berbeda misalnya testing, development, production, tanpa modifikasi ke infrastruktur aplikasinya, dan karena kontainer memakai OS yang sama jadi lebih ringan dan cepat dibanding virtual machine mempercepat aplikasi dan menjaga efisiensi, dan yang paling penting adalah CI/CDnya artinya aplikasi bersifat modular dan fleksibel [15].

2.2.3. *Employability and Entrepreneurial Skill (EES)*

a) **Bulan I**

Soft skills yang dipakai untuk proyek saya pada bulan ini Adalah Critical and Creative Thinking untuk project menyimpan tanda tangan pada pdf Standard Operating Procedure. Karena harus integrasi sebuah library javascript ke dalam program yang berbahasa PHP yakni Yii yang memerlukan tingkat problem solving yang tinggi, dan pengalaman dalam berbahasa javascript dan PHP, seperti disaat saya mencoba untuk mengubah library pdfJS supaya bisa save coretan di viewernya dan mengubah fungsi saveny menjadi meynimpan ke database secara Base64, ini memerlukan keahlian refactoring dan critical thinking untuk mencari fungsi mana yang perlu di ubah dan kreatifitas untuk mencari cara mengubah fungsinya yang dari menyimpan ke lokal menjadi ke database. Sehingga terjadi peningkatan di keahlian saya dalam code refactoring yang sekaligus meningkatkan keahlian saya di soft skill critical thinking dan creative thinking.

Kemudian untuk project RBAC, soft skill yang diperlukan Adalah Collaboration, karena di projek ini tugas saya adalah Backend development, dan teman saya Kennard sebagai Frontend development sehingga perlu komunikasi yang kuat antar kedua pihak supaya proyek berjalan dengan benar, ini karena dari ramotion, kerjanya front end dan backend saling berhubungan, frontend memerlukan backend untuk data dan fungsionalitas, dan backend memerlukan requirementsnya dari frontend. Sehingga komunikasi dan kolaborasi yang efektif sangat diperlukan supaya saling mengerti requirements projectnya. Peningkatan yang saya mengalami Adalah Pelajaran terhadap pentingnya membuat dokumentasi Bersama, code review Bersama dan sharing ide Bersama untuk mencari tau manakah metode paling pas untuk projek kami.

b) **Bulan II**

Soft skill yang terpakai di bulan ini adalah Adaptability dan Digital and Technology Fluency karena di bulan ini saya memegang 2 project yang memakai bahasa-bahasa coding yang berbeda dan konteksnya berbeda.

Dari segi Adaptability, ini penting karena saya harus mengertikan cara membuat sebuah website yang bertujuan untuk melakukan role based access control, dan pembuatan formulir medical check up dimana kedua project tidak sama konteksnya, satu merupakan projek formulir simple yang bersifat web dan satunya merupakan website untuk manage user-user punya hak melalui role based access control, sehingga perlu adaptasi di cara saya pertamanya membuat projek RBAC ke membuat formulir yang simple. Peningkatan soft skillnya adalah cepatnya saya merancang dan mendevelop kedua app sekaligus, jadinya saya develop kedua projek secara paralel karena yang RBAC sudah mendekati penyelesaian, dan merancang UML beserta codingkan formulir medical check up sekaligus.

Dari segi Digital and Techonology Fluency, ini lebih ke pengalaman saya develop kedua app yang tidak seiring bahasa codingnya, mereka kedua memakai PHP laravel backend tetapi tidak untk frontend, memakai NextJS dan React untuk frontendnya, beserta itu dengan disarankan memakai beberapa library dari Manager saya dan karyawan magang sesama seperti Ant Design, JWTAuth, dll. Saya merasa mendapat peningkatan dalam pengetahuan saya dalam integrasi berbagai library dalam aplikasi saya dan prinsip-prinsip penting seperti JWTAuth, RBAC, Component Driven Design.

c) **Bulan III**

Untuk bulan ini saya mulai mengerjakan banyak projek, ini dikarenakan karyawan magang yang develop aplikasinya sudah tidak lagi bekerja di SIRS sehingga perlu ada yang menggantikannya.

