



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**PENGEMBANGAN APLIKASI *BRIDGING* BPJS &
REMUNERASI DOKTER PADA RUMAH SAKIT XYZ
MENGUNAKAN METODE SPIRAL**

TUGAS AKHIR

Farhan Ma'arif Lubis

0110220116

PROGRAM STUDI TEKNIK INFORMATIKA

DEPOK

MEI 2024



**STT TERPADU
NURUL FIKRI**

SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**PENGEMBANGAN APLIKASI *BRIDGING* BPJS &
REMUNERASI DOKTER PADA RUMAH SAKIT XYZ
MENGUNAKAN METODE SPIRAL**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar
Sarjana**

STT - NF
Farhan Ma'arif Lubis
0110220116

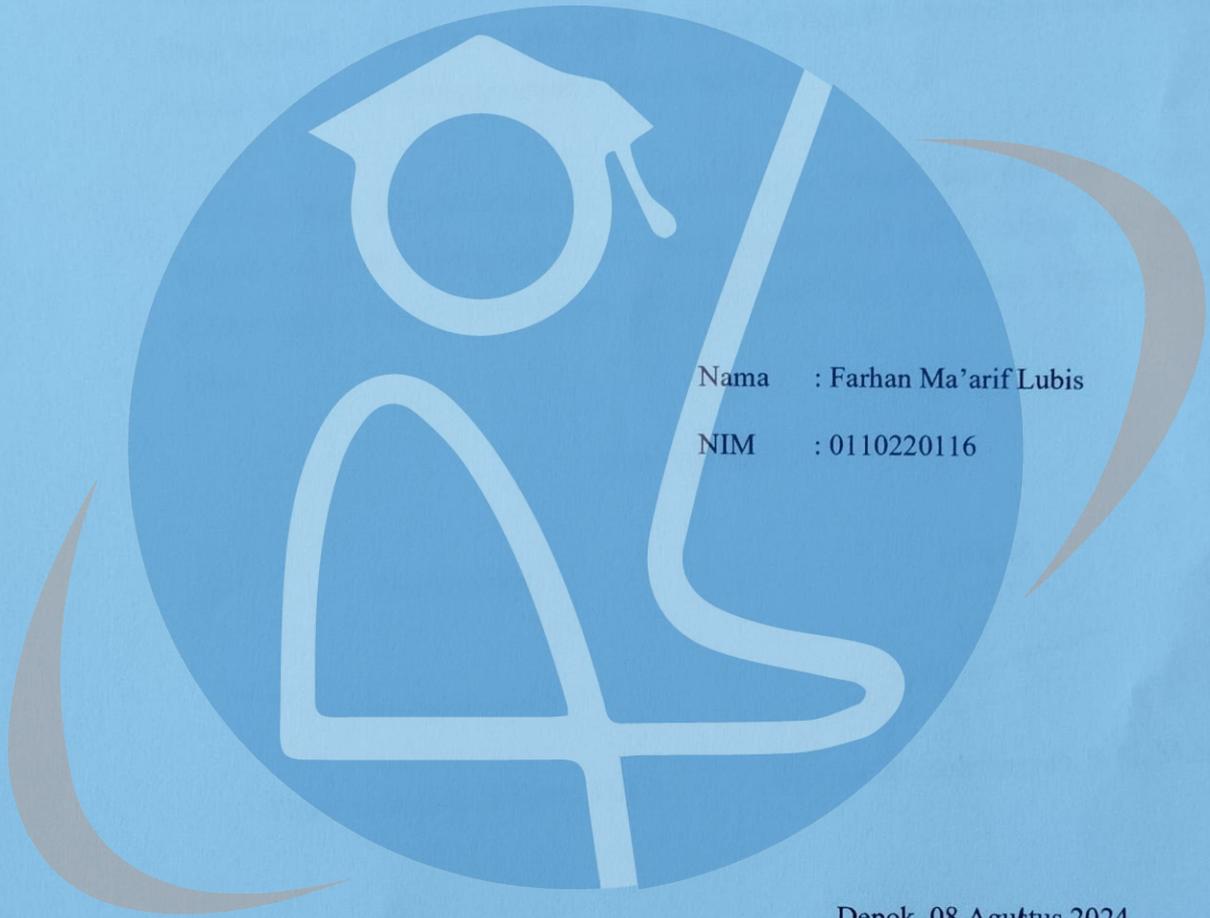
PROGRAM STUDI TEKNIK INFORMATIKA

DEPOK

MEI 2024

HALAMAN PERNYATAAN ORISINALITAS

Skripsi/Tugas Akhir ini adalah hasil karya penulis, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

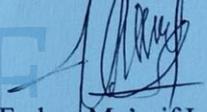


Nama : Farhan Ma'arif Lubis

NIM : 0110220116

Depok, 08 Agustus 2024

STT - NE


Farhan Ma'arif Lubis

HALAMAN PENGESAHAN

Skripsi/Tugas Akhir ini diajukan oleh :

Nama : Farhan Ma'arif Lubis

NIM : 0110220116

Program Studi : Teknik Informatika

Judul Skripsi : Pengembangan Aplikasi *Bridging Bpjs & Remunerasi Dokter* Pada Rumah Sakit Xyz Menggunakan Metode Spiral

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri.

DEWAN PENGUJI

Pembimbing

Penguji



Tifani Nabarian, S.Kom, M.T.I.



Ahmad Rio Adriansyah, S.Si. M.Si.

STT - NF

Ditetapkan di : Depok

Tanggal : 08 Agustus 2024

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan skripsi/Tugas Akhir ini. Penulisan skripsi/Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana komputer Program Studi Teknik Informatika pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi/tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Allah SWT.
2. Orang tua dan semua anggota keluarga yang telah memberikan dorongan baik secara moril maupun materil dalam penyelesaian tugas ini.
3. Bapak Dr. Lukman Rosyidi selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Ibu Tifani Nabarian, S.Kom., M.T.I. selaku Ketua Program Studi Teknik Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri dan selaku Dosen Pembimbing Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
5. Henry Saptono, S.Si, M.Kom selaku Dosen Pembimbing Akademik yang telah membimbing penulis selama berkuliah di Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
6. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing penulis dalam menuntut ilmu yang telah diberikan.
7. Rumah Sakit XYZ beserta karyawan yang telah meluangkan waktunya untuk memberikan data yang diperlukan bagi penulisan ilmiah ini.

Dalam penulisan ilmiah ini tentu saja masih banyak terdapat kekurangan-kekurangan yang mungkin disebabkan oleh keterbatasan kemampuan dan pengetahuan yang penulis miliki. Walaupun demikian, penulis telah berusaha menyelesaikan penulisan ilmiah ini sebaik mungkin. Oleh karena itu apabila terdapat kekurangan di dalam penulisan ilmiah ini, dengan rendah hati penulis menerima kritik dan saran dari pembaca.

Akhir kata, penulis berharap Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Jakarta, Mei 2024

Farhan Ma'arif Lubis



STT - NF

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda tangan di bawah ini:

Nama : Farhan Ma'arif Lubis

NIM : 0110220116

Program Studi : Teknik Informatika

Jenis karya : Tugas Akhir

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT-NF Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty – Free Right*) atas karya ilmiah saya yang berjudul:

Pengembangan Aplikasi *Bridging Bpjs & Remunerasi Dokter Pada Rumah Sakit Xyz Menggunakan Metode Spiral*

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 08 Agustus 2024

STT - NF

Yang menyatakan



Farhan Ma'arif Lubis

ABSTRAK

Nama : Farhan Ma'arif Lubis
NIM : 0110220116
Program Studi : Teknik Informatika
Judul : Pengembangan Aplikasi *Bridging* BPJS & Remunerasi Dokter
Pada Rumah Sakit Xyz Menggunakan Metode Spiral

Penelitian ini bertujuan untuk meningkatkan manajemen keuangan dalam sektor kesehatan dengan mengembangkan aplikasi yang mengintegrasikan klaim BPJS dan remunerasi dokter. Latar belakang penelitian ini muncul dari kebutuhan peningkatan transparansi dan akuntabilitas dalam pengelolaan keuangan rumah sakit. Melalui aplikasi ini, dilakukan konversi klaim menggunakan sistem *Indonesia Case Based Groups* (INA-CBGs) untuk alokasi dana yang lebih proporsional dan adil. Aplikasi dikembangkan dengan memanfaatkan metode Spiral, yang memungkinkan adaptasi terhadap perubahan kondisi dan kebutuhan melalui iterasi yang berkelanjutan serta integrasi umpan balik secara *real-time*. Hasil pengujian aplikasi menunjukkan bahwa *endpoint /ppn* memiliki waktu respons paling cepat dengan rata-rata 329ms dan *endpoint /shifting-journal* menunjukkan waktu respons paling lama dengan rata-rata 28.33 detik. Secara keseluruhan, aplikasi dapat menangani beban kerja tinggi dan menghasilkan data yang cepat dan akurat, berpotensi besar untuk meningkatkan efisiensi operasional rumah sakit. Hasil penelitian ini diharapkan dapat menjadi referensi dalam pengembangan sistem informasi keuangan yang lebih efektif di sektor kesehatan.

Kata kunci: Manajemen Keuangan Kesehatan, Klaim BPJS, Remunerasi Dokter, Metode Spiral, INA-CBGs.

ABSTRACT

Name : Farhan Ma'arif Lubis
NIM : 0110220116
Study Program : *Informatics Engineering*
Title : *Development of BPJS Bridging Application & Doctor Remuneration At Xyz Hospital using the spiral method*

This research aims to enhance financial management within the healthcare sector by developing an application that integrates BPJS claims and doctor remuneration. The background of this study arises from the need to improve transparency and accountability in hospital financial management. Through this application, claims are converted using the Indonesia Case Based Groups (INA-CBGs) system for more proportional and fair fund allocation. The application was developed using the Spiral method, which facilitates adaptation to changing conditions and needs through continuous iterations and real-time feedback integration. Testing results reveal that the /ppn endpoint has the fastest response time with an average of 329ms, while the /shifting-journal endpoint shows the longest response time with an average of 28.33 seconds. Overall, the application can handle high workloads and produce fast and accurate data, holding significant potential to improve hospital operational efficiency. The findings of this research are expected to serve as a reference for developing more effective financial information systems in the healthcare sector.

Keywords: *Healthcare Financial Management, BPJS Claims, Doctor Remuneration, Spiral Method, INA-CBGs.*

DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PERNYATAAN ORISINALITAS... Error! Bookmark not defined.	
HALAMAN PENGESAHAN..... Error! Bookmark not defined.	
KATA PENGANTAR	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.. Error! Bookmark not defined.	
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS Error! Bookmark not defined.	
ABSTRAK	vii
<i>ABSTRACT</i>	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xv
BAB I	1
PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan dan Manfaat Penelitian.....	3
1.4 Batasan Masalah.....	4
1.5 Sistematika Penulisan.....	4
BAB II.....	6
KAJIAN LITERATUR.....	6
2.1 Definisi Sistem Tarif INA-CBGs.....	6
2.2 Sejarah Tarif INA-CBGs di Indonesia.....	7
2.3 Sistem Coding dalam INA-CBGs.....	8
2.4 Remunerasi Dokter dari Pembagian Dana Klaim BPJS Kesehatan.....	10
2.5 Metode Konversi dan Proporsi dalam Alokasi Pendapatan BPJS di Rumah Sakit	12
2.5.1 Konversi	12
2.5.1.1 Tahap Pertama: Metode Konversi	12

2.5.1.2	Tahap Kedua: Distribusi	15
2.5.1.3	Keunggulan Metode Konversi	17
2.5.1.4	Kelemahan Metode Konversi	18
2.5.2	Metode Proporsi	18
2.5.2.1	Tahap Pertama - Proporsi	19
2.5.2.2	Tahap Kedua - Konversi	22
2.5.2.3	Tahap Ketiga : Distribusi	23
2.6	Metode Spiral:	24
2.6.1	Metode Pengujian dalam Metode Spiral	26
2.7	<i>Framework</i> Laravel:	26
2.8	React.js:	27
2.9	Daisy UI:	28
2.10	Teknologi Yang Digunakan	29
2.11	Penelitian Terdahulu	30
BAB III		33
METODOLOGI PENELITIAN		33
3.1	Tahapan Penelitian	33
3.2	Rancangan Penelitian	34
3.2.1	Jenis Penelitian	35
3.2.2	Metode Analisis Data	36
3.2.3	Metode Pengumpulan Data	37
3.2.4	Metode Pengujian	38
3.2.5	Metode Implementasi dan Evaluasi	39
3.2.6	Lingkungan Pengembangan	40
3.2.7	Metode Pengembangan Perangkat Lunak	41
BAB IV		43
IMPLEMENTASI DAN EVALUASI		43
4.1	Identifikasi Kebutuhan Pengguna	43
4.2	Rancangan Penelitian	44
4.2.1	Cara Kerja	44
4.2.2	Arsitektur Sistem	44
4.2.3	Tahapan iterasi	45

4.2.4	<i>Flowchart</i>	49
4.2.5	Wireframe.....	61
4.2.6	<i>ERD Diagram</i>	68
4.2.7	<i>Class Diagram</i>	71
4.3	Hasil Implementasi.....	73
4.3.1	Menu Aplikasi:	73
4.4	Hasil Pengujian	80
4.4.1	<i>Blackbox Testing</i>	81
4.4.2	<i>Load Testing</i>	83
4.5	Wawancara Dengan Pengguna.....	99
4.6	Fitur Inovatif dan Rencana Pengembangan Selanjutnya	101
BAB V.....		103
KESIMPULAN DAN SARAN.....		103
5.1	Kesimpulan	103
5.2	Saran.....	103
DAFTAR PUSTAKA		105



STT - NF

DAFTAR GAMBAR

Gambar 2.1: Jasa Pelayanan.....	15
Gambar 2.2: Persentase Jasa Pelayanan.....	16
Gambar 2.3: Persentase Jasa Pelayanan Metode Proporsi.....	23
Gambar 3.1 Tahapan Penelitian [16]	33
Gambar 4.1 <i>Flowchart Login</i>	49
Gambar 4.2 <i>Flowchart Dashboard</i>	50
Gambar 4.3 <i>Flowchart Income Conversion</i>	51
Gambar 4.4 <i>Flowchart Piutang BPJS</i>	52
Gambar 4.5 <i>Flowchart Data Hasil Konversi</i>	52
Gambar 4.6 <i>Flowchart Laporan Keuangan</i>	53
Gambar 4.7 <i>Flowchart Jurnal</i>	53
Gambar 4.8 <i>Flowchart Shifting</i>	53
Gambar 4.9 <i>Flowchart Allocation & Distribution</i>	54
Gambar 4.10 <i>Flowchart Data Dokter</i>	54
Gambar 4.11 <i>Flowchart KPI Dokter</i>	55
Gambar 4.12 <i>Flowchart Tarif Dokter</i>	55
Gambar 4.13 <i>Flowchart Pasien Non BPJS & BPJS</i>	56
Gambar 4.14 <i>Flowchart Data KPI</i>	56
Gambar 4.15 <i>Flowchart Data Vendor Obat</i>	57
Gambar 4.16 <i>Flowchart Data Kelas Tarif</i>	57
Gambar 4.17 <i>Flowchart Pelayanan Rumah Sakit</i>	58
Gambar 4.18 <i>Flowchart Jasa Sarana</i>	58
Gambar 4.19 <i>Flowchart Jasa Pelayanan</i>	59
Gambar 4.20 <i>Flowchart Tarif Rumah Sakit</i>	59
Gambar 4.21 <i>Flowchart Profil</i>	60
Gambar 4.22 <i>Flowchart Logout</i>	60
Gambar 4.23 <i>Wireframe Login</i>	61
Gambar 4.24 <i>Wireframe Dashboard</i>	61
Gambar 4.25 <i>Wireframe Income Conversion</i>	61
Gambar 4.26 <i>Wireframe Piutang BPJS</i>	62

Gambar 4.27 <i>Wireframe</i> Data Konversi.....	62
Gambar 4.28 <i>Wireframe</i> Laporan Keuangan	62
Gambar 4.29 <i>Wireframe</i> Jurnal	63
Gambar 4.30 <i>Wireframe</i> Remunerasi	63
Gambar 4.31 <i>Wireframe</i> Data Dokter	63
Gambar 4.32 <i>Wireframe</i> Data KPI Dokter.....	64
Gambar 4.33 <i>Wireframe</i> Darif Dokter	64
Gambar 4.34 <i>Wireframe</i> Pasien Non BPJS.....	64
Gambar 4.35 <i>Wireframe</i> Data Pasien BPJS	65
Gambar 4.36 <i>Wireframe</i> Data KPI.....	65
Gambar 4.37 <i>Wireframe</i> Data Vendor Obat	65
Gambar 4.38 <i>Wireframe</i> Kelas Tarif.....	65
Gambar 4.39 <i>Wireframe</i> Pelayanan Rumah Sakit.....	66
Gambar 4.40 <i>Wireframe</i> Jasa Sarana	66
Gambar 4.41 <i>Wireframe</i> Jasa Pelayanan.....	66
Gambar 4.42 <i>Wireframe</i> Tarif Rumah Sakit.....	66
Gambar 4.43 <i>Wireframe</i> Profil.....	67
Gambar 4.44 <i>Wireframe</i> Header	67
Gambar 4.45 <i>Wireframe</i> Footer	67
Gambar 4.46 <i>ERD Diagram</i>	68
Gambar 4.47 <i>Class Diagram</i>	71
Gambar 4.48 Aplikasi - <i>Dashboard</i>	73
Gambar 4.49 Aplikasi – <i>Income Conversion</i>	74
Gambar 4.50 Aplikasi – <i>Income Conversion Uploaded</i>	74
Gambar 4.51 Aplikasi – Piutang BPJS	74
Gambar 4.52 Aplikasi – Data Konversi	75
Gambar 4.53 Aplikasi – Laporan Keuangan.....	75
Gambar 4.54 Aplikasi – Jurnal.....	75
Gambar 4.55 Aplikasi – Data Dokter.....	76
Gambar 4.56 Aplikasi – Jurnal.....	76
Gambar 4.57 Aplikasi – Tarif Dokter	76
Gambar 4.58 Aplikasi – Pasien Non BPJS	77

Gambar 4.59 Aplikasi – Pasien BPJS	77
Gambar 4.60 Aplikasi – Data KPI	77
Gambar 4.61 Aplikasi – Data KPI 2	78
Gambar 4.62 Aplikasi – Vendor Obat.....	78
Gambar 4.63 Aplikasi – Kelas Tarif	78
Gambar 4.64 Aplikasi – Persentase Pelayanan	79
Gambar 4.65 Aplikasi – Jasa Sarana.....	79
Gambar 4.66 Aplikasi – Jasa Pelayanan 1	79
Gambar 4.67 Aplikasi – Jasa Pelayanan 2	79
Gambar 4.68 Aplikasi – Tarif Rumah Sakit.....	80
Gambar 4.69 Aplikasi – Profil	80
Gambar 4.70 Instalasi K6.....	84
Gambar 4.71 Skrip Pengujian K6	85
Gambar 4.72 Menjalankan Pengujian K6	85
Gambar 4.73 Contoh Hasil Pengujian Pembuatan Jurnal	89
Gambar 4.74 Contoh Hasil Pengujian Piutang BPJS	90
Gambar 4.75 Contoh Hasil Pengujian PPN	90
Gambar 4.76 Contoh Hasil Pengujian Data Konversi	91
Gambar 4.77 Contoh Hasil Pengujian Shifting Remunerasi.....	92
Gambar 4.78 Contoh Hasil Pengujian Shifting Remunerasi.....	92

STT - NF

DAFTAR TABEL

Table 2.1: Kode INA-CBGs.....	9
Table 2.2: Distribusi Jasa Pemeriksaan Dokter per Penerima	17
Table 2.3: Distribusi Jasa Pemeriksaan Darah Rutin per Penerima.....	17
Table 2.4: Distribusi Jasa Pemeriksaan Rontgen Thorax per Penerima	17
Table 2.5: Perbedaan Metode Konversi dan Proporsi.....	19
Table 2.6: Distribusi Jasa Pemeriksaan Dokter per Penerima	23
Table 2.7: Distribusi Jasa Pemeriksaan Darah Rutin per Penerima.....	24
Table 2.8: Distribusi Jasa Pemeriksaan Rontgen Thorax per Penerima	24
Table 2.9: Tabel Penelitian Terdahulu	30
Table 4.1: Tabel Tahapan Iterasi.....	45
Table 4.2: Tabel <i>Blackbox Testing</i>	81
Table 4.3 Tabel Hasil Pengujian /shifting-journal	89
Table 4.4: Tabel Hasil Pengujian /piutang-bpjs-tak-tertagih	89
Table 4.5: Tabel Hasil Pengujian /ppn.....	90
Table 4.6: Tabel Hasil Pengujian /data-konversi	91
Table 4.7: Tabel Hasil Pengujian /shifting-remunerasi.....	91
Table 4.8: Tabel Hasil Pengujian /allocation-distribution-remunerasi	92

STT - NF

BAB I

PENDAHULUAN

Tujuan bab ini adalah untuk memberikan konteks penelitian dan menguraikan masalah yang perlu dipecahkan agar temuan dapat berkontribusi terhadap efisiensi keuangan Rumah Sakit XYZ.