Disini soft skill Initiative sangat penting di bulan ini karena harus ada yang melanjutkan proyeknya dan setup proyeknya lagi sehingga kami berinisiatif untuk setup proyeknya dan melanjutkan yang sudah ada meskipun kami juga memiliki proyek-proyek sendiri. Peningkatan skill yang saya rasa lebih ke hasil dari iniatifnya, disini saya mempelajari banyak hal seperti cara update app di Docker, lebih cepat untuk analisa coding dan mencari tujuannya, dan akhirnya soft skill saya tentang Adaptability dan Digital Fluency lebih meningkat karena mempelajari hal-hal baru seperti NodeJS backend, strukturnya bagaimana sama cara setupnya bagaimana, mempelajari library WhatsappWebJS untuk Carlobot, sekaligus melanjutkan proyek kami yang sebelumnya.

d) **Bulan IV**

Untuk bulan ini karena saya diprioritaskan untuk melanjutkan Antrian Farmasi, disini soft skill yang sangat terpakai adalah Collaboration, disini saya komunikasi ketat bersama pihak farmasi terutama Dokter Natania, dan manager saya Pak Martinus, untuk melakukan problem solving dan menggantikan requirements antrian farmasi supaya lebih sesuai dengan yang diinginkan pasien dan farmasi. Sehingga, adanya perubahan sistem antrian dan nomor antrian, dll. Disini saya merasa lebih tidak tegang untuk berkomunikasi dengan kedua pihak dan berani memberi solusi dan mempresentasikan software kami yang sedang dikerjakan.

Soft skill kedua yang penting adalah adaptability, karena sering terjadinya perombakan atau perubahan sistem-sistem penting di software kami sehingga perlu adaptabilitas yang kuat dan ide-ide yang kreatif untuk menyelesaikan perubahannya. Peningkatan soft skill yang saya alami adalah lebih cepatnya saya mencari solusi-solusi untuk integrasi sistem-sistem baru dan mudahnya saya untuk merubah sistem yang saya ada untuk mengakomodasi requirement requirement baru.

e) **Bulan V**

Soft skill yang dipakai di bulan ini masih sama seperti bulan sebelumnya dengan penambahan critical thinking and creative thinking, karena di bulan ini lebih mementingkan penambahan fitur-fitur Quality of Life untuk antrian farmasi baik dari pihak Farmasi atau Pasien. Jadi munculnya usulan dari Pak Martin untuk sebuah dashboard yang memvisualiskan data di bulan sebelumnya yang dilanjutkan, dan fitur-fitur seperti filter by calendar, search by name, layout baru untuk pasien supaya mudah untuk dibaca. Semua ini memerlukan critical dan creative thinking untuk mencari tau bagaimana menunjukkan informasi penting dalam ruang sedikit mungkin atau cara mudah untuk mencari data mempercepat layanan.

Sehingga peningkatan yang saya merasakan lebih ke area kreatifitas, bagaimana saya mempermudah pasien atau farmasi untuk mengerti UI/UX antrian farmasi, baik melalui colour coding berdasarkan status, menunjukkan nama, dll.

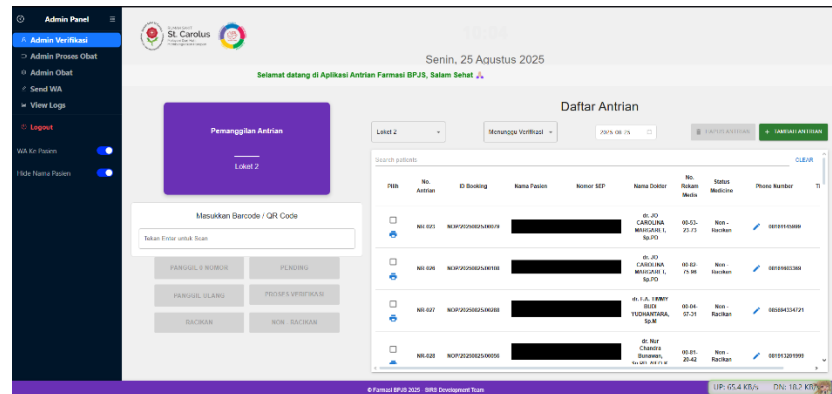
f) **Bulan VI**

Soft skills pada bulan keenam sama seperti bulan kelima dengan penambahan social awareness, social awareness disini lebih ke mengajarkan pihak-pihak farmasi cara memakai aplikasi farmasinya dan bekerja sama dengan farmasi saat live testing, dan juga membantu mengedukasi pasien-pasien BPJS cara kerja sistem Wanya. Peningkatan yang saya rasa adalah lebih lancar dan rileks saya dalam membantu dan menjelaskan naplikasinya ke orang lain.