1.1 Latar belakang

Manajemen kesehatan, khususnya aspek keuangan yang terkait dengan klaim BPJS dan penggajian dokter, menimbulkan sejumlah tantangan yang memerlukan penelitian dan solusi yang mendalam. Beberapa permasalahan yang ada mendorong perlunya penelitian ini.

Salah satu permasalahan utama terletak pada proses kompleks alokasi dana dari BPJS ke pendapatan rumah sakit berdasarkan Indonesia Case Based Groups (INA-CBGs). Kompleksitas proses ini membutuhkan perhatian khusus karena tidak adanya sistem yang terstruktur untuk menentukan distribusi proporsional dana di antara berbagai layanan medis dalam satu klaim. Tantangan ini menghambat efisiensi penggunaan dana dan dapat menyebabkan ketidaksesuaian dalam kompensasi berbagai layanan yang diberikan oleh rumah sakit.

Siklus pendapatan rumah sakit secara tidak langsung dipengaruhi oleh strategi pemerintah dalam menerapkan Jaminan Kesehatan Nasional (JKN). Tarif yang berlaku di rumah sakit dan JKN terbilang unik. Peraturan pemerintah terkait kesehatan menjadi dasar penetapan tarif JKN. Tarif ditentukan berdasarkan diagnosis, digabungkan, dan mengasumsikan bahwa pengobatan pasien diberikan sesuai standar atau mengikuti jalur klinis, atau yang sekarang disebut Tarif INA-CBGs. Di sisi lain, sebagian besar rumah sakit di Indonesia membebankan biaya kepada pasien berdasarkan biaya layanan, artinya jumlah yang dibayarkan pasien bergantung pada layanan yang mereka terima. Akibatnya, catatan pendapatan rumah sakit menunjukkan perbedaan yang besar; misalnya pada tahun 2018, peneliti Sri Wahyuni dkk. membandingkan biaya aktual rumah sakit dengan tarif INA-CBG pada pasien thalassemia di RSUD Zainoel Abidin Banda Aceh dan ditemukan perbedaan sebesar 34,27% [1].

Selain itu, sistem remunerasi untuk dokter masih kurang mendapatkan pendekatan yang komprehensif dan otomatis. Metode saat ini sering melibatkan perhitungan manual, yang berpotensi mengakibatkan kesalahan dan ketidakefisienan. Situasi ini menimbulkan kekhawatiran terhadap kompensasi yang adil dan akurat bagi para profesional medis, staf non-medis, fasilitas, dan unit biaya lainnya yang terlibat dalam proses kesehatan.

Dukungan untuk urgensi penelitian ini dapat ditemukan dalam literatur dan studi terkait yang menyoroti tantangan berkelanjutan dalam manajemen keuangan kesehatan. Referensi seperti ini memberikan gambaran tentang pentingnya mengatasi masalah ini untuk meningkatkan akuntabilitas dan transparansi secara keseluruhan di sektor kesehatan.

Meskipun beberapa penelitian telah menyentuh aspek manajemen keuangan dalam pelayanan kesehatan, solusi yang holistik dan terintegrasi, mencakup baik klaim BPJS maupun penggajian dokter, masih jarang dieksplorasi. Untuk mengatasi kekurangan ini, penelitian ini akan fokus pada pembuatan aplikasi baru yang mengotomatiskan dan menggabungkan prosedur-prosedur ini untuk efisiensi dan akurasi yang lebih baik sekaligus memecahkan masalah yang ditemukan.

Secara ringkas, kompleksitas yang ada dalam distribusi dana BPJS menunjukkan perlunya solusi komprehensif. Penelitian ini bertujuan untuk memberikan kontribusi berupa aplikasi inovatif yang tidak hanya mengidentifikasi dan memperbaiki masalah yang ada, tetapi juga sejalan dengantujuan lebih luas untuk meningkatkan akuntabilitas dan transparansi dalam proses keuangan di ranah kesehatan.

1.2 Rumusan Masalah

Setelah menyiapkan tahapan dengan informasi latar belakang, maka pertanyaan penelitian pada penelitian ini adalah: bagaimana menggunakan metode pengembangan Spiral untuk menciptakan aplikasi yang menjembatani BPJS dan sistem kompensasi dokter yang efisien, transparan, dan akuntabel?

1.3 Tujuan dan Manfaat Penelitian

Tujuan dari penelitian ini adalah:

1. Mengembangkan Aplikasi *Bridging*

Mengembangkan aplikasi *bridging* yang mampu mengintegrasikan klaim BPJS dengan sistem manajemen keuangan rumah sakit, termasuk penghitungan dan distribusi dana berdasarkan *Indonesia Case Based Groups* (INA-CBGs).

2. Mengimplementasikan Metode Pengembangan Spiral

Menerapkan metode pengembangan Spiral untuk memastikan iterasi berkelanjutan dalam pengembangan aplikasi, memungkinkan perbaikan terus-menerus berdasarkan umpan balik pengguna dan hasil pengujian.

3. Mengotomatisasi Penghitungan Kontribusi Layanan

Merancang algoritma untuk menghitung kontribusi relatif dari setiap layanan dalam satu klaim BPJS, serta mendistribusikan dana secara proporsional sesuai dengan kontribusi masing-masing layanan.

4. Membangun Sistem Remunerasi Dokter yang Adil

Mengembangkan sistem remunerasi yang transparan dan adil untuk dokter dan tenaga medis lainnya, berdasarkan kontribusi mereka dalam proses pelayanan kesehatan.

5. Mengoptimalkan Transparansi dan Akuntabilitas keuangan rumah sakit

Memastikan bahwa aplikasi yang dikembangkan meningkatkan transparansi dan akuntabilitas dalam pengelolaan dana klaim BPJS dan remunerasi dokter, sehingga semua pihak yang terlibat dapat memantau dan memverifikasi alokasi dana.

Manfaat Penelitian:

1. Meningkatkan efisiensi dalam pengelolaan dana klaim BPJS, memastikan alokasi yang tepat dan proporsional berdasarkan INA-CBGs.
2. Memastikan keadilan dalam kompensasi dokter dan pihak terlibat lainnya, berdasarkan kontribusi nyata dalam proses pelayanan kesehatan.

1.4 Batasan Masalah

1. Aplikasi ini akan difokuskan pada proses alokasi dana dari BPJS ke rumah sakit berdasarkan INA-CBGs dan sistem remunerasi dokter.
2. Cakupan perhitungan remunerasi hanya dibatasi terhadap komponen yang terdapat pada INA-CBGs, artinya aplikasi hanya akan menghitung dan mengalokasikan dana untuk unit pelayanan yang terdefinisi dan dapat dihitung dalam sistem INA-CBGs.
3. Batasan melibatkan implementasi pada satu rumah sakit sebagai studi kasus.
4. Aplikasi ini berbasis web dan hanya memiliki satu *login*, yaitu untuk admin atau bagian keuangan saja.
5. Data yang dapat diimpor ke dalam aplikasi hanya dapat berupa file *Excel*.

1.5 Sistematika Penulisan

Berikut ini gambaran detail cara penulisan tugas akhir ini:

1. BAB I Pendahuluan mencakup latar belakang yang memperkenalkan konteks masalah, rumusan masalah yang menjelaskan tantangan utama, tujuan dan manfaat penelitian yang menggambarkan sasaran utama serta dampak yang diharapkan, batasan masalah yang mengidentifikasi lingkup dan limitasi penelitian, dan sistematika penulisan yang memberikan gambaran umum tentang struktur tugas akhir ini.
2. BAB II Kajian Literatur akan mendalami teori dan konsep yang relevan dengan pengembangan aplikasi *bridging* BPJS dan sistem remunerasi dokter, termasuk teori manajemen informasi kesehatan dan sistem pembayaran kesehatan di Indonesia.
3. BAB III Metodologi Penelitian menjelaskan langkah-langkah penelitian yang diambil, dengan penekanan pada pendekatan pengembangan spiral. Bab ini mencakup proses penelitian dan rancangan sistem berdasarkan analisis kebutuhan pengguna.
4. BAB IV Implementasi dan Evaluasi terdiri dari analisis kebutuhan pengguna dan sistem yang menjadi dasar implementasi aplikasi, deskripsi rancangan sistem yang dibangun, dan evaluasi terhadap aplikasi yang mencakup aspek fungsionalitas dan kinerja.

5. BAB V Kesimpulan dan Saran menyajikan ringkasan temuan dan hasil penelitian, serta memberikan rekomendasi untuk potensi peningkatan atau fitur baru yang dapat disertakan dalam aplikasi di masa mendatang.



STT - NF

BAB II

KAJIAN LITERATUR

Bab kajian literatur ini berisi aspek terkait pengembangan aplikasi *Bridging* BPJS dari tarif INA-CBGs dan remunerasi dokter sebagai solusi peningkatan akuntabilitas dan transparansi pendapatan rumah sakit. Dalam bab ini, akan diuraikan berbagai aspek penting yang menjadi landasan teoritis dan pemahaman yang mendalam terhadap topik penelitian.

2.1 Definisi Sistem Tarif INA-CBGs

Salah satu skema penggantian biaya perawatan medis yang potensial adalah tarif "INA-CBGs (*Indonesia Case Base Groups*)". Dalam sistem ini, biaya pelayanan kesehatan sudah ditetapkan sebelumnya sebelum pasien menerima perawatan. Konsep ini melibatkan pengelompokan pasien ke dalam kategori-kategori yang serupa berdasarkan diagnosis dan kasus yang memiliki kesamaan karakteristik klinis [2]

Untuk memastikan pemanfaatan sumber daya yang konsisten dan inklusi pasien dengan gambaran klinis yang sebanding, sistem Casemix INA-CBGs mengategorikan episode perawatan pasien. Perawatan pasien diganti melalui kelompok berbasis kasus (CBGs) yang didasarkan pada diagnosis atau kejadian yang secara substansial serupa. Rumah sakit akan diberi kompensasi sesuai dengan rata-rata pengeluaran untuk kategori diagnosis tertentu. Untuk memastikan bahwa semua pasien rawat inap dimasukkan ke dalam kelompok dengan gejala klinis dan biaya pengobatan yang sebanding, setiap langkah pelayanan kesehatan dikategorikan ke dalam kelompok dengan arti yang kurang lebih sama.

Program ini, INA-CBGs, merupakan perpanjangan dari program sebelumnya, INA-DRGs (*Indonesia Diagnosis Related Groups*), yang digunakan untuk pengelompokan terkait diagnosis di Indonesia. Sejak tahun 2008 dan seterusnya, penerapan INA-CBGs menggantikan INA-DRGs. Casemix, yang dikembangkan dan digunakan oleh *United Nations University-International Institute for Global Health*, merupakan dasar metode yang digunakan dalam INA-CBGs. Daripada merinci biaya berdasarkan layanan yang diberikan, rumah sakit dan pembayar yang

terlibat dalam pembayaran CBG hanya memberikan diagnosis pasien dan nomor DRG. Ini bisa berupa jumlah yang ditentukan pemerintah atau jumlah yang telah disepakati oleh penyedia layanan dan perusahaan asuransi untuk setiap diagnosis. Setelah mempertimbangkan diagnosis spesifik dan penyakit yang dihadapi, penulis telah mengevaluasi perkiraan durasi rawat inap untuk setiap pasien.

2.2 Sejarah Tarif INA-CBGs di Indonesia

Pendekatan Fee For Care pertama kali digunakan dalam sistem pembayaran layanan kesehatan di Indonesia. Akibatnya, tidak ada pedoman yang seragam mengenai besarnya biaya perawatan kesehatan di seluruh negeri, dan konsumen akhirnya mengeluarkan biaya yang berbeda-beda ketika mereka mengunjungi rumah sakit. Kondisi ini merugikan konsumen, terutama golongan masyarakat miskin. Untuk mengatasi masalah ini, Indonesia memperkenalkan sistem pembayaran INA DRG pada tahun 2005, yang kemudian digantikan oleh INA CBG pada tahun 2010.

Pergantian tersebut dilatarbelakangi oleh kelemahan INA DRG, seperti keterbatasan pada kasus penyakit akut, ketidakadeguan tarif untuk kasus subakut dan kronik, serta prosedur khusus seperti MRI. INA CBG kemudian diperkenalkan untuk mengklasifikasikan episode perawatan pasien berdasarkan diagnosis yang serupa, dengan pembayaran berdasarkan rata-rata biaya kelompok diagnosis.

Rumah sakit dan pembayar dalam sistem INA CBG kini hanya menggunakan kode diagnostik dan CBG untuk mengelompokkan biaya, bukan layanan tertentu. Pembayaran telah disepakati sebelumnya, termasuk estimasi lama perawatan pasien. Ini memberikan kepastian kepada pasien dan mendorong efisiensi di rumah sakit.

Manfaatnya melibatkan pasien yang mendapatkan perhatian lebih fokus berdasarkan keparahan penyakit, pengurangan pemeriksaan berlebihan, dan kepastian biaya perawatan. Bagi rumah sakit, ini meningkatkan mutu pelayanan dan efisiensi, sementara pemerintah mendapat manfaat dari pengalokasian anggaran yang lebih efisien dan peningkatan kepuasan pasien. Dasar hukum penerapan INA-CBGs mempunyai dasar hukum antara lain:

- a. UU Nomor 40 Tahun 2004 Tentang Sistem Jaminan Sosial Nasional (SJSN)

- b. UU Nomor 29 Tahun 2004 Tentang Praktik Kedokteran
- c. UU Nomor 36 Tahun 2009 Tentang Kesehatan
- d. UU Nomor 44 Tahun 2009 Tentang Rumah Sakit
- e. SK Direktur Jenderal Bina Upaya Kesehatan Nomor HK.03.05/I/589/2011 Tentang Kelompok Kerja *Centre for Casemix* tahun 2011.

2.3 Sistem Coding dalam INA-CBGs

Logika perangkat lunak yang digunakan untuk menghitung laju implementasi *Case Mix* INA-CBGs didasarkan pada ICD 9 CM untuk aktivitas atau prosedur dan pedoman ICD 10 untuk diagnosis, oleh karena itu pengkodean memainkan peran yang sangat penting. Diagnosis dan Prosedur menentukan besaran tarif yang ditampilkan dalam program INA-CBGs. Tarif tersebut akan dipengaruhi oleh kesalahan penulisan diagnosis. Kisaran tarif yang lebih luas dimungkinkan. Penting untuk mengidentifikasi kelainan primer dan sekunder untuk sampai pada diagnosis sesuai dengan pedoman CBGs. Komplikasi dan penyakit penyerta adalah komponen diagnosis bersamaan. [4]

Intensitas gejala yang dialami pasien, yang dipengaruhi oleh diagnosis yang menyertainya, pada gilirannya dapat mengubah besarnya angka kejadiannya. Jumlah waktu yang dihabiskan pasien di rumah sakit berhubungan langsung dengan kemungkinan timbulnya masalah selama perawatan. Biaya terapi akan meningkat jika diperlukan waktu lebih lama dari yang diharapkan untuk menyelesaikan masalah. Tarif diterapkan pada paket sebenarnya dalam logika perangkat lunak INA-CBGs apabila terjadi tingkat keparahan level 2 atau level 3. Kemampuan untuk mengidentifikasi diagnosis primer atau sekunder sangat penting bagi pembuat kode jika beberapa diagnosis ditemukan pada akhir masa pengobatan.

- a. Empat set kode berbeda membentuk INA-CBGs. I-4-10-I merupakan contoh kode INA-CBGs yang menunjukkan diagnosis infark miokard akut ringan pada seorang pasien. Grup Utama Casemix (CMG) ditunjukkan pada Sub Grup 1. INA CBG memiliki 31 kode untuk CMG. Beberapa kasus dimana INA-CBGs menggunakan kode CMG ditunjukkan di bawah ini.[5]

Table 2.1: Kode INA-CBGs

No	Casemix Main Groups (CMG)	CMG Codes
1	<i>Central Nervous System Groups</i>	G
2	<i>Eye and Adnexa Groups</i>	H
3	<i>Ear, Nose, Mouth, & Throat Groups</i>	U
4	<i>Respiratory System Groups</i>	J
5	<i>Cardiovascular System Groups</i>	I
6	<i>Digestive System Groups</i>	K
7	<i>Hepatobiliary & Pancreatic System Groups</i>	B
8	<i>Musculoskeletal System & Connective Tissue Groups</i>	M
9	<i>Skin, Subcutaneous Tissue & Breast Groups</i>	L
10	<i>Endocrine System, Nutrition & Metabolisme System</i>	E

- b. Jenis kasusnya ditunjukkan pada Sub Grup 2, dengan sistem INA-CBGs menggunakan kelompok kasus bernomor 1 sampai dengan 9. Jika terjadi kesalahan maka akan ditampilkan kelompok 10. Berikut penjelasan detail kodenya:
- (1) Prosedur Rawat Inap
 - (2) Prosedur Besar Rawat Jalan
 - (3) Prosedur Signifikan Rawat Jalan
 - (4) Rawat Inap Bukan Prosedur
 - (5) Rawat Jalan Bukan Prosedur
 - (6) Rawat Inap Kebidanan
 - (7) Rawat Jalan Kebidanan
 - (8) Rawat Inap Neonatal
 - (9) Rawat Jalan Neonatal
 - (10) Error
- c. Kode CBGs ditampilkan pada Sub Grup Ketiga. Dengan 789 kode untuk rawat inap dan 288 kode untuk rawat jalan, kode INA-CBGs memiliki kekuatan 1077 kode.
- d. Tingkat keparahannya ditunjukkan pada Sub Kelompok 4. Tingkat Keparahan I paling ringan, Tingkat Keparahan II paling sedang, dan Tingkat Keparahan III paling parah.

2.4 Remunerasi Dokter dari Pembagian Dana Klaim BPJS Kesehatan

Pada tahun 2019, dari total populasi 269 juta jiwa, 84% atau 224 juta jiwa telah terdaftar dalam Program Jaminan Kesehatan Nasional (JKN), menurut Dewan Jaminan Sosial Badan Penyelenggara Jaminan Sosial (BPJS) Kesehatan. 2020). Dengan puncaknya sebesar 444 per 10.000 peserta pada tahun 2018, angka rawat inap juga meningkat dari 319 per 10.000 peserta pada tahun 2014, atau merupakan kenaikan mutlak sebesar 39%. Dengan semakin banyaknya pasien yang diperkirakan akan datang, rumah sakit mungkin akan mengalami peningkatan pendapatan.[6]

Penatausahaan dan penggunaan sumber keuangan BPJS Pembayaran klaim kesehatan diubah sesuai peraturan perundang-undangan bagi rumah sakit/pusat milik pemerintah/pemerintah daerah yang belum berstatus Badan Layanan Umum Daerah (BLUD) atau Badan Layanan Umum (BLU), dan sesuai ketentuan BLU atau BLUD bagi yang memilikinya. Pelayanan yang diberikan oleh fasilitas kesehatan rawat jalan lanjutan milik pemerintah (FKRTL) menyumbang 30–50% dari keseluruhan pendapatan fasilitas tersebut. Namun jika menyangkut kepemilikan pribadi, tanggung jawab untuk menyediakan akomodasi ada pada institusi kesehatan (Permenkes Nomor 28 Tahun 2014). [7]

Fasilitas dan pelayanan kesehatan rawat inap, rawat jalan, dan gawat darurat merupakan komponen tarif rumah sakit. Rumah Sakit mendapat pembayaran atas penggunaan gedung, barang non-medis, obat-obatan, serta peralatan dan perlengkapan medis habis pakai sebagai bagian dari komponen layanan fasilitas. Pelayanan tenaga kesehatan dan pelayanan kepegawaian lainnya merupakan komponen pelayanan, yaitu bayaran penyedia atas pelayanan yang diberikannya kepada pasien. Mengingat keberlangsungan pelayanan, maka penggunaan tarif atas produk dan jasa serta belanja modal minimal 40% (Peraturan Menteri Kesehatan Nomor 85 Tahun 2015). [8]

Gaji, honorarium, tunjangan tetap, insentif, tantiem kinerja, pesangon, pensiun, dan bentuk imbalan kerja pegawai lainnya semuanya dianggap remunerasi berdasarkan Peraturan Menteri Kesehatan Nomor 63 Tahun 2016. Besaran imbalan BLU dihitung dengan memperhatikan [9]

- a. Proporsionalitas, artinya mempertimbangkan besaran dan jumlah aset yang dikelola BLU serta derajat pelayanannya;
- b. Parity, yaitu dengan fokus pada sektor jasa yang berkaitan erat;
- c. Kesesuaian yaitu memodifikasi potensi pendapatan BLU; Dan
- d. Seberapa baik BLU menjalankan bisnisnya sehari-hari, termasuk metrik keuangan, kualitas layanan, dan manfaat bagi masyarakat.

Berikut ini unsur-unsur sistem kompensasi:

- a. Pembayaran untuk jabatan (*pay for position*) adalah imbalan yang diberikan atas kinerja baik yang dimaksudkan untuk memotivasi dan mengakui tugas berkelanjutan yang terlibat dalam melaksanakan proses pekerjaan. Terdapat gaji pokok dan tunjangan struktural/fungsional yang diberikan dalam bentuk rupiah murni dan pendapatan BLU yang menjadi sistem kompensasi Pegawai Negeri Sipil (PNS). Dibayar dari pendapatan BLU, struktur kompensasi bagi non-PNS sama dengan pegawai negeri ditambah tunjangan;
- b. Pembayaran untuk kinerja (*pay for performance*) adalah sejenis penghargaan kinerja yang memanfaatkan insentif dan bonus untuk mengakui dan menghargai individu atas upaya mereka dan membantu mereka tetap termotivasi untuk mencapai tujuan kinerja mereka.
- c. Pembayaran untuk perorangan (*pay for people*) merupakan jaring pengaman yang menyediakan sumber daya untuk membantu masyarakat hidup nyaman dan bijaksana berdasarkan kebutuhan unik mereka; sumber daya ini dapat berupa premi asuransi, kompensasi pesangon, atau pensiun.

Indikator evaluasi untuk menentukan kompensasi dokter termasuk pengalaman dan masa kerja (*basic index*), keterampilan dan ilmu pengetahuan (*competency index*), risiko kerja (*risk index*), tingkat kegawatdaruratan (*emergency index*), jabatan yang disandang (*position index*), dan hasil kinerja (*performance index*) [10]

Sistem kompensasi ini dirancang untuk memastikan bahwa dokter dan tenaga kesehatan lainnya mendapatkan remunerasi yang adil dan proporsional berdasarkan kontribusi mereka dalam memberikan pelayanan kesehatan. Alokasi insentif sering

kali menjadi tantangan, karena kebutuhan untuk memenuhi kriteria keadilan dan memastikan semua pihak yang terlibat merasa dihargai.

2.5 Metode Konversi dan Proporsi dalam Alokasi Pendapatan BPJS di Rumah Sakit

2.5.1 Konversi

Proses konversi nilai dari satu sistem satuan ke sistem satuan lainnya disebut konversi. Yang dimaksud dengan “Pelayanan” adalah proses konversi tarif sistem paket INA-CBGs menjadi harga pelayanan berdasarkan bobot “mediator”, persentase iuran, atau RVU. Dua langkah yang terlibat dalam proses konversi ini adalah:

- a. Tahap pertama yaitu melakukan konversi dengan merubah tarif sistem paket INA-CBGs menjadi FFS melalui perhitungan bobot atau RVU; dan
- b. Tahap kedua yaitu melakukan distribusi dari hasil konversi tersebut.

2.5.1.1 Tahap Pertama: Metode Konversi

Pada tahap ini proses perhitungan *Relative Value Unit* (RVU) dari setiap komponen tercantum pada lembar tagihan rumah sakit kepada pasien sesuai tarif layana rumah sakit (FFS). Oleh karena itu, sangat penting untuk memastikan tarif rumah sakit selain rasio JS dan JP, atau layanan fasilitas, terhadap tarif rumah sakit. Layanan Fasilitas plus Layanan Pelayanan adalah metode untuk menentukan biaya rumah sakit.

Berdasarkan komponen Fasilitas dan Pelayanan yang membentuk tarif rumah sakit, terdapat tiga kategori pelayanan yang ditawarkan rumah sakit:

- a. Pelayanan Fasilitas (JS) dan Pelayanan Pelayanan (JP), contoh: Pemeriksaan Rontgen Thorax mempunyai pelayanan fasilitas (JS) dan pelayanan pelayanan (JP) sehingga tarifnya merupakan penjumlahan dari pelayanan fasilitas dan pelayanan pelayanan.
- b. Pelayanan Fasilitas Saja (JS), contoh: akomodasi kamar rawat inap hanya mempunyai komponen pelayanan fasilitas (JS) dan tidak ada pelayanan, sehingga tarifnya sama dengan pelayanan fasilitas.

- c. Hanya Jasa Pelayanan (JP), contoh: Visite dokter spesialis yaitu hanya tarif jasa pelayanan.

Contoh perhitungan konversi kasus rawat jalan di rumah sakit:

a. Pemeriksaan Dokter

Jasa Sarana : Rp10.000
 Jasa Pelayanan : Rp30.000 +
 Tarif : Rp40.000

b. Pemeriksaan Darah Rutin

Jasa Sarana : Rp25.000
 Jasa Pelayanan : Rp10.000 +
 Tarif : Rp35.000

c. Pemeriksaan Rontgen Thorax

Jasa Sarana : Rp40.000
 Jasa Pelayanan : Rp40.000 +
 Tarif : Rp80.000

d. Obat dan Bahan Habis Pakai

Jasa Sarana : Rp50.000
 Jasa Pelayanan : Rp0+
 Tarif : Rp50.000

Sehingga total tagihan pasien rawat jalan sebesar : Rp40.000 + Rp 35.000
 + Rp80.000 + Rp50.000 = Rp205.000

Komponen-komponen tersebut jika diuraikan:

- | | | |
|----|-----------------------------------|-------------------|
| a. | Jasa Sarana Px Dokter | Rp10.000 |
| b. | Jasa Pelayanan Px. Dokter | Rp30.000 |
| c. | Jasa Sarana Px. Darah Rutin | Rp25.000 |
| d. | Jasa Pelayanan Px. Darah Rutin | Rp10.000 |
| e. | Jasa Sarana Px. Rontgen Thorax | Rp40.000 |
| f. | Jasa Pelayanan Px. Rontgen Thorax | Rp40.000 |
| g. | Jasa Sarana Obat dan BHP | Rp50.000 |
| h. | Jasa Pelayanan Obat dan BHP | <u>Rp0,0000 +</u> |

Total Tagihan Rp205.000

Langkah selanjutnya adalah menentukan persentase kontribusi atau *Relative Value Unit* (RVU) dari delapan komponen yang tercantum dalam RUU di atas. RVU setiap komponen dihitung sebagai berikut:

- | | | |
|----|-----------------------------------|---|
| a. | Jasa Sarana Px Dokter | $(Rp10.000/Rp205.000) \times 100 = 4,88\%$ |
| b. | Jasa Pelayanan Px. Dokter | $(Rp30.000/Rp205.000) \times 100 = 14,63\%$ |
| c. | Jasa Sarana Px. Darah Rutin | $(Rp25.000/Rp205.000) \times 100 = 12,20\%$ |
| d. | Jasa Pelayanan Px. Darah Rutin | $(Rp10.000/Rp205.000) \times 100 = 4,88\%$ |
| e. | Jasa Sarana Px. Rontgen Thorax | $(Rp40.000/Rp205.000) \times 100 = 19,51\%$ |
| f. | Jasa Pelayanan Px. Rontgen Thorax | $(Rp40.000/Rp205.000) \times 100 = 19,51\%$ |
| g. | Jasa Sarana Obat dan BHP | $(Rp50.000/Rp205.000) \times 100 = 24,39\%$ |
| h. | Jasa Pelayanan Obat dan BHP | $(Rp0,0000/Rp205.000) \times 100 = 0,00\%$ |
| | Total RVU | 100% |

Setelah pasien pulang, lalu kemudian resume medis pasien ditulis diagnosa dan atau tindakan oleh dokter pada dokumen rekam medis pasien. Petugas bagian coding dan grouping lalu menterjemahkan diagnosa menjadi kode diagnosa berdasarkan *International Code of Diagnose* (ICD) Versi 10 dan menterjemahkan nama tindakan atau prosedur menjadi kode prosedur berdasarkan ICD Versi 9 atau *International Code of Procedure Medicine* (ICPM).

Sebagai contoh, misalkan data berikut (Tarif Paket Standar INA-CBGs 2016 RSUD Kelas B I) benar-benar dihasilkan dari hasil pengkodean dan pengelompokan untuk memasukkan diagnosa dan prosedur pasien, jika ada:

Kode CBGs : Q-5-44-0

Deskripsi : Penyakit Kronis Kecil Lain-lain

Tarif : Rp220.400

Data harga Paket INA-CBGs telah dikumpulkan dan kini dipilah berdasarkan jenis layanan dan tarif rumah sakit.

Maka langkah selanjutnya melakukan RVU masing-masing komponen sebagai berikut:

- | | | |
|----|---------------------------|---------------------------------------|
| a. | Jasa Sarana Px Dokter | $4,88\% \times Rp220.400 = Rp10.751$ |
| b. | Jasa Pelayanan Px. Dokter | $14,63\% \times Rp220.400 = Rp32.254$ |

c.	Jasa Sarana Px. Darah Rutin	12,20% x Rp220.400 = Rp26.878
d.	Jasa Pelayanan Px. Darah Rutin	4,88% x Rp220.400 = Rp10.751
e.	Jasa Sarana Px. Rontgen Thorax	19,51% x Rp220.400 = Rp43.005
f.	Jasa Pelayanan Px. Rontgen Thorax	19,51% x Rp220.400 = Rp43.005
g.	Jasa Sarana Obat dan BHP	24,39% x Rp220.400 = Rp53.756
h.	Jasa Pelayanan Obat dan BHP	0,00% x Rp220.400 = Rp0,0000

Total Tarif CBGs **Rp220.400**

Tahap pertama yang meliputi penghitungan nominal rupiah tiap komponen telah selesai dan kini akan berlanjut ke tahap kedua yaitu pendistribusian.

2.5.1.2 Tahap Kedua: Distribusi

Pada tahap ini, maka harus disepakati terlebih dahulu bahwa kerangka konsep dasar pembagian atau distribusi Jasa Pelayanan secara *Fee for Services* (FFS) per jenis layanan adalah sebagai berikut:



Gambar 2.1: Jasa Pelayanan

Tenaga profesional di bidang kedokteran, seperti dokter, perawat, dokter spesialis, teknisi laboratorium, teknisi rontgen, ahli terapi fisik, apoteker, dan pembantunya merupakan penerima Pelayanan Langsung (JL).

Sedangkan JTL adalah pelayanan yang diberikan kepada seluruh pekerja, baik langsung maupun tidak langsung, termasuk petugas administrasi, IPSRS, laundry, CSSD, personalia, keuangan, dokter, paramedis, dan sebagainya.

Pelayanan diberikan kepada pelaksana secara perseorangan (berdasarkan nama dokter atau nama pelaksana lainnya), sedangkan penugasan asisten dilakukan secara berkelompok. Pengelompokan jasa pelayanan berdasarkan jenis pelayanan bukan berdasarkan unit kerja di rumah sakit. Rumah sakit bersama para dokter jasa pelayanan membuat peraturan mengenai pembagian jasa pelayanan yang disahkan bersama dengan kesejahteraan rumah sakit dan para pelaku jasa layanan didalamnya.

Sebagai contoh, hasil dari rumusan tim untuk jenis layanan pemeriksaan dokter sebagai berikut:



Gambar 2.2: Persentase Jasa Pelayanan

Komponen pelayanan berikut akan dibagi berdasarkan hasil perhitungan INACBG tahap pertama, seperti pada kasus sebelumnya:

- a. Jasa Pelayanan Px. Dokter $14,63\% \times \text{Rp}220.400 = \text{Rp}32.254$
- b. Jasa Pelayanan Px. Darah Rutin $4,88\% \times \text{Rp}220.400 = \text{Rp}10.751$
- c. Jasa Pelayanan Px. Rontgen Thorax $19,51\% \times \text{Rp}220.400 = \text{Rp}43.005$

Berikut ini adalah proporsi distribusi yang ditentukan oleh pemilik rumah sakit, yang kemudian digunakan untuk mengalokasikan setiap layanan dari jumlah tersebut di atas dan memasukkannya ke dalam tabel *Excel*:

Table 2.2: Distribusi Jasa Pemeriksaan Dokter per Penerima

No	Pemeriksaan Dokter	Proporsi Pembagian (%)	Jumlah
1	JL Pelaksana	45	Rp14.514
2	JL Asisten	15	Rp4.838
3	JTL Struktural	10	Rp3.225
4	JTL Pegawai (Bakul)	30	Rp9.676
	Jumlah	100	Rp32.254

Table 2.3: Distribusi Jasa Pemeriksaan Darah Rutin per Penerima

No	Pemeriksaan Darah Rutin	Proporsi Pembagian (%)	Jumlah
1	JL Pelaksana	22,5	Rp2.419
2	JL Asisten	37,5	Rp4.032
3	JTL Struktural	10	Rp1.075
4	JTL Pegawai (Bakul)	30	Rp3.225
	Jumlah	100	Rp10.751

Table 2.4: Distribusi Jasa Pemeriksaan Rontgen Thorax per Penerima

No	Pemeriksaan Rontgen Thorax	Proporsi Pembagian (%)	Jumlah
1	JL Pelaksana	45	Rp19.352
2	JL Asisten	15	Rp6.451
3	JTL Struktural	10	Rp4.301
4	JTL Pegawai (Bakul)	30	Rp12.902
	Jumlah	100	Rp43.005

Demikian proporsi perhitungan jasa sarana dan pelayanan berdasarkan metode konversi.

2.5.1.3 Keunggulan Metode Konversi

- Sebuah. “Keuntungan dinikmati bersama, kerugian ditanggung bersama” adalah pedoman yang sah untuk diikuti.
- Biaya penyerapan sumber daya rumah sakit saat ini harus dikontrol secara aktif, dan penyedia layanan langsung harus berperan proaktif dalam hal ini.
- Dalam hal klaim akhir melebihi tarif rumah sakit, penyedia layanan langsung berhak mendapatkan “Imbalan” (JP) layanan langsung yang lebih tinggi.
- Jika klaim akhir lebih kecil dari tarif rumah sakit, penyedia layanan langsung akan mendapatkan pengurangan hukuman layanan langsung (JP).

- e. Manajemen mungkin mendorong dokter untuk mempraktikkan efektivitas biaya atau pengendalian biaya untuk memberikan layanan yang efisien.

2.5.1.4 Kelemahan Metode Konversi

- a. Layanan akan langsung terpengaruh oleh jumlah faktur dikurangi jumlah klaim.
- b. Jika satu atau dua penyedia layanan langsung melakukan tindakan yang boros, seluruh pekerja akan bertanggung jawab atas “penurunan” layanan yang terjadi (hukuman sistemis).
- c. Konversi negatif menimbulkan kemungkinan tidak terbayarnya obat-obatan dan BHP, sehingga rumah sakit tidak memiliki dana untuk membeli kebutuhan tersebut.
- d. Jika terdapat selisih positif akan membuat alokasi obat dan BHP lebih tinggi dari biaya ril obat dan BHP menimbulkan pertanyaan dan pranksa negative kepada manajemen untuk alokasi sisanya.

2.5.2 Metode Proporsi

Pada prinsipnya pembagian jasa pelayanan dengan metode proporsi adalah merupakan pengembangan dari metode konversi. Hasil Analisa dan review terhadap metode konversi ternyata terdapat beberapa hal yang dirasa kurang adil atau fair diantaranya adalah ketika perhitungan *Relative Value Unit* (RVU) maka seluruh komponen dihitung tanpa membedakan apakah komponen itu merupakan **Jasa Sarana** atautkah **Jasa Pelayanan**.

Secara eksplisit sudah disampaikan bahwa Jasa pelayanan yang akan dibagikan ke masing-masing penerima penghasilan atas jasa yang telah dilakukan sedangkan untuk jasa sarana tidak dibagikan dan akan menjadi pendapatan operasional rumah sakit. Berdasarkan hal itu, maka dapat disimpulkan bahwa seharusnya yang dilakukan perhitungan RVU hanyalah komponen jasa pelayanan sehingga lebih adil dan fair.

Dalam memahami metode proporsi, maka tabel perbedaan antara metode konversi dan proporsi sebagai berikut:

Table 2.5: Perbedaan Metode Konversi dan Proporsi

Metode Konversi	Metode Proporsi
Terdapat 2 langkah yaitu	Terdapat 3 langkah yaitu
1) Tahap pertama menggunakan RVU 2) Tahap kedua melakukan distribusi per penerima	1) Pada langkah pertama, jumlah klaim dibagi dengan persentase layanan yang diberikan. 2) Langkah selanjutnya adalah menggunakan RVU untuk konversi. 3) Distribusi per penerima adalah langkah ketiga.
Untuk mendapatkan RVU untuk setiap komponen, semua item pada faktur, termasuk fasilitas dan layanan, dikonversi.	Agar penulis dapat menentukan RVU layanan untuk setiap komponen, penulis hanya mengonversi komponen yang berkontribusi pada layanan.

2.5.2.1 Tahap Pertama - Proporsi

Tahapan menghitung proporsi jasa pelayanan yang akan ditetapkan dan dijadikan dasar perhitungan secara tetap atau fixed-proportional. Klaim yang diajukan rumah sakit dinilai dan diterima berdasarkan Berita Acara Klaim untuk setiap berkas klaim, dan persentase layanan kemudian dihitung dari sana.

Dua (2) barang (perbekalan) sah yang menjadi acuan persentase jasa terhadap jumlah klaim, yaitu:

- a. Permenkes Nomor 28 Tahun 2014 tentang pedoman pelaksanaan program Jaminan Kesehatan Nasional (JKN)
- b. Permenkes Nomor 12 Tahun 2013 tentang pola tarif BLU rumah sakit di Lingkungan Kementerian Kesehatan
- c. Permenkes Nomor 85 Tahun 2015 tentang Pola Tarif Nasional rumah sakit

Berikut ini yang tercakup dalam Bab V Peraturan Menteri Nomor 28 Tahun 2014 Tentang Pendanaan Fasilitas Kesehatan Rujukan Tingkat Lanjut D2: [7]

- a. Mekanisme pemanfaatan:
 - 1) Rumah sakit / Balai Non BLU/ Non BLUD

Institusi dan rumah sakit yang dikendalikan oleh pemerintah federal atau negara bagian yang belum berstatus BLU atau BLUD dikelola dan didanai menggunakan dana pembayaran klaim.

2) Rumah sakit / Balai BLU / BLUD

Dalam pengelolaan dan pembelanjaan dana pembayaran klaim, fasilitas kesehatan milik pemerintah dan daerah yang berstatus BLU/BLUD mematuhi peraturan yang ditetapkan oleh BLU/BLUD.

3) Rumah sakit swasta/klinik utama

Dana hasil pembayaran klaim RS swasta/klinik utama diserahkan kepada faskes.

- b. Pelayanan kesehatan menyumbang 30–50% dari keseluruhan pendapatan fasilitas kesehatan di FKRTL milik pemerintah. Sementara itu, tanggung jawab Puskesmas adalah menyediakan fasilitas yang diperlukan bagi klinik rujukan swasta tingkat lanjut.

Berikut ini diatur dalam Peraturan Menteri Nomor 12 Tahun 2013 tentang Kesehatan:[12]

- a. Biaya pegawai, biaya operasional, dan biaya investasi merupakan tiga kategori utama pengeluaran rumah sakit yang didanai langsung dari pendapatan BLU rumah sakit.
- b. Pimpinan BLU rumah sakit memutuskan bagaimana pengalokasian dana sesuai dengan maksud ayat 1 dengan persentase sebagai berikut:
- 1) Biaya pegawai paling besar 44%; dan
 - 2) Biaya operasional dan biaya investasi paling kecil 56%

Permenkes Nomor 85 Tahun 2015 mengatur sebagai berikut: [13]

- a. Direktur rumah sakit atau kepala rumah sakit memutuskan bagaimana membelanjakan uang sesuai dengan ayat 1, yang meliputi uang untuk:
- 1) Belanja oleh karyawan; Dan
 - 2) Membeli produk dan jasa serta berinvestasi dalam belanja modal.