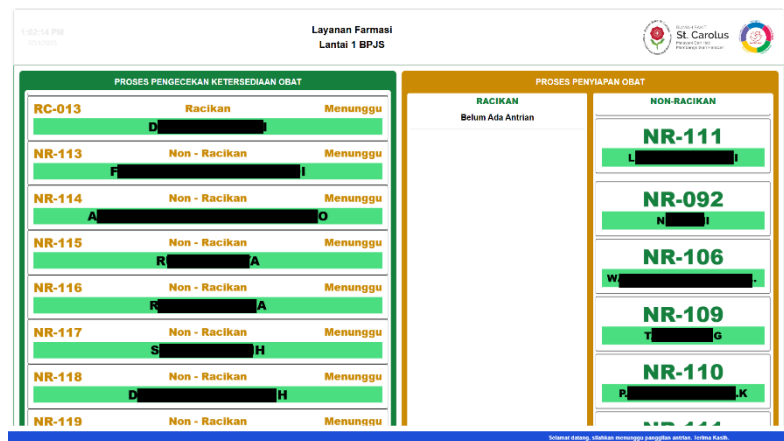
2.3. Penuntasan Tugas dan Penanganan Masalah

2.3.1. Hasil Magang

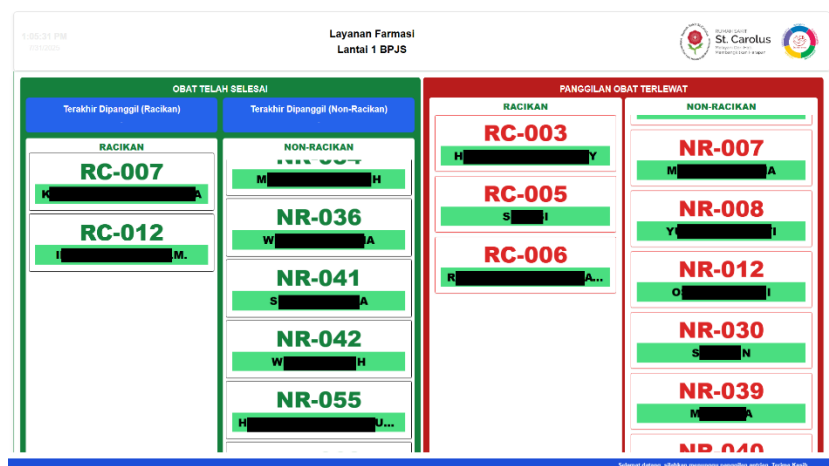
1. Antrian Farmasi BPJS



Gambar 2.1 Dashboard Admin Farmasi



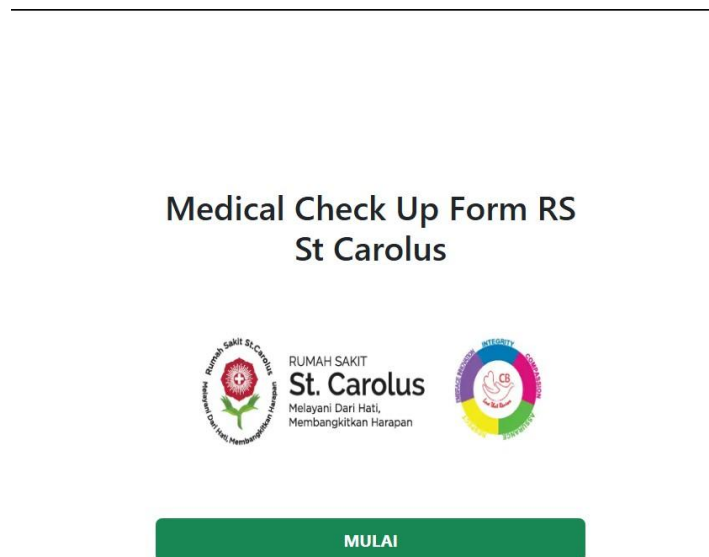
Gambar 2.2 Display Antrian Farmasi



Gambar 2.3 Display Antrian Farmasi Siap Diambil

Projek ini sudah di deploy dan sedang dipakai oleh Farmasi BPJS, masih ada beberapa bugs yang minor perlu di fix tetapi selain itu aplikasi sudah berjalan dengan lancar.