- b. Sekurang-kurangnya 40% dari dana yang dialokasikan untuk belanja modal dan belanja barang dan jasa sebagaimana dimaksud pada ayat (2) huruf a dihitung dengan memperhatikan kelangsungan pelayanan.
- c. Fraksi pengeluaran rumah sakit mengacu pada persentase yang disebutkan pada paragraf 2 dan 3.

Berdasarkan tiga ketentuan tersebut maka bisa diambil kesimpulan bahwa besaran jasa pelayanan dari nilai klaim bisa ditetapkan oleh rumah sakit pada rentang nilai 30 – 50% artinya bahwa proporsi jasa pelayanan terhadap nilai klaim adalah tidak boleh kurang dari 30% dan tidak boleh melebihi 50%.

Perbedaan utama pada rumah sakit vertical yang sudah menjalankan remunerasi penuh dengan metode P1, P2, dan P3 maka pembiayaan remunerasi masuk dalam belanja pegawai. Sedangkan apabila tetap menggunakan nomenklatur jasa pelayanan maka pembiayaannya masuk dalam kode rekening belanja barang jasa dan bukan belanja pegawai.

Dalam Kepmenkes Nomor 625 Tahun 2010 maka remunerasi merupakan belanja pegawai. Sehingga saat mengalokasikan anggaran belanja pegawai rumah sakit khususnya milik pemerintah, maka harus menjadi bahan pertimbangan manajemen rumah sakit bahwa jasa pelayanan hanya salah satu komponen belanja pegawai. Artinya manajemen juga harus menetapkan besaran untuk pembiayaan Diklat, Gaji Pegawai non PNS (Pegawai BLUD), Honorarium yang sumber pengajiannya dari pendapatan operasional rumah sakit dan asuransi pegawai non PNS.

Sebagai contoh sebuah rumah sakit telah menetapkan bahwa besaran proporsi jasa pelayanan adalah 40% dari nilai setiap klaim disetujui pembayarannya oleh BPJS Kesehatan berdasarkan Berita Acara (BA) Form pengajuan klaim (FPK).

Berdasarkan contoh perhitungan pada metode Konversi, besarnya proporsi rumah sakit mengambil 40% dari setiap nilai klaim sebagai jasa pelayanan. Sebagai contoh berdasarkan tarif INA-CBGs kode Q-5-44-0 sebesar Rp220.400 x 40% = Rp88.160. Maka besaran jasa pelayanan untuk berkas klaim sebesar Rp88.160.

2.5.2.2 Tahap Kedua - Konversi

Tahap konversi metode proporsi ini sama dengan tahapan konversi pada metode sebelumnya. Perbedaan utama adalah pada metode proporsi hanya jasa pelayanan yang dikonversikan sedangkan jasa sarana tidak ikut dikonversikan namun langsung masuk menjadi pendapatan operasional rumah sakit.

Komponen khusus jasa pelayanan berdasarkan contoh sebelumnya sebagai berikut:

Jasa Pelayanan Px. Dokter	Rp30.000
Jasa Pelayanan Px. Darah Rutin	Rp10.000
Jasa Pelayanan Px. Rontgen Thorax	<u>Rp40.000 +</u>
Total Jasa Pelayanan	Rp80.000

Langkah berikutnya adalah menghitung *relative value unit* dari ke 4 komponen jasa pelayanan tersebut.

Jasa Pelayanan Px. Dokter	$(Rp30.000 / Rp80.000) \times 100 = 37,50\%$
Jasa Pelayanan Px. Darah Rutin	$(Rp10.000 / Rp80.000) \times 100 = 12,50\%$
Jasa Pelayanan Px. Rontgen Thorax	<u>$(Rp40.000 / Rp80.000) \times 100 = 50,00\%$</u>
Total RVU	100%

Sesuai dengan Standar Tarif Paket INA-CBGs Tahun 2016 untuk RSUD Kelas B I, nampaknya data-data berikut ini dihasilkan dari hasil pengkodean dan pengelompokan yang digunakan untuk menyampaikan diagnosa pasien dan prosedurnya, jika ada:

Kode CBGs : Q-5-44-0

Deskripsi : Penyakit Kronis Kecil Lain-lain

Tarif : Rp220.400

Jasa yang boleh diambil sebesar Rp 88.160 dihitung $40\% \times Rp 220.400$, sesuai aturan pada tahap pertama. Pangsa jasa dari klaim adalah 40%. Maka langkah selanjutnya melakukan RVU masing-masing komponen sebagai berikut:

Jasa Pelayanan Px. Dokter	$37,50\% \times Rp88.160 = Rp33.060$
Jasa Pelayanan Px. Darah Rutin	$12,50\% \times Rp88.160 = Rp11.020$
Jasa Pelayanan Px. Rontgen Thorax	<u>$50,00\% \times Rp88.160 = Rp44.080$</u>
Total Porsi Japel dari Klaim	Rp88.160

2.5.2.3 Tahap Ketiga : Distribusi

Sebagai contoh, hasil dari rumusan tim untuk jenis layanan pemeriksaan dokter sebagai berikut:



Gambar 2.3: Persentase Jasa Pelayanan Metode Proporsi

Komponen layanan berikut akan dialokasikan berdasarkan temuan komputasi tahap pertama, dengan menggunakan situasi sebelumnya sebagai contoh:

Jasa Pelayanan Px. Dokter	37,50% x Rp88.160	= Rp33.060
Jasa Pelayanan Px. Darah Rutin	12,50% x Rp88.160	= Rp11.020
Jasa Pelayanan Px. Rontgen Thorax	50,00% x Rp88.160	= Rp44.080
Total Porsi Japel dari Klaim		Rp88.160

Berikut ini adalah proporsi distribusi yang ditentukan oleh pemilik rumah sakit, yang kemudian digunakan untuk mengalokasikan setiap layanan dari jumlah tersebut di atas dan memasukkannya ke dalam tabel Excel:

Table 2.6: Distribusi Jasa Pemeriksaan Dokter per Penerima

No	Pemeriksaan Dokter	Proporsi Pembagian (%)	Jumlah
1	JL Pelaksana	45	Rp14.877
2	JL Asisten	15	Rp4.959
3	JTL Struktural	10	Rp3.306
4	JTL Pegawai (Bakul)	30	Rp9.918
	Jumlah	100	Rp33.060

Table 2.7: Distribusi Jasa Pemeriksaan Darah Rutin per Penerima

No	Pemeriksaan Darah Rutin	Proporsi Pembagian (%)	Jumlah
1	JL Pelaksana	22,5	Rp2.480
2	JL Asisten	37,5	Rp4.133
3	JTL Struktural	10	Rp1.102
4	JTL Pegawai (Bakul)	30	Rp3.306
	Jumlah	100	Rp11.020

Table 2.8: Distribusi Jasa Pemeriksaan Rontgen Thorax per Penerima

No	Pemeriksaan Rontgen Thorax	Proporsi Pembagian (%)	Jumlah
1	JL Pelaksana	45	Rp19.836
2	JL Asisten	15	Rp6.612
3	JTL Struktural	10	Rp4.408
4	JTL Pegawai (Bakul)	30	Rp13.224
	Jumlah	100	Rp44.080

Demikian proporsi perhitungan jasa pelayanan berdasarkan metode proporsi.

2.6 Metode Spiral:

Metode Spiral adalah salah satu pendekatan pengembangan perangkat lunak yang menggabungkan elemen-elemen dari model waterfall dan iteratif. Diperkenalkan oleh Barry Boehm pada tahun 1986, metode ini dirancang untuk mengelola risiko dengan lebih efektif melalui pengulangan (iterasi) yang terstruktur. Pendekatan ini sangat berguna dalam proyek-proyek besar dan kompleks di mana risiko dan perubahan kebutuhan sering terjadi. [14]

Metode Spiral terdiri dari empat fase utama yang berulang-ulang hingga proyek selesai: [15]

1. Perencanaan (*Planning*):

- Penentuan tujuan: Mendefinisikan tujuan, batasan, dan persyaratan sistem. Tujuan ini termasuk kebutuhan pengguna, fitur yang harus disertakan, dan fungsi-fungsi utama aplikasi.
- Penetapan sumber daya: Mengidentifikasi sumber daya yang diperlukan, seperti tenaga kerja, perangkat keras, dan perangkat lunak yang dibutuhkan untuk proyek.

2. Analisis Risiko (*Risk Analysis*):

- Identifikasi risiko: Mengidentifikasi potensi risiko yang dapat mempengaruhi proyek, seperti keterbatasan teknologi, perubahan kebutuhan, dan isu-isu keamanan.
 - Penilaian dan mitigasi risiko: Menilai dampak dan kemungkinan risiko, serta merencanakan langkah-langkah untuk mengurangi atau mengatasi risiko tersebut.
3. Pengembangan dan Validasi (*Engineering & Development*):
- Pembuatan prototipe: Mengembangkan versi awal sistem atau komponen-komponen kunci untuk diuji dan divalidasi. Prototipe ini memungkinkan pengguna dan pengembang untuk mengevaluasi fungsi dan desain awal.
 - Desain dan implementasi: Merancang dan mengimplementasikan sistem berdasarkan umpan balik dari prototipe. Proses ini mencakup pengkodean, pengujian unit, dan integrasi komponen.
4. Evaluasi Pelanggan (*Customer Evaluation*):
- Uji coba dan evaluasi: Melibatkan pengguna dalam pengujian sistem untuk memastikan bahwa sistem memenuhi kebutuhan dan harapan mereka. Umpan balik dari pengguna digunakan untuk memperbaiki dan meningkatkan sistem.
 - Penyempurnaan: Berdasarkan evaluasi dan umpan balik, perbaikan dan penyesuaian dilakukan untuk memenuhi spesifikasi yang diinginkan.

Keempat fase ini diulangi dalam bentuk spiral, dengan setiap putaran (iterasi) bertujuan untuk menyempurnakan sistem lebih lanjut hingga akhirnya mencapai produk yang final dan siap digunakan. Setiap iterasi juga memungkinkan penambahan fitur baru dan peningkatan fungsionalitas sistem .

Metode Spiral sangat cocok untuk pengembangan aplikasi yang kompleks dan dinamis, seperti aplikasi *Bridging BPJS* dan remunerasi dokter, karena memungkinkan pengembangan bertahap dengan evaluasi dan penyesuaian berkelanjutan.

2.6.1 Metode Pengujian dalam Metode Spiral

Dalam konteks Metode Spiral, pengujian adalah bagian integral dari setiap iterasi dan dilakukan secara terus-menerus untuk memastikan kualitas dan keandalan produk perangkat lunak. Metode pengujian yang digunakan meliputi:

1. Pengujian Unit (*Unit Testing*):
 - Dilakukan selama fase pengembangan untuk memeriksa setiap komponen atau unit perangkat lunak secara individual. Tujuannya adalah untuk memastikan bahwa setiap unit berfungsi sesuai dengan spesifikasinya.
2. Pengujian Integrasi (*Integration Testing*):
 - Setelah unit-unit diuji secara individual, mereka digabungkan dan diuji sebagai satu kesatuan untuk memastikan bahwa modul-modul tersebut berfungsi bersama dengan baik.
3. Pengujian Sistem (*System Testing*):
 - Melibatkan pengujian seluruh sistem sebagai entitas lengkap untuk memastikan bahwa semua komponen bekerja bersama secara harmonis dan sistem memenuhi semua persyaratan fungsional.
4. Pengujian Risiko (*Risk-based Testing*):
 - Fokus pada area yang memiliki risiko tertinggi untuk memastikan bahwa potensi masalah teridentifikasi dan diatasi lebih awal dalam proses pengembangan.

Dengan adanya pengujian berkelanjutan dalam setiap iterasi, Metode Spiral memastikan bahwa setiap versi yang dihasilkan memiliki kualitas tinggi dan siap digunakan oleh pengguna. Pendekatan ini memungkinkan deteksi dini dan perbaikan kesalahan, serta penyesuaian sesuai dengan umpan balik pengguna, sehingga menghasilkan produk akhir yang lebih baik. [15]

2.7 Framework Laravel:

Laravel adalah salah satu *framework* pengembangan aplikasi web yang paling populer dan kuat yang dibangun menggunakan bahasa pemrograman PHP.

Framework ini menyediakan berbagai fitur yang memungkinkan pengembang untuk membangun aplikasi web secara cepat dan efisien.

Beberapa fitur utama Laravel meliputi:

1. *Routing* yang Ekspresif: Laravel menyediakan sistem *routing* yang ekspresif dan mudah digunakan, yang memungkinkan pengembang untuk dengan cepat menentukan rute untuk aplikasi web mereka.
2. Pengelolaan Basis Data: *Framework* ini dilengkapi dengan ORM (*Object-Relational Mapping*) yang kuat bernama *Eloquent*, yang menyederhanakan proses interaksi dengan basis data dan memungkinkan pengembang untuk melakukan operasi CRUD (*Create, Read, Update, Delete*) dengan mudah.
3. Otentikasi Pengguna: Laravel menyediakan sistem otentikasi pengguna yang terintegrasi, yang memungkinkan pengembang untuk dengan mudah mengimplementasikan fitur otentikasi dan otorisasi dalam aplikasi mereka.
4. *Blade Templating Engine*: Laravel menggunakan Blade, sebuah mesin templating yang intuitif dan kuat, yang memungkinkan pengembang untuk membuat tampilan yang dinamis dan menarik dengan mudah.
5. Paket Pengelolaan *Dependensi Composer*: Laravel menggunakan *Composer* sebagai manajer dependensi PHP, yang memungkinkan pengembang untuk dengan mudah mengelola dependensi dan paket-paket PHP yang digunakan dalam proyek mereka.

Dengan struktur yang kuat dan berbagai fitur yang disediakan, Laravel sangat cocok untuk pengembangan aplikasi web yang kompleks dan skala besar.

2.8 React.js:

React.js adalah pustaka JavaScript yang digunakan untuk membangun antarmuka pengguna (UI) yang dinamis dan interaktif dalam aplikasi web. Salah satu fitur utama dari React.js adalah penggunaan pendekatan komponen, di mana antarmuka pengguna dibagi menjadi komponen-komponen kecil yang dapat diatur ulang secara independen.

Beberapa keunggulan utama React.js meliputi:

1. Komponen *Reusable*: React.js memungkinkan pengembang untuk membuat komponen-komponen UI yang dapat digunakan kembali, yang mempercepat proses pengembangan dan memungkinkan pengembang untuk mengelola kode dengan lebih efisien.
2. Virtual DOM: React.js menggunakan Virtual DOM, yang merupakan representasi virtual dari DOM aktual, untuk meningkatkan kinerja dan efisiensi dalam memperbarui tampilan aplikasi.
3. Ekosistem yang Kuat: React.js didukung oleh ekosistem yang kuat dan aktif, yang mencakup berbagai alat, pustaka, dan sumber daya pendukung yang memudahkan pengembangan aplikasi web.
4. Kinerja yang Cepat: React.js terkenal karena kinerjanya yang cepat, yang memungkinkan pembuatan aplikasi web yang responsif dan efisien bahkan dengan jumlah data yang besar.

Dengan kombinasi fitur-fitur ini, React.js menjadi salah satu pilihan utama untuk pengembangan antarmuka pengguna yang dinamis dan interaktif dalam aplikasi web modern.

2.9 Daisy UI:

Daisy UI adalah salah satu kerangka kerja desain (UI *framework*) yang dibangun di atas Tailwind CSS. Tailwind CSS adalah sebuah kerangka kerja CSS yang memungkinkan pengembang untuk membangun antarmuka pengguna dengan cepat menggunakan kelas-kelas yang telah ditentukan sebelumnya.

Daisy UI menyediakan berbagai komponen desain yang siap pakai untuk mempercepat pengembangan antarmuka pengguna dalam proyek menggunakan Tailwind CSS. Beberapa komponen yang disediakan oleh Daisy UI meliputi tombol, kartu, formulir, dan banyak lagi.

Keunggulan Daisy UI meliputi:

1. Desain yang Responsif: Komponen-komponen dalam Daisy UI dirancang untuk menjadi responsif, sehingga antarmuka pengguna yang dibangun dengan menggunakan kerangka kerja ini dapat menyesuaikan diri dengan berbagai ukuran layar dan perangkat.

2. **Fleksibilitas:** Daisy UI memungkinkan pengembang untuk menyesuaikan dan menyesuaikan komponen-komponen desain sesuai dengan kebutuhan dan preferensi proyek mereka.
3. **Integrasi Mudah:** Daisy UI dapat dengan mudah diintegrasikan dengan proyek-proyek yang menggunakan Tailwind CSS, sehingga pengembang dapat dengan cepat membangun antarmuka pengguna yang menarik dan responsif tanpa perlu menulis kode CSS tambahan secara manual.

Dengan menggunakan Daisy UI, pengembang dapat mempercepat proses pengembangan dan fokus pada logika aplikasi tanpa harus menghabiskan waktu untuk merancang dan mengkodekan komponen desain dari awal.

2.10 Teknologi Yang Digunakan

Dalam pengembangan aplikasi *bridging* BPJS dan sistem remunerasi dokter, beberapa teknologi, aplikasi, dan tools digunakan untuk menyelesaikan permasalahan penelitian. Berikut adalah penjelasan mengenai teknologi, aplikasi, dan tools yang digunakan:

1. **Sistem Operasi *Windows* 11:**
Digunakan sebagai platform utama untuk menjalankan aplikasi dan tools yang digunakan dalam pengembangan.
2. **Visual Studio Code (VS Code):**
Untuk menulis dan mengedit kode dalam berbagai bahasa pemrograman, ini adalah editor kode sumber yang ringan, kuat, dan sangat dapat dikonfigurasi.
3. **Heidi SQL:**
Sebuah aplikasi client MySQL yang memungkinkan pengguna untuk mengelola dan mengelola database MySQL dengan antarmuka pengguna yang intuitif.
4. **XAMPP:**
Server web Apache, database MySQL, PHP, dan Perl semuanya merupakan bagian dari bundel perangkat lunak pengembangan ini. Ini digunakan untuk mengatur dan mengelola lingkungan lokal untuk pengembangan web.

5. GitHub Desktop:
Aplikasi desktop yang memungkinkan pengguna untuk berinteraksi dengan repositori GitHub tanpa perlu menggunakan terminal atau antarmuka baris perintah.
6. Figma:
Lingkungan tempat desainer dapat berkolaborasi dalam konsep antarmuka pengguna dan pengalaman pengguna (UI/UX), membangun prototipe interaktif, dan mengirimkan konsep tersebut ke tim pengembangan.
7. Google Chrome:
Web browser yang digunakan untuk mengakses berbagai sumber daya online, dokumentasi, serta untuk pengujian aplikasi web.
8. MySQL Workbench:
Aplikasi yang menyediakan alat visualisasi dan pengelolaan database MySQL, digunakan untuk merancang, mengelola, dan menjalankan kueri terhadap database.
9. Microsoft Todo:
Aplikasi manajemen tugas yang membantu pengguna dalam mengorganisir tugas-tugas, membuat daftar kerja, dan mengatur jadwal.
10. Trello:
Aplikasi manajemen proyek yang digunakan untuk mengorganisir tugas-tugas, mengelola alur kerja, dan berkolaborasi dengan tim pengembang.

2.11 Penelitian Terdahulu

Adapun tabel penelitian terdahulu sebagai berikut:

Table 2.9: Tabel Penelitian Terdahulu

No	Nama dan Tahun	Judul	Topik	Subjek	Hasil
1	Indriyati Oktaviano Rahayuningrum, Didik Gunawan Tamtomo, Arief Suryono (2017)	Analisis tarif rumah sakit dibandingkan dengan tarif <i>indonesian case based groups</i> pada pasien rawat inap peserta jaminan	INA-CBG's	Rumah Sakit	Cara pembayarannya menggunakan sistem paket sesuai penyakit pasien

No	Nama dan Tahun	Judul	Topik	Subjek	Hasil
		kesehatan nasional di rumah sakit			
2	Djoni Darmajaya, (2020)	Penyusunan sistem remunerasi dokter pada era disrupsi teknologi informasi di rumah sakit	Remunerasi Dokter	Rumah Sakit	Sistem remunerasi
3	Ingenida Hadning*, Fitriannisa Fathurrohmah, Muhammad Ridwan, Bangunawati Rahajeng, Pina sti Utami, Indriastuti Cahyaningsih, (2020)	<i>Cost Analysis of Indonesia Case Based Groups (INA-CBGs) Tariff for Stroke Patients</i>	INA-CBG's	Rumah Sakit	Mencegah kerugian rumah sakit
4	DANIEL BUDI WIBOWO, (2019)	Pengelolaan dana dan sistem remunerasi dokter pada pasien bpjs - persi	Remunerasi Dokter	Rumah Sakit	Pengelolaan dana dan sistem remunerasi dokter

1. Perbandingan Tarif Rawat Inap Rumah Sakit dengan *Tarif Indonesia Case Based Group* (INA-CBGs) dan Peserta Program Jaminan Kesehatan Nasional

- **Topik:** INA-CBG's, Rumah Sakit

- **Hasil:** Cara pembayarannya menggunakan sistem paket sesuai penyakit pasien

Penelitian ini relevan dengan tugas akhir ini karena membahas tentang tarif rumah sakit yang dibandingkan dengan INA-CBGs. Dalam tugas akhir ini yang berkaitan dengan pembuatan aplikasi *bridging* BPJS perlu memperhitungkan tarif INA-CBGs untuk memastikan alokasi dana yang tepat kepada dokter dan rumah sakit.

2. Mengatasi Disrupsi Teknologi Informasi Rumah Sakit melalui Pengembangan Sistem Kompensasi Dokter

- **Topik:** Remunerasi Dokter, Rumah Sakit

- **Hasil:** Sistem remunerasi

Penelitian ini relevan karena membahas tentang penyusunan sistem remunerasi dokter. Dalam tugas akhir ini juga mencakup aspek sistem remunerasi dokter sebagai bagian dari aplikasi yang dikembangkan.

3. *Cost Analysis of Indonesia Case Based Groups (INA-CBGs) Tariff for Stroke Patients*

- **Topik:** INA-CBG's, Rumah Sakit

- **Hasil:** Mencegah kerugian rumah sakit

Penelitian ini penting karena melibatkan analisis biaya tarif INA-CBGs untuk pasien stroke. Dalam tugas akhir ini, penulis dapat menggunakan hasil analisis ini sebagai bahan referensi untuk memastikan alokasi dana yang efisien dan mencegah kerugian rumah sakit.

4. Pengelolaan Dana dan Sistem Remunerasi dokter pada pasien BPJS - PERSI

- **Topik:** Remunerasi Dokter, Rumah Sakit

- **Hasil:** Pengelolaan dana dan sistem remunerasi dokter

Penelitian ini relevan karena membahas tentang pengelolaan dana dan sistem remunerasi dokter untuk pasien BPJS. Dalam tugas akhir ini, penulis juga perlu mempertimbangkan aspek pengelolaan dana dan sistem remunerasi dokter yang terkait dengan pasien BPJS dalam aplikasi yang penulis kembangkan.

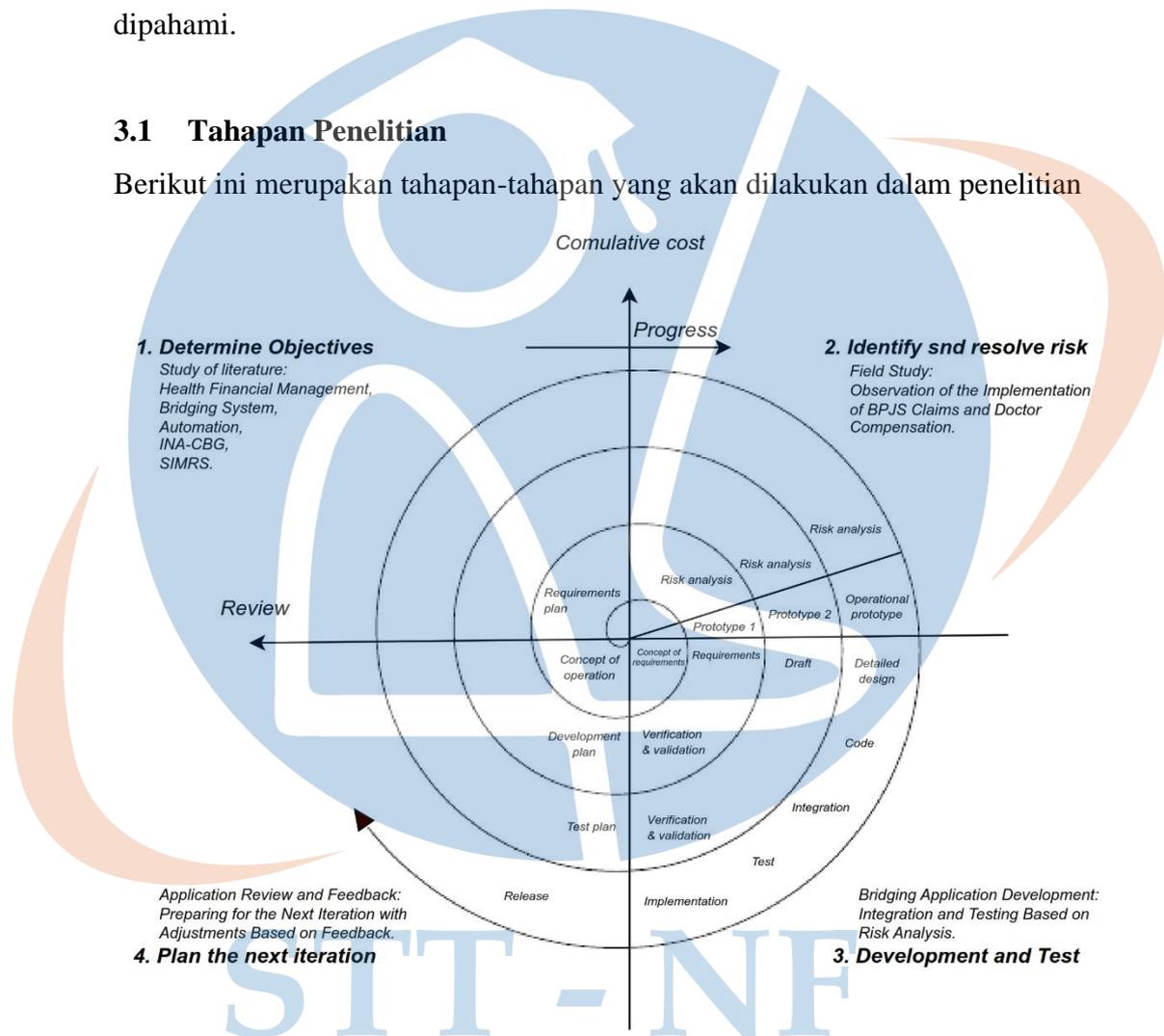
STT - NF

BAB III METODOLOGI PENELITIAN

Pada bagian proses penelitian ini, penulis menguraikan beberapa alur dengan sangat rinci. Penulis memastikan bahwa setiap prosedur siap dijalankan sehingga penulis dapat melakukan penelitian dengan cara yang terorganisir dan mudah dipahami.

3.1 Tahapan Penelitian

Berikut ini merupakan tahapan-tahapan yang akan dilakukan dalam penelitian



Gambar 3.1 Tahapan Penelitian [16]

- a. *Determine Objectives* (Tentukan Tujuan): Dalam konteks penelitian penulis, fase ini mencakup tujuan penelitian umum dan spesifik, seperti studi literatur yang mencakup teori dan konsep terkait manajemen keuangan kesehatan, sistem *bridging*, otomatisasi, INA-CBG, dan SIMRS. Pada tahap ini,

prototyping akan digunakan untuk mengembangkan konsep awal dari aplikasi *bridging* BPJS. Ini melibatkan pembuatan prototipe sederhana yang mencakup antarmuka pengguna dasar dan fungsionalitas utama yang akan dimiliki oleh aplikasi.

- b. *Identify and Resolve Risks* (Identifikasi dan Atasi Risiko): Fase ini menjelaskan bagaimana penulis menggunakan studi lapangan untuk mengidentifikasi risiko dalam implementasi sistem manajemen klaim BPJS dan kompensasi dokter secara manual, serta bagaimana *prototyping* akan digunakan untuk mengatasi risiko ini. Prototyping akan digunakan untuk merancang solusi alternatif yang dapat mengatasi risiko-risiko ini. Ini mungkin melibatkan pengembangan beberapa versi prototipe yang memperlihatkan berbagai pendekatan untuk mengurangi atau menghilangkan risiko-risiko yang telah diidentifikasi.
- c. *Development and Test* (Pengembangan dan Pengujian): Di sini penulis menjabarkan proses pengembangan aplikasi *bridging* BPJS yang penulis buat berdasarkan analisis risiko dan kebutuhan yang diidentifikasi selama studi literatur dan lapangan. Di tahap ini, prototyping akan menjadi fokus utama dalam pengembangan aplikasi *bridging* BPJS. Tim akan menggunakan prototipe untuk membangun iterasi awal dari aplikasi, yang akan terus diperbaiki dan dikembangkan seiring berjalannya waktu.
- d. *Plan the Next Iteration* (Rencanakan Iterasi Selanjutnya): Tahap ini menggambarkan bagaimana penulis akan meninjau aplikasi yang telah dikembangkan, menerima umpan balik, dan merencanakan pengembangan lebih lanjut untuk iterasi selanjutnya dari penelitian dan pengembangan aplikasi. Prototyping akan digunakan untuk merancang versi iteratif baru dari aplikasi, berdasarkan umpan balik yang diterima dari pengguna dan evaluasi hasil dari tahap sebelumnya.

3.2 Rancangan Penelitian

Analisis lingkungan atau studi kasus, definisi kebutuhan, pemrosesan data, prosedur eksperimen, metode penilaian, dan aktivitas lainnya semuanya akan

menjadi bagian dari proses studi pada tahap ini. Bersamaan dengan itu, kerangka penelitian yang menguraikan tema dan metodologi proyek juga tersedia.

3.2.1 Jenis Penelitian

Penelitian ini menggunakan metode pengembangan (R&D) yang difokuskan pada pembangunan atau pengembangan aplikasi baru, yaitu aplikasi *bridging* antara BPJS dan sistem remunerasi dokter. Metode pengembangan ini mencakup tahapan analisis kebutuhan, perancangan sistem, pengembangan aplikasi, implementasi, dan evaluasi. Tujuan utamanya adalah menghasilkan sebuah aplikasi yang dapat memfasilitasi integrasi dan alokasi dana yang efisien antara BPJS dan rumah sakit, serta meningkatkan akuntabilitas dan transparansi dalam sistem pembayaran layanan kesehatan.

Penelitian ini memiliki karakteristik sebagai penelitian pengembangan (R&D), yang bertujuan untuk menciptakan solusi praktis dalam bentuk aplikasi teknologi informasi. Dalam konteks ini, luaran penelitian yang diharapkan adalah pengembangan aplikasi *bridging* BPJS yang dapat memperbaiki proses alokasi dana di rumah sakit, meningkatkan transparansi dalam pengelolaan keuangan, serta memfasilitasi penggunaan dana klaim BPJS secara efisien dan proporsional.

Target Hasil Penelitian:

- a. Pengembangan aplikasi *bridging* yang dapat secara efektif mengintegrasikan data klaim BPJS dengan sistem remunerasi dokter di rumah sakit.
- b. Meluncurkan sistem yang dapat melaporkan secara instan dan akurat bagaimana klaim BPJS disalurkan ke berbagai pihak, termasuk dokter.
- c. Evaluasi aplikasi untuk memastikan keandalan, fungsionalitas, dan kinerja optimal dalam mengelola dan mendistribusikan dana secara proporsional dan transparan.
- d. Menyediakan solusi yang dapat meningkatkan akuntabilitas dan transparansi dalam sistem pembayaran layanan kesehatan, sehingga memperkuat integritas dan kepercayaan stakeholders.

Menurut para pakar:

- a. Menurut Gall, Gall, dan Borg (2003:569), R&D merupakan paradigma pengembangan pendidikan yang digunakan untuk menciptakan barang atau metode baru. Tingkat kemandirian, kualitas, atau standar yang sama dapat dicapai melalui pengujian lapangan, evaluasi, dan peningkatan yang metodis.
- b. Menurut Gay, Mills, dan Airasian (2009: 18), tujuan R&D adalah menciptakan produk inovatif yang dapat bekerja dengan baik di ruang kelas, tidak hanya untuk menguji atau merumuskan ide.
- c. Penelitian dan pengembangan menurut Nana Syaodih Sukmadinata (2015:164) merupakan pendekatan penelitian yang ampuh untuk meningkatkan praktik. Penelitian ini merupakan proses untuk membuat item baru atau menyempurnakan item yang sudah ada.

3.2.2 Metode Analisis Data

Penelitian ini akan menggunakan pendekatan kualitatif, dengan fokus pada pemahaman mendalam terhadap implementasi klaim BPJS dan sistem remunerasi dokter di rumah sakit.

Penelitian ini akan menggunakan dua pendekatan utama:

- a. **Studi Literatur:** Studi literatur akan dilakukan untuk menganalisis teori-teori terkait manajemen keuangan kesehatan dan sistem klaim BPJS. Landasan konseptual akan diperoleh melalui review literatur yang meliputi jurnal ilmiah, buku, dan dokumen-dokumen resmi terkait. Selain itu, informasi yang relevan juga akan diperoleh dari seminar-seminar terkait industri kesehatan, termasuk pengalaman dan keluhan yang disampaikan oleh rumah sakit dalam seminar yang dihadiri peneliti. Seminar ini telah memberikan wawasan awal tentang masalah transparansi keuangan, khususnya dalam alokasi dana dari pasien BPJS. Keluhan-keluhan ini menjadi dasar untuk memahami isu-isu yang dihadapi oleh rumah sakit secara langsung.
- b. **Studi Kasus:** Penelitian akan melibatkan studi kasus pada beberapa rumah sakit. Pemilihan rumah sakit akan dilakukan dengan cermat untuk memperoleh representasi yang baik dari berbagai konteks dan skala. Data

akan diperoleh melalui wawancara mendalam dengan pihak terkait di rumah sakit, seperti manajer keuangan, staf administrasi, dan dokter. Selain itu, observasi langsung akan dilakukan untuk memahami proses implementasi klaim BPJS dan sistem remunerasi dokter. Data kualitatif yang terkumpul akan dianalisis secara tematis untuk mengidentifikasi tantangan yang dihadapi oleh rumah sakit dalam mengelola keuangan kesehatan, termasuk dalam hal transparansi alokasi dana dari pasien BPJS.

Dengan menggunakan pendekatan kualitatif ini, diharapkan penelitian dapat memberikan pemahaman yang mendalam tentang permasalahan yang dihadapi oleh rumah sakit terkait dengan klaim BPJS dan sistem remunerasi dokter, serta memberikan rekomendasi yang relevan untuk meningkatkan transparansi dan efisiensi dalam alokasi dana kesehatan.

3.2.3 Metode Pengumpulan Data

Berikut ini adalah rincian prosedur pengumpulan data yang akan digunakan dalam penelitian ini:

- a. Wawancara:
 - 1) Tujuan: Mendapatkan wawasan langsung dari pihak terkait, seperti petugas administrasi rumah sakit, dokter, dan staf BPJS.
 - 2) Sumber Data: Informasi dan pandangan dari responden yang diwawancarai.
 - 3) Bentuk Data: Catatan transkripsi wawancara, mengidentifikasi pandangan, pengalaman, dan masukan terkait klaim BPJS dan remunerasi dokter.
- b. Observasi:
 - 1) Tujuan: Mengamati langsung proses klaim BPJS dan sistem remunerasi dokter di rumah sakit terpilih.
 - 2) Sumber Data: Pengamatan langsung terhadap proses kerja dan interaksi antar *stakeholder*.
 - 3) Bentuk Data: Catatan observasi yang mencakup detail proses, kendala, dan interaksi yang diamati.

c. Analisis Dokumen:

- 1) Tujuan: Mengumpulkan data historis terkait implementasi klaim BPJS dari dokumen dan catatan rumah sakit.
- 2) Sumber Data: Dokumen resmi, kebijakan rumah sakit, catatan klaim BPJS, dan data historis lainnya.
- 3) Bentuk Data: Data terstruktur dari dokumen-dokumen tersebut, termasuk kebijakan, peraturan, dan rekam jejak pelaksanaan klaim BPJS.

Dengan menggabungkan berbagai metode pengumpulan data ini, diharapkan penelitian dapat memperoleh informasi yang komprehensif dan mendalam terkait proses klaim BPJS dan sistem remunerasi dokter di rumah sakit.

3.2.4 Metode Pengujian

Metode pengujian pada tugas akhir ini direncanakan untuk memastikan keandalan, fungsionalitas, dan kinerja optimal dari aplikasi *bridging* BPJS dan sistem remunerasi dokter. Langkah-langkah pengujian meliputi:

a. *Black Box Testing*:

- 1) Menyusun skenario pengujian yang mencakup berbagai kasus penggunaan untuk setiap fitur aplikasi.
- 2) Mengidentifikasi input yang relevan untuk setiap skenario pengujian.
- 3) Menjalankan skenario pengujian dan mencatat output yang dihasilkan.
- 4) Membandingkan output yang diharapkan dengan output yang diperoleh untuk menentukan apakah fungsi aplikasi berjalan sesuai yang diharapkan.

b. *Load Testing*:

- 1) Menentukan skenario pengujian yang mencerminkan situasi penggunaan puncak.
- 2) Memantau kinerja aplikasi selama pengujian untuk mengidentifikasi batasan kinerja dan potensi titik lemah.
- 3) Menganalisis hasil pengujian untuk menentukan apakah aplikasi tetap responsif dan dapat menangani beban kerja yang tinggi.

3.2.5 Metode Implementasi dan Evaluasi

Metode implementasi dan evaluasi dalam tugas akhir ini bertujuan untuk menerapkan aplikasi *bridging* BPJS dan sistem remunerasi dokter ke dalam lingkungan rumah sakit. Langkah-langkah implementasi dan evaluasi dirinci sebagai berikut:

a. Persiapan Implementasi:

1) Identifikasi Ruang Lingkup Implementasi:

Tentukan area atau departemen rumah sakit yang akan terlibat dalam implementasi, seperti unit klaim BPJS dan bagian administrasi.

2) Koordinasi dengan Pihak Terkait:

Lakukan koordinasi dengan pihak terkait, termasuk petugas administrasi, dokter, dan staf BPJS, untuk memastikan dukungan dan partisipasi selama proses implementasi.

b. Pelaksanaan Implementasi:

1) Instalasi Perangkat Lunak atau Implementasi Alat:

Lakukan instalasi aplikasi *bridging* dan sistem remunerasi dokter di lingkungan rumah sakit.

2) Pelatihan Pengguna:

Selenggarakan pelatihan untuk pengguna yang terlibat agar dapat menggunakan sistem dengan efektif.

3) Pengalihan Data:

Untuk transisi bebas masalah, mungkin diperlukan pemindahan data dari sistem sebelumnya ke sistem baru.

4) Uji Coba Fungsionalitas:

Lakukan uji coba fungsionalitas sistem secara menyeluruh untuk memastikan kinerja yang optimal sebelum diimplementasikan sepenuhnya.

c. Evaluasi Proses Implementasi:

1) Pemantauan dan Evaluasi Keseluruhan:

Lakukan pemantauan dan evaluasi menyeluruh terhadap seluruh proses implementasi.

2) Identifikasi dan Penanganan Masalah:

Identifikasi serta tangani masalah atau hambatan yang muncul selama proses implementasi.

3) Pemantauan Tingkat Adopsi:

Monitor tingkat adopsi oleh pengguna dan identifikasi kemungkinan perubahan yang diperlukan.

3.2.6 Lingkungan Pengembangan

a. Lokasi Penelitian

Penelitian dilakukan secara online dikediaman penulis yang beralamat di GARASI ILMU, Jl. Al Islamiyah No.36, RT.6/RW.1, Sukabumi Sel., Kec. Kb. Jeruk, Kota Jakarta Barat, Daerah Khusus Ibukota Jakarta 11560.

b. Framework dan Tools Pengembangan:

- 1) Framework Laravel (versi terbaru): Digunakan untuk pengembangan aplikasi web.
- 2) MySQL: Sebagai sistem manajemen basis data untuk menyimpan data klaim BPJS dan data remunerasi dokter.

c. Alat dan Bahan:

- 1) Laptop Dell Inspiron 3501
 - RAM: 16GB
 - 11th Gen Intel(R) Core(TM) i3-1115G4 @ 3.00GHz (4 CPUs), ~3.0GHz

Windows 11 Home 64-bit (10.0, Build 22H2)

- 2) HeidiSQL: Untuk mengatur database mysql.
- 3) Microsoft Excel 2019: Untuk sebagai mendata atau testing fitur import data keuangan dan klaim BPJS.
- 4) WhatsApp: Sebuah aplikasi layanan pesan untuk diskusi terkait data pendapatan dan klaim BPJS.
- 5) Github Desktop: Aplikasi yang digunakan untuk menyimpan, melacak, dan sebagai tempat kolaborasi antar developer dalam suatu proyek perangkat lunak.