2. Medical Check-Up Form



Gambar 2.4 Home Page MCU

Medical Check Up Form RS St Carolus

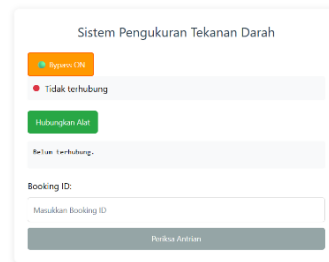
1. Identitas Pribadi

No Register	NPK
<input type="text" value="Diisi Oleh Petugas"/>	<input type="text" value="Khusus Karyawan RSSC"/>
NIK	
<input type="text" value="Diisi NIK Sesuai KTP"/>	
Nama Lengkap	
<input type="text" value="Diisi Nama Lengkap"/>	
Jenis Kelamin	
<input type="checkbox"/> Pria	<input type="checkbox"/> Wanita

Gambar 2.5 Contoh Form Identitas MCU

Projek ini belum di deploy dan dipakai oleh RSSC, tetapi form sudah bisa, menyimpan jawaban, mengambil jawaban berdasarkan user dan dapat diupdate, dan dapat di download sebagai PDF, dan sign in untuk update jawaban.

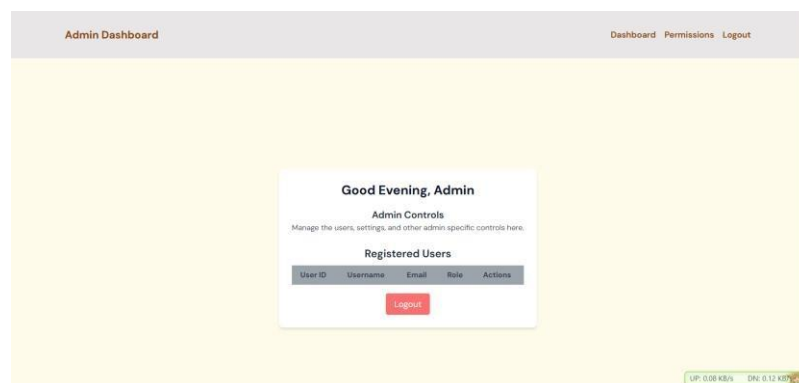
3. Tensi Mandiri



Gambar 2.6 Home Page Tensi Mandiri

Projek ini sedang dilanjutkan sejak bulan Agustus, untuk sementara sudah bisa menghubungkan aplikasi ke alat tensi mandiri, menerima hasil scan dan memberhentikan scan jikalau ada error, dan send data ke database rumah sakit berdasarkan booking id pasien, sekarang ingin mencoba bisa mengirim hal-hal lain seperti saturasi, GCS, suhu, dll.

4. Role Based Access Control Website



Gambar 2.8 Dashboard RBAC

Website ini sudah bisa memberi user role, dan untuk role tertentu dapat menambahkan permission ke role-role tertentu, user juga bisa login. Projek tidak dideploy karena merupakan bahan latihan kami selagi belum ada projek untuk kami lakukan.

5. Aplikasi Rumah Regulasi (Menyimpan SPO yang sudah ditandatangani)

September 4th 2025, 06:29:35 PM

Home / Rumah Regulasi

List Data Rumah Regulasi

Upload SPO

Active 2 Inactive 0 Deleted 0

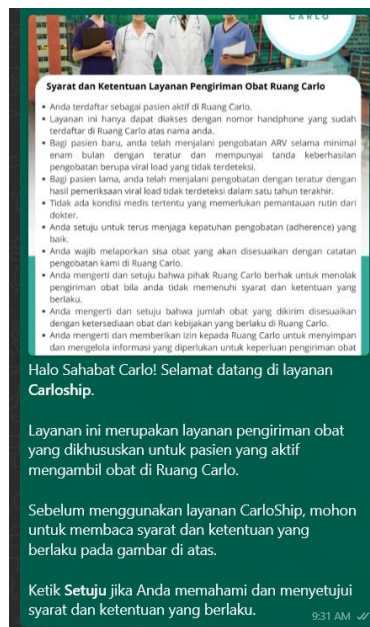
Data Rumah Regulasi Showing 1-2 of 2 items.

#	Nama	Tipe	Keterangan	Status	Tgl Dibuat	Dibuat oleh
1	Test SPO I	SPO		Active	2025-05-26 16:06:18	5408
2	Test SPO II	SPO		Active	2025-05-26 16:19:04	5408

Gambar 2.9 Halaman Rumah Regulasi

Aplikasi ini sudah selesai dibuat, dan sekarang lagi ingin diintegrasikan ke eRSSC sistem informasi Rumah Sakit Santo Carolus, aplikasi sudah bisa login, upload pdf SPO dan ditanda tangankan, terus disimpan ke dalam database. Juga bisa melihat dari siapa saja yang sudah edit atau tanda tangan pdfnya.