- 6) Visual Studio Code: Sebagai aplikasi teks editor untuk membangun program.
- 7) XAMPP: Sebuah lokal server, digunakan untuk me run aplikasi pada masa pengembangan.
- 8) ClickUp: Aplikasi untuk mengecek sejauh mana proyek ini dikerjakan.

3.2.7 Metode Pengembangan Perangkat Lunak

Untuk proyek ini, penulis mengikuti langkah-langkah metodologi Spiral untuk pengembangan perangkat lunak:

a. Identifikasi:

- 1) Komunikasi antara penulis (sebagai pengembang) dan CEO perusahaan penulis (atau pemangku kepentingan lainnya) untuk mengidentifikasi tujuan, alternatif, dan batasan proyek.
- 2) Memahami dengan baik persyaratan sistem dan merencanakan proyek dengan cermat.

Pada tahap identifikasi, penulis berkomunikasi secara intensif dengan CEO perusahaan penulis untuk memahami tujuan akhir proyek, mempertimbangkan alternatif yang mungkin, dan menetapkan batasan yang jelas. Selain itu, penulis juga melakukan analisis menyeluruh terhadap persyaratan sistem dan merencanakan strategi pengembangan yang tepat.

b. Desain:

- 1) Generasi dan evaluasi solusi alternatif serta identifikasi serta resolusi risiko yang mungkin terjadi.
- 2) Mungkin melibatkan pembuatan prototipe, simulasi, atau model untuk menguji ide-ide desain.

Dalam fase desain, penulis menghasilkan berbagai solusi alternatif yang mungkin, kemudian mengevaluasi setiap solusi dengan cermat. Penulis juga mengidentifikasi dan memecahkan risiko yang mungkin muncul selama pengembangan. Proses ini dapat melibatkan pembuatan prototipe atau simulasi untuk memvalidasi ide-ide desain penulis.

c. Konstruksi atau Build:

- 1) Implementasi desain yang dipilih dengan membangun sistem berdasarkan rancangan yang telah dibuat.
- 2) Mengonfirmasi fungsionalitas sistem yang diinginkan melalui pengujian.

Pada tahap konstruksi, penulis mulai mengimplementasikan desain yang telah dipilih, membangun sistem berdasarkan rencana yang telah penulis susun. Sebagai langkah tambahan, penulis sering menguji sistem untuk memastikan sistem memenuhi semua persyaratan.

d. Evaluasi dan Risiko Analisis:

- 1) Evaluasi hasil dari fase konstruksi dan melakukan analisis risiko untuk tahap berikutnya dari proyek.
- 2) Melibatkan review oleh CEO perusahaan penulis dan pengguna akhir untuk memastikan bahwa sistem memenuhi kebutuhan rumah sakit.

Terakhir, dalam fase evaluasi dan analisis risiko, penulis meninjau hasil dari fase konstruksi secara menyeluruh. Penulis juga melakukan analisis risiko untuk mengidentifikasi dan mengatasi masalah yang mungkin muncul di masa mendatang. Selain itu, penulis melibatkan CEO perusahaan penulis dalam proses evaluasi untuk memastikan bahwa sistem penulis memenuhi kebutuhan dan harapan rumah sakit.

STT - NF

BAB IV

IMPLEMENTASI DAN EVALUASI

Dalam bab ini, akan dijelaskan secara rinci tentang implementasi dan evaluasi aplikasi yang telah dikembangkan. Bab ini bertujuan untuk memberikan gambaran mendalam tentang langkah-langkah yang diambil dalam pengembangan aplikasi serta hasil dari pengujian aplikasi tersebut. Implementasi mencakup proses penerapan desain aplikasi dalam lingkungan nyata, sedangkan evaluasi melibatkan penilaian terhadap aplikasi yang telah diimplementasikan untuk memastikan bahwa semua fungsi berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan.

4.1 Identifikasi Kebutuhan Pengguna

Sebelum memulai proses pengembangan aplikasi, penulis melakukan serangkaian observasi dan wawancara dengan stakeholder utama di Rumah Sakit XYZ. Tujuan dari kegiatan ini adalah untuk mengidentifikasi kebutuhan spesifik yang harus diatasi oleh aplikasi. Dari hasil observasi dan wawancara tersebut, penulis mendapatkan *insight* penting tentang proses operasional dan kebutuhan informasi yang ada. Berdasarkan data yang dikumpulkan, penulis mengidentifikasi bahwa pengguna membutuhkan beberapa fitur kunci untuk meningkatkan efisiensi dan transparansi dalam pengelolaan keuangan dan remunerasi dokter, yang meliputi:

- a. Fitur *Bridging* BPJS: Untuk mengintegrasikan data klaim BPJS dengan sistem manajemen keuangan rumah sakit secara otomatis.
- b. Kalkulator Remunerasi Dokter: Untuk menghitung secara otomatis pembagian pendapatan berdasarkan kontribusi dokter dan staff medis lainnya dalam penanganan kasus.
- c. Sistem Pelaporan: Untuk menghasilkan laporan keuangan dan remunerasi secara periodik yang dapat diakses oleh manajemen rumah sakit.

Dengan memahami kebutuhan ini, penulis dapat merancang dan menerapkan solusi yang tidak hanya teknis memadai tetapi juga *align* dengan kebutuhan operasional Rumah Sakit XYZ.

4.2 Rancangan Penelitian

Sebelum memasuki detail teknis dari implementasi, penting untuk memahami struktur keseluruhan dan metodologi yang digunakan dalam rancangan penelitian ini. Rancangan penelitian ini dirancang untuk memastikan bahwa aplikasi yang dikembangkan dapat memenuhi semua kebutuhan pengguna dan sekaligus mengatasi permasalahan yang ada dalam sistem manajemen klaim BPJS dan remunerasi dokter. Ini mencakup persiapan awal, pengembangan sistem, hingga evaluasi akhir.

4.2.1 Cara Kerja

- a. Fase Inisiasi: Penulis melakukan identifikasi masalah dan pengumpulan kebutuhan dari Rumah Sakit XYZ untuk integrasi klaim BPJS dan remunerasi dokter.
- b. Fase Perencanaan: Menetapkan *roadmap* aplikasi dengan milestones yang jelas, termasuk tahapan pengembangan spiral dan iterasi yang diharapkan.
- c. Fase Desain: Pengembangan arsitektur sistem dan *prototyping* antarmuka berdasarkan kebutuhan yang teridentifikasi.
- d. Fase Implementasi: Coding dan integrasi komponen aplikasi menggunakan Laravel sebagai *backend* dan React.js sebagai *frontend*.
- e. Fase Pengujian: Melakukan serangkaian test untuk memastikan aplikasi berfungsi sesuai kebutuhan dan aman dari *bug*.

4.2.2 Arsitektur Sistem

- a. *Backend*: Laravel digunakan untuk mengelola database dan logika bisnis terkait pengolahan data remunerasi dan klaim BPJS.
- b. *Frontend*: React.js digunakan untuk membangun antarmuka pengguna yang interaktif dan responsif.
- c. Database: MySQL untuk penyimpanan data klaim, remunerasi, dan informasi pengguna.
- d. Keamanan: Implementasi pengamanan data melalui enkripsi dan autentikasi pengguna.

4.2.3 Tahapan iterasi

Berikut adalah tahapan-tahapan iterasi yang penulis lakukan.

Table 4.1: Tabel Tahapan Iterasi

Iterasi	Tahapan	Deskripsi
Iterasi 1	Perencanaan	Menentukan kebutuhan pengguna terkait fitur konversi pendapatan rumah sakit dan klaim BPJS.
Iterasi 1	Analisis Risiko	Mengidentifikasi risiko kesalahan dalam konversi data pendapatan dan klaim BPJS.
Iterasi 1	Pengembangan	Membuat modul dasar untuk fitur Income Conversion dan pengelolaan klaim BPJS.
Iterasi 1	Evaluasi Pengguna	Pengujian awal oleh pengguna untuk validasi hasil konversi pendapatan.
Iterasi 2	Perencanaan	Rencana pengembangan fitur untuk melihat klaim BPJS yang belum terbayar.
Iterasi 2	Analisis Risiko	Risiko terkait ketepatan data klaim yang belum terbayar.
Iterasi 2	Pengembangan	Implementasi fitur untuk menampilkan dan memfilter klaim BPJS yang belum terbayar.
Iterasi 2	Evaluasi Pengguna	Uji fitur klaim BPJS oleh pengguna rumah sakit.
Iterasi 3	Perencanaan	Merencanakan fitur pelaporan keuangan yang mendukung keputusan manajemen.
Iterasi 3	Analisis Risiko	Risiko ketidakakuratan dalam laporan keuangan akibat kesalahan konversi data.
Iterasi 3	Pengembangan	Pengembangan modul laporan keuangan untuk menampilkan total pendapatan dan pengeluaran.
Iterasi 3	Evaluasi Pengguna	Evaluasi laporan keuangan oleh tim keuangan rumah sakit.
Iterasi 4	Perencanaan	Perencanaan pengembangan fitur pengelolaan data dokter dan KPI (Key Performance Indicators).
Iterasi 4	Analisis Risiko	Risiko dalam integrasi data KPI dengan pendapatan dokter.
Iterasi 4	Pengembangan	Pengembangan fitur pengelolaan data dokter dan pengaturan KPI dokter.
Iterasi 4	Evaluasi Pengguna	Evaluasi oleh dokter dan manajemen rumah sakit terhadap fitur KPI.
Iterasi 5	Perencanaan	Rencana untuk menyempurnakan fitur Income Conversion dan laporan keuangan.

Iterasi	Tahapan	Deskripsi
Iterasi 5	Analisis Risiko	Analisis risiko terkait pemrosesan data besar pada fitur Income Conversion.
Iterasi 5	Pengembangan	Penyempurnaan fitur Income Conversion dan optimasi laporan keuangan.
Iterasi 5	Evaluasi Pengguna	Uji coba penyempurnaan fitur oleh pengguna dan manajemen keuangan.
Iterasi 6	Perencanaan	Perencanaan fitur untuk memonitor pasien BPJS dan non-BPJS.
Iterasi 6	Analisis Risiko	Risiko ketidakakuratan dalam memisahkan data pasien BPJS dan non-BPJS.
Iterasi 6	Pengembangan	Implementasi fitur yang menampilkan data pasien BPJS dan non-BPJS.
Iterasi 6	Evaluasi Pengguna	Pengujian fitur oleh petugas administrasi rumah sakit.
Iterasi 7	Perencanaan	Rencana untuk mengembangkan fitur jurnal pendapatan yang terintegrasi dengan laporan keuangan.
Iterasi 7	Analisis Risiko	Risiko dalam pengelolaan jurnal yang akurat berdasarkan data pendapatan.
Iterasi 7	Pengembangan	Pengembangan fitur jurnal pendapatan dan integrasi dengan sistem laporan keuangan.
Iterasi 7	Evaluasi Pengguna	Evaluasi oleh tim akuntansi terhadap akurasi dan kegunaan jurnal.
Iterasi 8	Perencanaan	Rencana untuk memperluas fitur pengelolaan tarif rumah sakit dan dokter.
Iterasi 8	Analisis Risiko	Risiko ketidaksesuaian tarif antara rumah sakit dan dokter.
Iterasi 8	Pengembangan	Implementasi fitur pengelolaan tarif rumah sakit dan dokter.
Iterasi 8	Evaluasi Pengguna	Uji fitur tarif oleh manajemen rumah sakit dan dokter.
Iterasi 9	Perencanaan	Rencana pengembangan fitur untuk mengelola vendor obat rumah sakit.
Iterasi 9	Analisis Risiko	Risiko ketidaksesuaian dalam pengelolaan vendor obat dan pengadaan.
Iterasi 9	Pengembangan	Pengembangan fitur untuk mengelola dan memverifikasi vendor obat.
Iterasi 9	Evaluasi Pengguna	Evaluasi fitur vendor obat oleh tim pengadaan rumah sakit.
Iterasi 10	Perencanaan	Rencana untuk meningkatkan keamanan data pasien dan pendapatan rumah sakit.

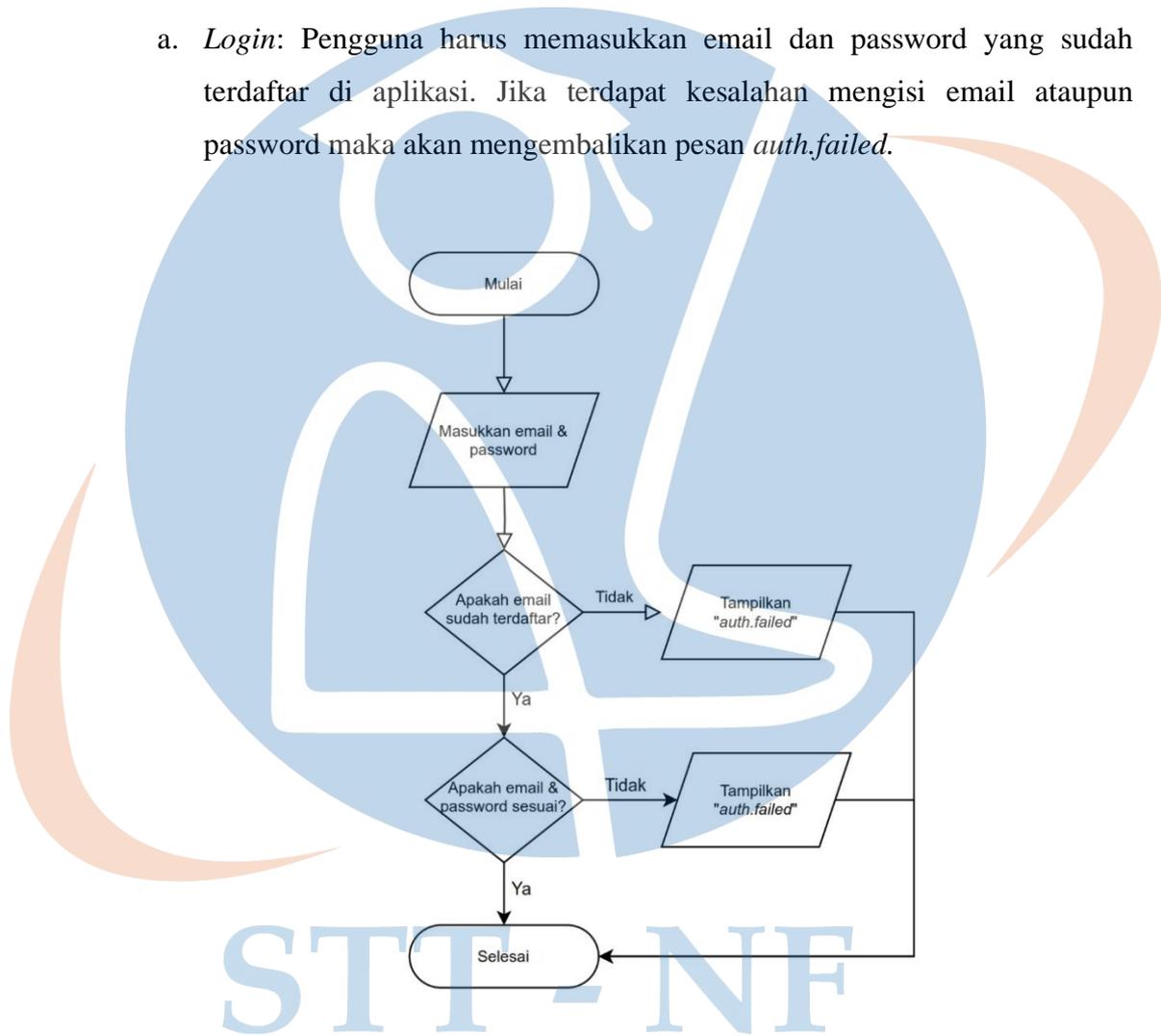
Iterasi	Tahapan	Deskripsi
Iterasi 10	Analisis Risiko	Risiko kebocoran data dan integritas sistem.
Iterasi 10	Pengembangan	Implementasi peningkatan keamanan pada fitur-fitur utama aplikasi.
Iterasi 10	Evaluasi Pengguna	Uji coba keamanan oleh tim IT rumah sakit.
Iterasi 11	Perencanaan	Perencanaan untuk peningkatan kinerja sistem secara keseluruhan.
Iterasi 11	Analisis Risiko	Risiko penurunan kinerja seiring bertambahnya pengguna dan data.
Iterasi 11	Pengembangan	Optimasi kinerja sistem dan database.
Iterasi 11	Evaluasi Pengguna	Pengujian performa sistem oleh pengguna akhir.
Iterasi 12	Perencanaan	Rencana pengembangan fitur tambahan berdasarkan umpan balik pengguna.
Iterasi 12	Analisis Risiko	Risiko kompleksitas fitur yang dapat mempengaruhi usability.
Iterasi 12	Pengembangan	Penambahan fitur berdasarkan umpan balik seperti optimasi tampilan dan navigasi.
Iterasi 12	Evaluasi Pengguna	Evaluasi usability oleh pengguna rumah sakit.
Iterasi 13	Perencanaan	Rencana untuk integrasi aplikasi dengan sistem eksternal yang digunakan rumah sakit.
Iterasi 13	Analisis Risiko	Risiko kompatibilitas antara aplikasi dan sistem eksternal.
Iterasi 13	Pengembangan	Pengembangan API untuk integrasi dengan sistem eksternal.
Iterasi 13	Evaluasi Pengguna	Pengujian integrasi sistem oleh tim IT.
Iterasi 14	Perencanaan	Perencanaan peningkatan fitur dashboard untuk visualisasi data lebih baik.
Iterasi 14	Analisis Risiko	Risiko informasi yang salah pada visualisasi data.
Iterasi 14	Pengembangan	Pengembangan fitur dashboard dengan grafis dan laporan interaktif.
Iterasi 14	Evaluasi Pengguna	Evaluasi fitur dashboard oleh manajemen rumah sakit.
Iterasi 15	Perencanaan	Rencana untuk mengimplementasikan pengelolaan akses dan otorisasi pengguna.

Iterasi	Tahapan	Deskripsi
Iterasi 15	Analisis Risiko	Risiko akses yang tidak sah dan kesalahan otorisasi.
Iterasi 15	Pengembangan	Implementasi sistem pengelolaan akses dan otorisasi pengguna.
Iterasi 15	Evaluasi Pengguna	Uji sistem akses dan otorisasi oleh pengguna terpilih.
Iterasi 16	Perencanaan	Rencana untuk pengembangan fitur pelacakan dan pelaporan aktivitas pengguna.
Iterasi 16	Analisis Risiko	Risiko terkait privasi pengguna dan penyimpanan data.
Iterasi 16	Pengembangan	Pengembangan fitur pelacakan aktivitas dan log pengguna.
Iterasi 16	Evaluasi Pengguna	Evaluasi fitur oleh tim manajemen dan keamanan.
Iterasi 17	Perencanaan	Rencana untuk mengembangkan modul pelatihan pengguna.
Iterasi 17	Analisis Risiko	Risiko rendahnya adopsi sistem oleh pengguna.
Iterasi 17	Pengembangan	Pengembangan modul tutorial dan bantuan untuk pengguna baru.
Iterasi 17	Evaluasi Pengguna	Uji coba modul pelatihan oleh pengguna baru.
Iterasi 18	Perencanaan	Rencana untuk peningkatan dokumentasi teknis dan user manual.
Iterasi 18	Analisis Risiko	Risiko dokumentasi yang tidak up-to-date atau tidak jelas.
Iterasi 18	Pengembangan	Penyusunan dan penyempurnaan dokumentasi teknis serta user manual.
Iterasi 18	Evaluasi Pengguna	Review dan feedback dari pengguna terkait dokumentasi.
Iterasi 19	Perencanaan	Rencana untuk pemeliharaan sistem dan penanganan bug secara berkelanjutan.
Iterasi 19	Analisis Risiko	Risiko downtime dan masalah tak terduga lainnya.
Iterasi 19	Pengembangan	Implementasi sistem monitoring dan prosedur pemeliharaan.
Iterasi 19	Evaluasi Pengguna	Evaluasi stabilitas sistem oleh pengguna akhir.
Iterasi 20	Perencanaan	Rencana untuk finalisasi dan peluncuran sistem secara penuh.
Iterasi 20	Analisis Risiko	Risiko pada penerapan penuh seperti resistensi perubahan oleh pengguna.