6. Carlobot



Gambar 2.11 Whats App Message Carlobot

Aplikasi ini sudah dideploy dan dipakai rumah sakit dari sebelum saya magang, pekerjaan saya hanya mengedit teks deskripsi, mengganti gambar dan menambah variabel alamat untuk disimpan ke dalam database.

2.3.2 Masalah dan Penganganan Masalah Magang

Masalah yang saya hadapi lebih ke mempelajari bahasa-bahasa baru seperti Docker, NodeJS, dll. Solusi saya menghadapi masalah tersebut adalah lebih banyak membaca dokumentasi dan belajar baik di kantor maupun diluar kantor. Selain itu tidak ada masalah-masalah selama magang.

KESIMPULAN

Terakhir, masa enrichment saya di SIRS selama 6 bulan ini sangat sibuk, tetapi menyenangkan. Saya mempelajari banyak hal-hal yang pasti saya akan pakai di lapangan kerja nanti, dan merupakan pengalaman yang unik. karena saya merasa magang di SIRS tidak seperti di kantor korporat, lebih beragam projek-projek kami dan oleh karena itu, mendapatkan pembelajaran dan pengalaman unik yang pasti saya tidak akan mendapati di tempat lain, seperti membuat aplikasi antrian, aplikasi penghubung tensi mandiri ke database, dll.

REFERENSI

1. Meng, M., Steinhardt, S., & Schubert, A. (2019). How developers use API documentation. *Communication Design Quarterly*, 7(2), 40–49.
<https://doi.org/10.1145/3358931.3358937>
2. Kumar, Y. (n.d.). *MVC ARCHITECTURE AND IT'S APPLICATION Computer Science and Engineering UNDER THE SUPERVISION OF.*
3. Shrestha, U. (n.d.). *IMPLEMENTING API IN REACTJS.*
4. John, B. (n.d.). *A study of uml diagrams in software development.*
<https://www.researchgate.net/publication/388683766>
5. Hamidli, N. (2023). *Introduction to UI/UX Design: Key Concepts and Principles.*
6. Crnkovic, I., Larsson, S., & Chaudron, M. (n.d.). *Component-based Development Process and Component Lifecycle.*
7. Asuquo, U. (2025). *INTERNATIONAL JOURNAL OF MODERN TECHNOLOGY AND ENGINEERING RESEARCH WEBSOCKETS VS. HTTP POLLING: IMPLICATIONS FOR HIGH-FREQUENCY DATA STREAMING APPLICATIONS.* 3. <https://doi.org/10.5281/zenodo.16893947>
8. *A Comparative Analysis of Modern Programming Languages in REST API Development* Tuukka Viitanen. (n.d.).
9. Qian, Y., & Li, Y. (n.d.). *Web-based PDF Document Display Research and Implementation.*
10. Unwin, A. (2020). Why is Data Visualization Important? What is Important in Data Visualization? *Harvard Data Science Review.*
<https://doi.org/10.1162/99608f92.8ae4d525>

11. Burnay, C., Dargam, F., & Zarate, P. (2019). Special issue: Data visualization for decision-making: an important issue. In *Operational Research* (Vol. 19, Issue 4, pp. 853–855). Springer Verlag. <https://doi.org/10.1007/s12351-019-00530-z>

12. Abudiyab, N. A., & Alanazi, A. T. (2022). Visualization Techniques in Healthcare Applications: A Narrative Review. *Cureus*.
<https://doi.org/10.7759/cureus.31355>

13. Ashwin Chavan. (2021). Exploring event-driven architecture in microservices- patterns, pitfalls and best practices. *International Journal of Science and Research Archive*, 4(1), 229–249.
<https://doi.org/10.30574/ijsra.2021.4.1.0166>

14. Calp, M. H., & Akcayol, M. A. (2015). The importance of human-computer interaction in the development process of software projects. *Global Journal of Information Technology*, 48. <https://doi.org/10.18844/gjit.v0i0.114>

15. Murali, N., & Koneru, K. (2024). Containerization Best Practices-Using Docker and Kubernetes for Enterprise Applications. In *Journal of Information Systems Engineering and Management* (Vol. 2025, Issue 45s).
<https://www.jisem-journal.com/>