Iterasi	Tahapan	Deskripsi
Iterasi 20	Pengembangan	Penyiapan dan pelaksanaan peluncuran sistem di seluruh organisasi.
Iterasi 20	Evaluasi Pengguna	Evaluasi akhir dan feedback dari pengguna untuk iterasi berikutnya (jika ada).

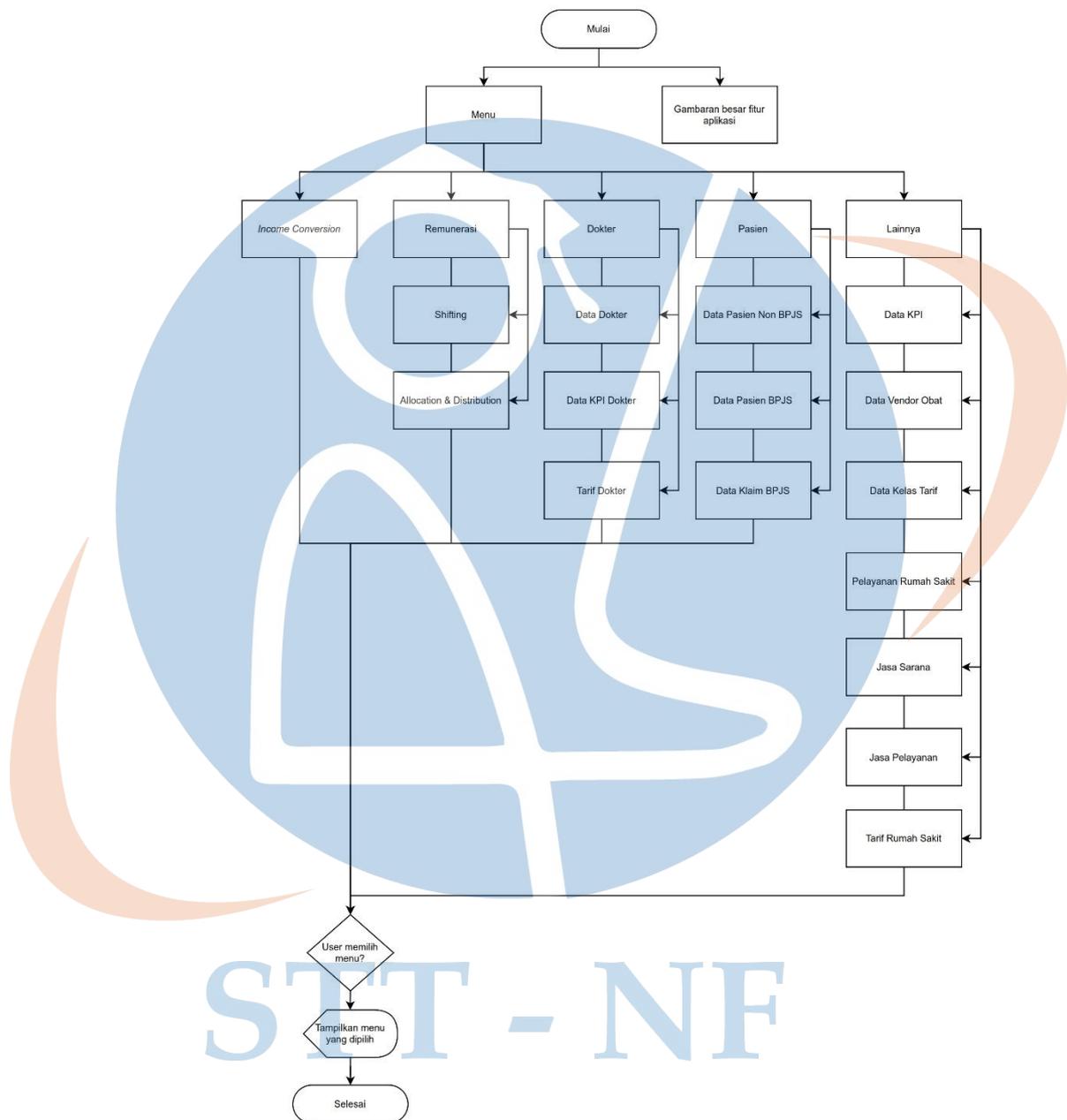
4.2.4 Flowchart

- a. *Login*: Pengguna harus memasukkan email dan password yang sudah terdaftar di aplikasi. Jika terdapat kesalahan mengisi email ataupun password maka akan mengembalikan pesan *auth.failed*.



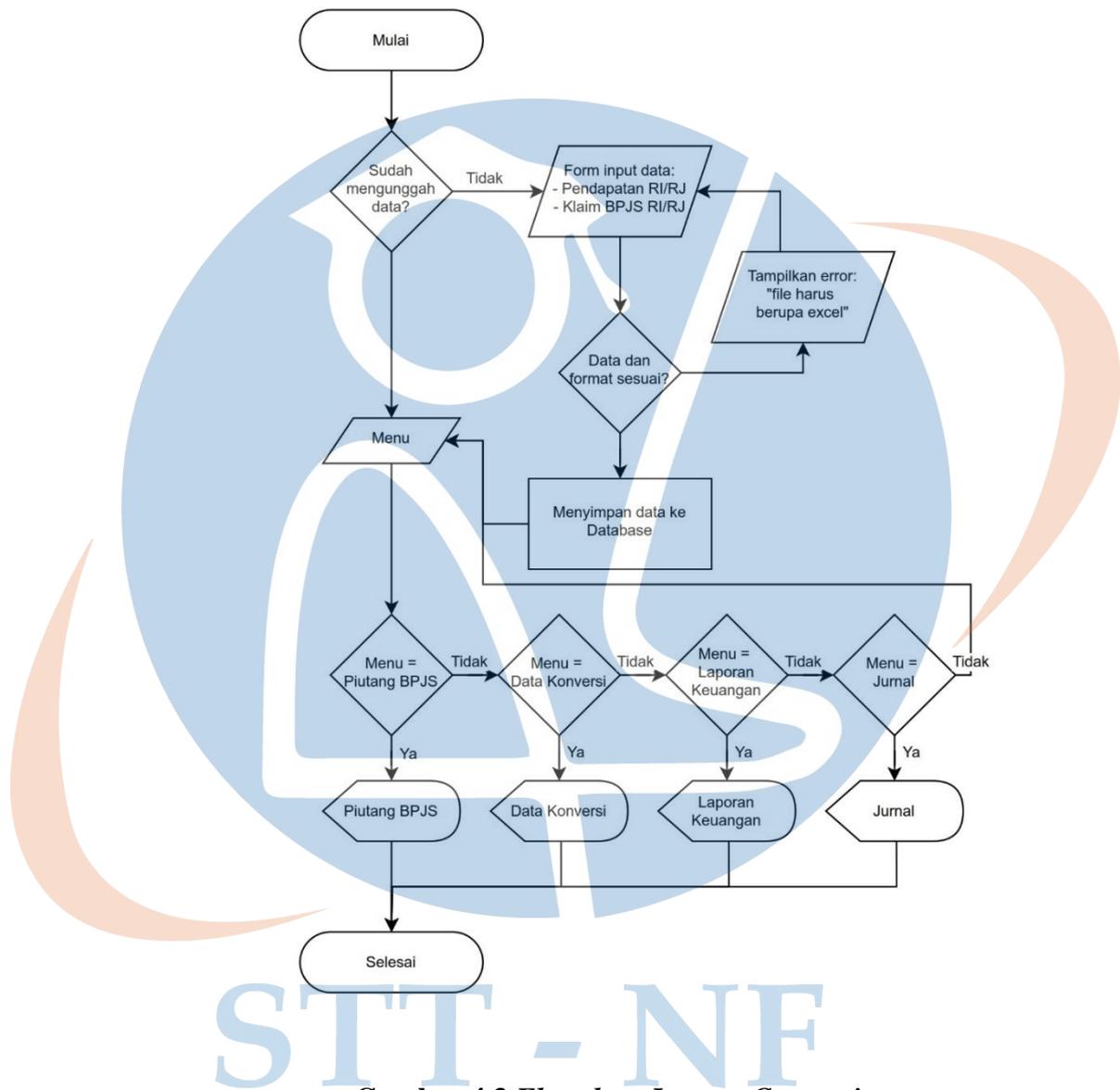
Gambar 4.1 Flowchart Login

- b. Dashboard: Pengguna bisa memilih beberapa menu seperti *income conversion*, remunerasi, dokter dan menu lainnya. Pengguna juga bisa melihat beberapa poin penting yang ada di aplikasi ini.



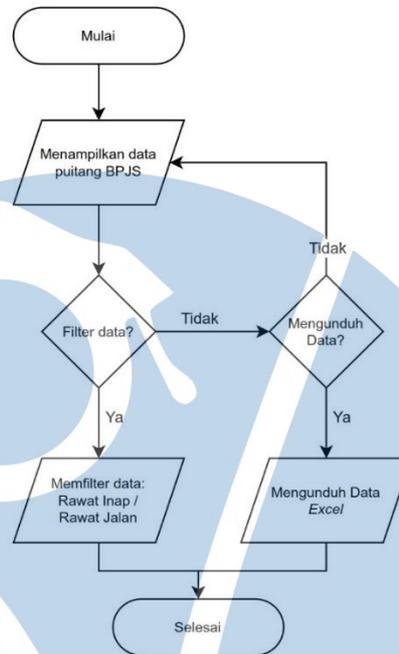
Gambar 4.2 Flowchart Dashboard

- c. *Income conversion*: Pengguna bisa melihat data yang sudah dikonversi setelah data pendapatan dan klaim BPJSnya sudah diunggah, jika belum maka pengguna harus mengunggahnya terlebih dahulu dalam bentuk *excel*.



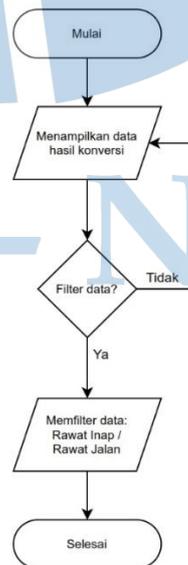
Gambar 4.3 Flowchart Income Conversion

- d. Piutang BPJS/klaim BPJS yang belum terbayar: Pengguna bisa melihat data-data pendapatan dari pasien BPJS baik pasien rawat inap atau rawat jalan yang dimana statusnya masih piutang.



Gambar 4.4 Flowchart Piutang BPJS

- e. Data hasil konversi: Pengguna dapat melihat semua data pendapatan pasien BPJS yang telah dikonversi. Pengguna juga bisa melakukan filter data berdasarkan pasien rawat inap atau rawat jalan.



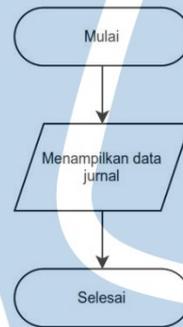
Gambar 4.5 Flowchart Data Hasil Konversi

- f. Laporan keuangan: Pengguna bisa melihat laporan keuangan dari data yang sudah dikonversi.



Gambar 4.6 Flowchart Laporan Keuangan

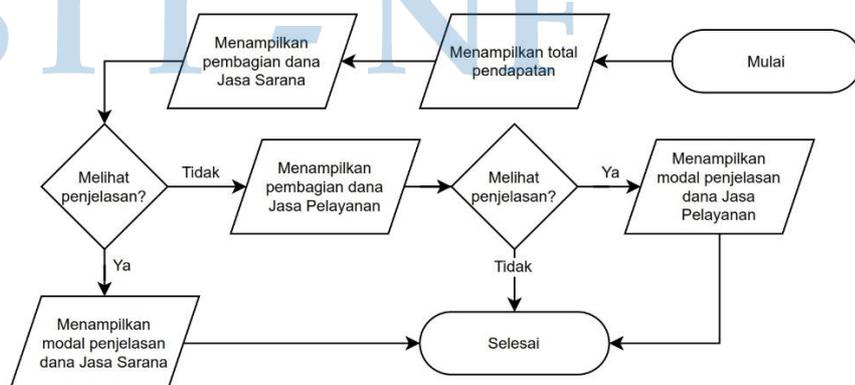
- g. Jurnal: Pengguna bisa melihat jurnal pendapatan dari data yang sudah dikonversi.



Gambar 4.7 Flowchart Jurnal

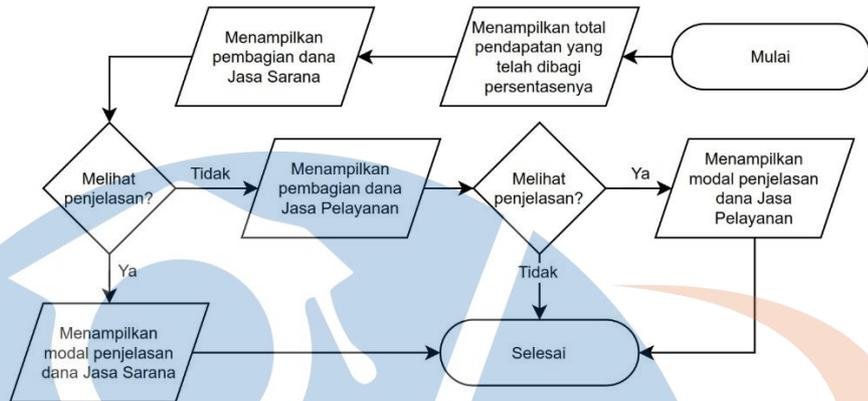
- h. Remunerasi dokter: Ada 2 metode alokasi dana untuk penggajian dokter yaitu metode shifting dan metode allocation & distribution.

- 1) *Shifting*: Pengguna bisa melihat bagaimana pembagian dana untuk setiap dokter berdasarkan metode *shifting*.



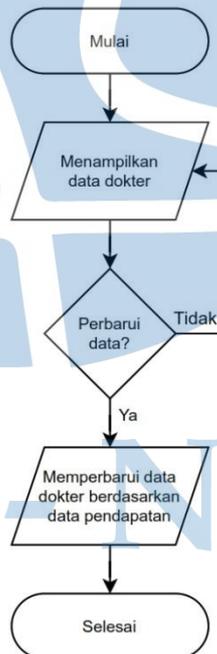
Gambar 4.8 Flowchart Shifting

2) *Allocation & distribution*: Pengguna bisa melihat bagaimana pembagian dana untuk setiap dokter berdasarkan metode *Allocation & Distribution*.



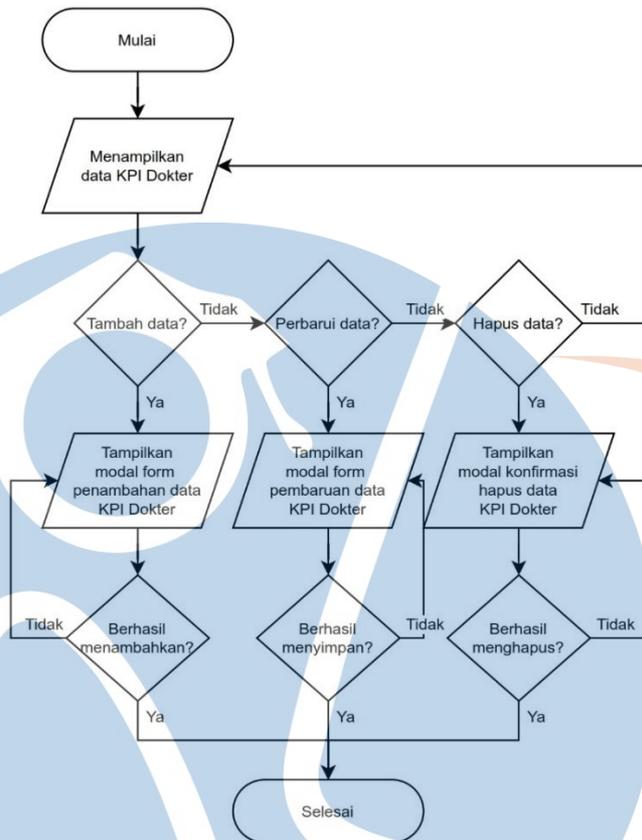
Gambar 4.9 Flowchart Allocation & Distribution

i. **Data dokter**: Pengguna dapat melihat semua data dokter yang menangani pelayanan di rumah sakit berdasarkan data yang diunggah. Pengguna juga dapat memperbarui data secara berkala ketika ada data diunggah ulang.



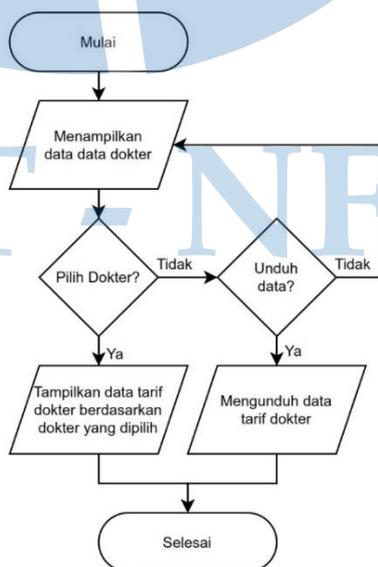
Gambar 4.10 Flowchart Data Dokter

- j. KPI (Key Performance Indicator) dokter: Pengguna dapat menentukan nilai KPI pada setiap dokter.



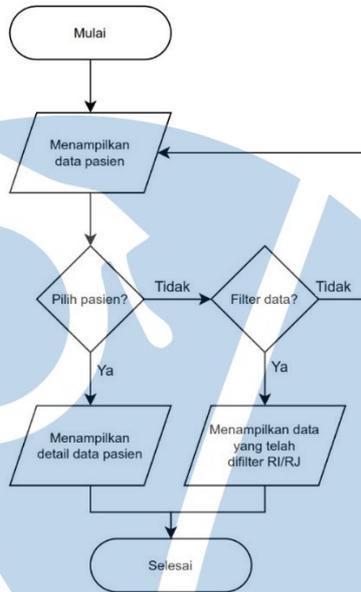
Gambar 4.11 Flowchart KPI Dokter

- k. Tarif dokter: Pengguna dapat memilih dan menentukan tarif setiap dokter di setiap pelayanan.



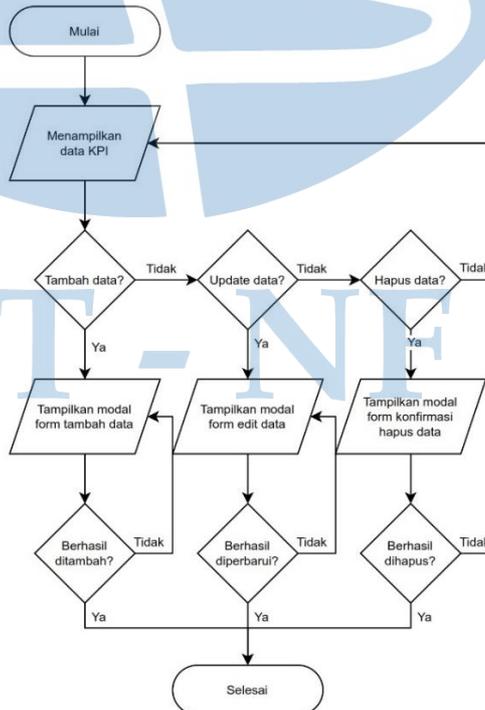
Gambar 4.12 Flowchart Tarif Dokter

1. Pasien non BPJS & pasien BPJS: Pengguna dapat melihat data pasien non BPJS dan pasien BPJS. Selain bisa melihat data setiap pasien, pengguna juga bisa mengunduh data yang berupa data *excel*.



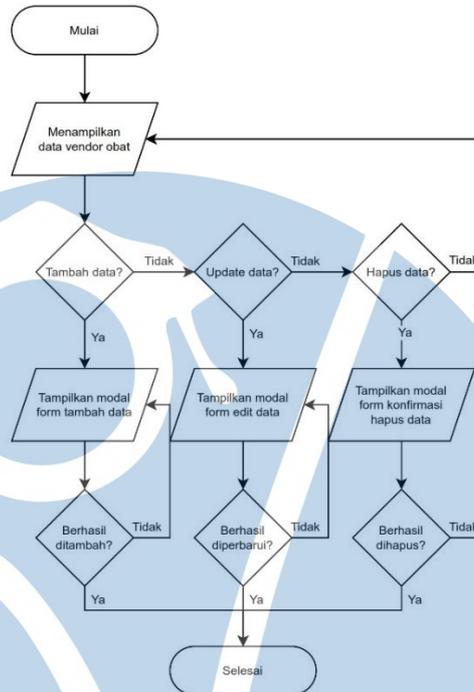
Gambar 4.13 Flowchart Pasien Non BPJS & BPJS

- m. Data KPI: Pengguna bisa mengatur data KPI untuk digunakan sebagai penilaian yang akan berpengaruh terhadap penggajian dokter.



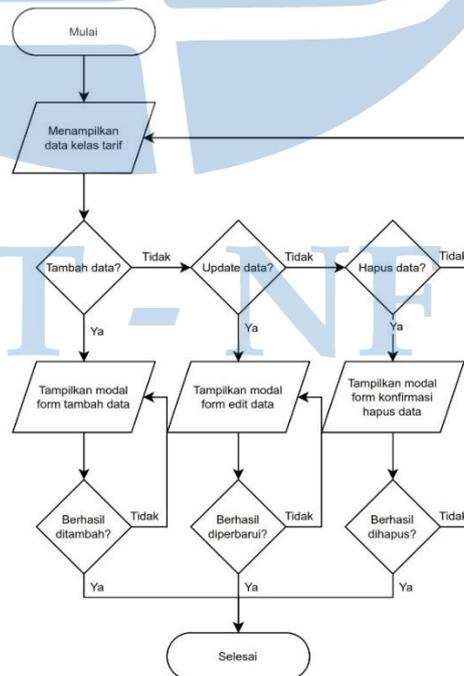
Gambar 4.14 Flowchart Data KPI

- n. Data vendor obat: Pengguna bisa mengatur data vendor obat untuk data koleksi mereka. Guna untuk bisa memfilter pembelian obat mana yang tidak menggunakan vendor yang telah ditetapkan oleh rumah sakit.



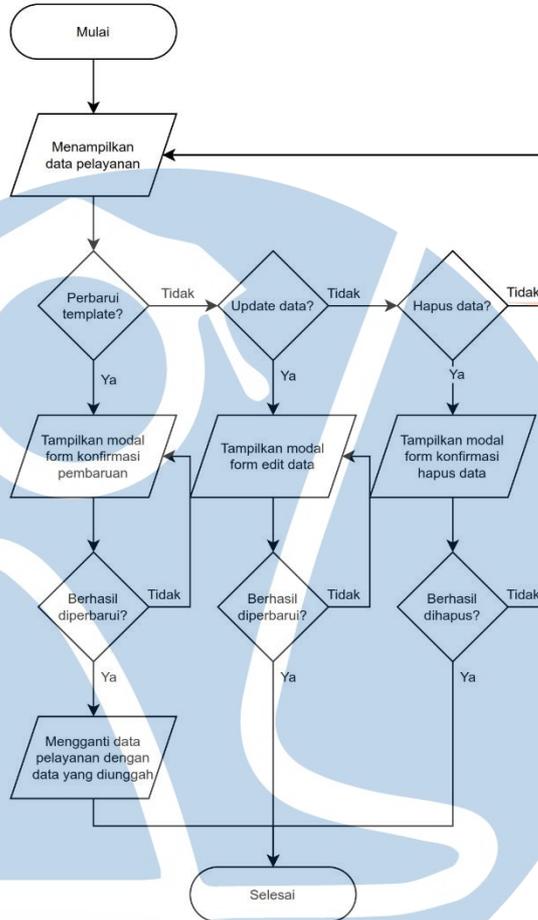
Gambar 4.15 Flowchart Data Vendor Obat

- o. Data kelas tarif: Pengguna dapat mengatur data kelas tarif yang dipakai, guna untuk membuat pelaporan atau laporan keuangan.



Gambar 4.16 Flowchart Data Kelas Tarif

- p. Pelayanan rumah sakit: Pengguna dapat melihat semua data pelayanan yang ada di rumah sakit, selain itu pengguna juga bisa menentukan persentase pembagian dana untuk jasa sarana dan jasa pelayanan.



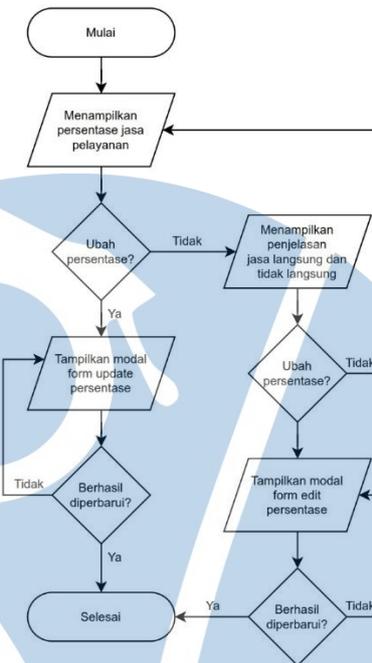
Gambar 4.17 Flowchart Pelayanan Rumah Sakit

- q. Jasa sarana: Pengguna dapat melihat persentase untuk jasa sarana dalam pengalokasian dana pendapatan.



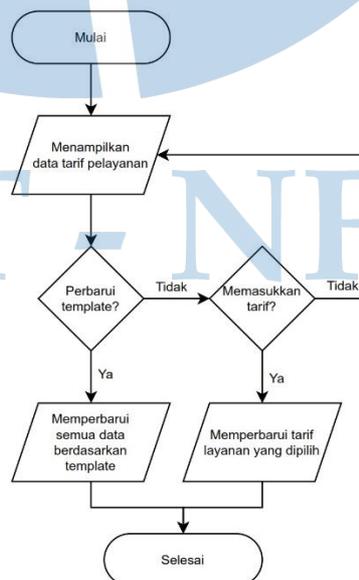
Gambar 4.18 Flowchart Jasa Sarana

- r. Jasa pelayanan: Pengguna dapat melihat persentase untuk jasa pelayanan, selain itu pengguna juga bisa langsung mengatur pembagian persentase untuk jasa pelayanan langsung dan tidak langsung.



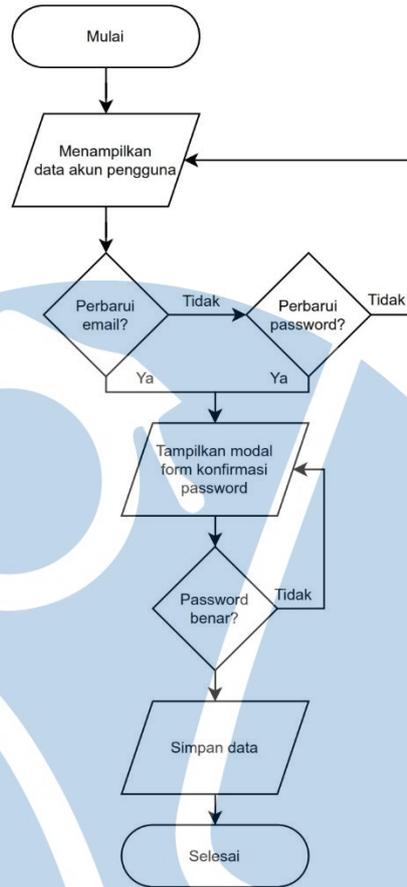
Gambar 4.19 Flowchart Jasa Pelayanan

- s. Tarif rumah sakit: Pengguna dapat melihat dan menentukan tarif rumah sakit untuk setiap layanan. Selain itu, pengguna bisa mengembalikan ke semula yakni dengan menggunakan template yang sudah ada pada tampilan aplikasi.



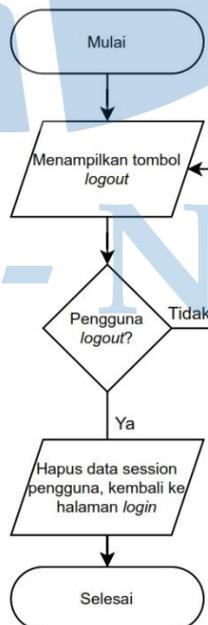
Gambar 4.20 Flowchart Tarif Rumah Sakit

t. Profil: Pengguna dapat mengatur data akun seperti email dan password.



Gambar 4.21 Flowchart Profil

u. Logout: Pengguna dapat keluar aplikasi.



Gambar 4.22 Flowchart Logout

4.2.5 Wireframe

- a. *Login*: Halaman ini menampilkan *form login* sebagai autentikasi aplikasi. Bukan hanya itu, terdapat juga logo dan nama aplikasinya.



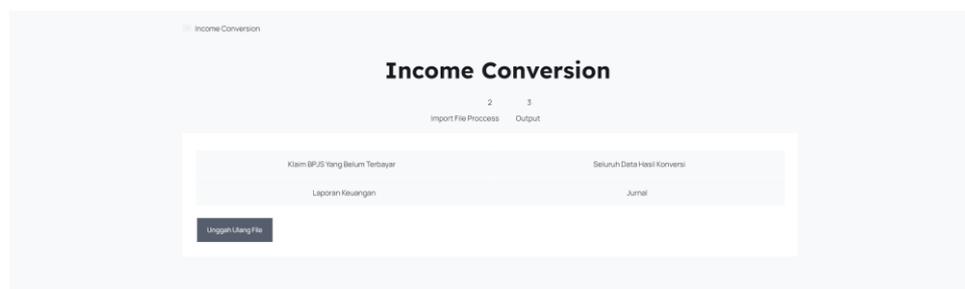
Gambar 4.23 Wireframe Login

- b. *Dashboard*: Halaman ini menampilkan poin penting beberapa fitur yang ada di aplikasi.



Gambar 4.24 Wireframe Dashboard

- c. *Income conversion*: Halman ini menampilkan beberapa data yang bisa diolah oleh sistem setelah data dikonversi. Seperti melihat piutang BPJS, data hasil konversi, laporan keuangan bahkan sampai pembuatan jurnal.



Gambar 4.25 Wireframe Income Conversion

- d. Piutang BPJS: Halaman ini menampilkan data para pasien yang dimana pembayarannya belum dibayar oleh pihak BPJS.

KIaim BPJS Yang Belum Terbayar

80 Pasien RJ 39 Pasien RI

Rawat Inap download excel

Menampilkan Data Pasien Rawat Inap

1. ADITYA LUTHFI DARY AN
No Trans: RBI-22-12-1437

RM	TANGGAL	UNIT	PRODUK	DOKTER	JUMLAH
000004	2023-01-01	Laboratorium	HT/TROMBO	Nama dokter	55.000
000004	2023-01-01	Laboratorium	BIAYA LABCRAT	Nama Dokter	0
Jumlah					55.000

Gambar 4.26 Wireframe Piutang BPJS

- e. Data konversi: Halaman ini menunjukkan data hasil konversi di setiap pasiennya, baik untuk pelayanan rawat inap maupun rawat jalan.

84 Pasien Rawat Jalan 10 Pasien Rawat Inap

Rawat Jalan download excel

Menampilkan Data Pasien Rawat Jalan

1. AMIN, NY-12184

Total Tarif RS kepada Pasien BPJS Rp. INACBG M-3-16-0
Total Tarif RI yang dibayarkan ke BPJS Rp. NAFTRMS RIJ-1U23102-039
Total Penawaran dan BPJS Rp. Tanggal 2 Jan 2023

PRICOLUK	DOKTER	TARIF RS	PERSENTASE	SETIAP DIKONVERSI
TINDAKAN FISIKOTERAPI STIMULASI	nama dokter	55.000	22.00%	25.036
KONSUL DOKTER SP REHAB MEDIK	nama dokter	150.000	52.00%	59.176
TINDAKAN FISIKOTERAPI MWD (MCKR)	nama dokter	55.000	22.00%	25.036
PENDAFTARAN RAWAT JALAN PASIEN	nama dokter	10.000	4.00%	4.952
Jumlah		250.000	100%	113.800

Gambar 4.27 Wireframe Data Konversi

- f. Laporan keuangan: Halaman ini akan memberikan laporan keuangan secara otomatis hasil data dari proses konversi.

Hasil Konversi Pendapatan BPJS untuk Menghilangkan Resiko Piutang Tak Tertagih.

Jenis	Pengantar	Total Pendapatan (Rp)
RAWAT INAP	NON BPJS	000000
Rawat Jalan	NON BPJS	000000
Rawat Inap	BPJS	000000
Rawat Jalan	BPJS	000000
Jumlah		000000

Rawat Inap	Pendapatan	Rawat Jalan	Pendapatan
Kelas Tarif		Kelas Tarif	
Jasa Penunjang medis dan Jasa Tenaga Ahli	000000	Jasa Penunjang medis dan Jasa Tenaga Ahli	000000
Obat dan Perengklapan Medis	000000	Obat dan Perengklapan Medis	000000
Fasilitas Rumah Sakit	000000	Fasilitas Rumah Sakit	000000
Kamar Operasi	000000	Kamar Operasi	000000
Pendapatan Administrasi dan Lainnya	000000	Jumlah	000000
Jumlah	000000		

Jumlah Pendapatan	
Rawat Inap	000000
Rawat Jalan	000000
Jumlah	000000

Gambar 4.28 Wireframe Laporan Keuangan

g. Jurnal: Membuat jurnal dari data yang sudah dikonversi.

Jurnal Pendapatan
[Download Jurnal](#)

Tanggal Invoice	Tanggal Claim BPJS	No RM	Nama Pasien	Tarif RS	Claim BPJS		
2022-09-02	2022-09-02	00004	Nama pasien 1	8.071,99	5.943,90	Bank	5.043,90
						RI - Jasa Penunjang medis dan jasa	
						Tenaga Ahli	245,538
						RI - Obat dan Perengklapan Medis	437,203
						RI - Fasilitas Rumah Sakit	5.713,434
						RI - Kamar Operasi	915,839
						RI - Pendapatan Administrasi dan Lainnya	132,055
2022-09-02	2022-09-02	00002	Nama pasien 2	2.887,13	4.229,80	Bank	4.229,80
			TN			RI - Jasa Penunjang medis dan jasa	
						Tenaga Ahli	586,021
						RI - Obat dan Perengklapan Medis	999,359

Gambar 4.29 Wireframe Jurnal

h. Remunerasi: Dalam halaman ini, terdapat pembagian dana rumah sakit terhadap jasa sarana dan jasa pelayanan rumah sakit. Selain itu terdapat juga pembagian langsung untuk jenis jasa langsung dan tidak langsung.



Gambar 4.30 Wireframe Remunerasi

i. Data dokter: Halaman ini menampilkan data nama-nama dokter yang terdaftar pada data pendapatan rumah sakit yang diunggah.

Data Tenaga Ahli dari Dokter Spesialis, Umum, Anestesi dan Lainnya (Jasa Langsung) **24%** dari total pendapatan

STI - NF [Perbaru](#)

No.	Nama Dokter
1	nama dokter
2	nama dokter
3	nama dokter
5	nama dokter
	nama dokter

Gambar 4.31 Wireframe Data Dokter

j. Data KPI dokter: Halaman ini sebagai penentu nilai KPI pada setiap dokter.

KPI DOKTER

Indikator Kinerja Utama (Key Performance Indicator/KPI) adalah parameter atau metrik yang digunakan untuk mengukur dan mengendalikan tingkat kinerja suatu organisasi atau institusi, termasuk rumah sakit. KPI rumah sakit dirancang untuk mengukur efektivitas, efisiensi, kualitas, keamanan, dan aspek lain dari pelayanan kesehatan yang disediakan. Berikut ini adalah beberapa contoh KPI yang umum digunakan dalam pengukuran kinerja rumah sakit:

Tambah Data

#	Indikator Kinerja Utama	Bobot
1	Indeks Dasar (Basic Index)	6
2	Indeks Kompetensi dan Kualifikasi	
	Indeks Resiko	
	Indeks Emergensi	
5	Indeks Posisi	
6	Indeks Kinerja dan Disiplin	

1 DR. MASALUS NANOOUTH, SPFD

Indikator Kinerja Utama	Kelompok	Nilai	Bobot	Total
Indeks Dasar (Basic Index)	Setiap Rp100.000.000 (seratus ribu rupiah) gaji pokok sama dengan (satu) nilai indeks, pegawai non PNS gaji pokoknya disesuaikan dengan gaji pokok terendah PNS	N/A	6	6
Indeks Kompetensi dan Kualifikasi	S1 Non Profesi dan DiA	6	6	6
Indeks Resiko	B. Grade II (Resiko Pisk) : Staf Admin Umum-Persantoran Shift, Staf Keuangan Shift, IPD/Shift, DiA, Laundry Farmasi Shift, Rawat Jalan, CSID, Radiologi Non Shift, Laboratorium Non Shift, Pemuliharaan Jenazah, Biosurance, Security, Porter Shift, PPSIS Shift, PNH, Generator Changan dan Incinerator	2,5	2,5	2,5
Indeks Emergensi	B. Grade II (Non Rawat - Shift dan Rawat - Non Shift) : Staf Administrasi Umum Persantoran Shift, Staf Keuangan Shift, PPSIS Shift, DiA, Laundry Farmasi Shift, Rawat Jalan, CSID, Radiologi Non Shift, Laboratorium Non Shift, Rawat Medis, Pemuliharaan Jenazah, Biosurance, Security, IPD/Shift, PPSIS Shift, PNH, Generator Changan dan Incinerator	2,5	2,5	2,5
Indeks Posisi	B. Grade II Pengawas Koperasi/Kawatir, Kepala Ruangan, Bendahara Pemerim, Pengelola dan Wakil Kepala Instalasi	2,5	2,5	2,5
Indeks Kinerja dan Disiplin	Kehadiran Tepat Waktu > 20%			

Gambar 4.32 Wireframe Data KPI Dokter

k. Tarif dokter: Halaman ini untuk menentukan nilai tarif dokter, dimana bisa ditentukan harganya secara manual ataupun unggah data berupa file excel. Data yang dimasukkan bisa per dokter sekaligus per pelayanan.

Tarif Dokter

Tentukan tarif masing-masing Dokter pada setiap prosedur atau pelayanan yang ada di Rumah Sakit

Download template tarif dokter

Pilih Dokter

Nama dokter

Dengan mengunggah file, seluruh data tarif dokter dr. Abdu Aziz, M.Sc. Sp.S akan diabaikan

Unggah File Dsm

CHOOSE FILE No file chosen
*file harus bertipe excel (xls)

Atau Manual

Cari pelayanan

No	Layanan	Nama dokter	Tarif Dokter
	Pelayanan 1		500000
	Pelayanan 2		500000
	Pelayanan 3		500000

Gambar 4.33 Wireframe Darif Dokter

l. Data pasien non BPJS: Menampilkan data pasien-pasien yang bukan BPJS. Yakni pasien umum dan asuransi.

2

PASIENT UMUM & ASURANSI RAWAT JALAN

Pilih Kategori Pasien:

Rawat Jalan

Previous page Next

1. FULANAH1, NY	RM: 000379
2. FULANAH2, NY	RM: 12421

Previous page Next

Gambar 4.34 Wireframe Pasien Non BPJS

m. Data pasien BPJS: Halaman ini menampilkan seluruh data pasien BPJS.

84
PASIEAN BPJS RAWAT JALAN

Pilih Kategori Pasien: Rawat Jalan

← Previous page Next →

1. Nama Pasien, TN	RM: 12411
2. Nama Pasien	RM: 12419
3. Nama Pasien, AN	RM: 12390
4. Nama Pasien AN	RM: 12396

Gambar 4.35 Wireframe Data Pasien BPJS

n. Data KPI: Halaman ini berfungsi sebagai penentu nilai apa saja yang akan dijadikan sebagai indikator dalam KPI.

Indikator Kinerja Utama (KPI)

Nilai terukur yang berfungsi untuk mengukur seberapa efektif rumah sakit dalam mencapai tujuan utamanya.

Tambah Data

#	Indikator Kinerja Utama	Bobot	Aksi
1	Indeks Dasar (Basis Index)	6	
2	Indeks Kompetensi dan Kualitas		
3	Indeks Resiko		
4	Indeks Emergensi		
5	Indeks Pasasi		
6	Indeks Kinerja dan Disiplin		

1. Indeks Dasar (Basis Index)

Kategori: Pokok

Setiap Rp/100,000 (satu ribu rupiah) gaji pokok sama dengan (satu) nilai indeks, pegawai non PNS gaji pokoknya diamakan dengan gaji pokok terendah PNS.

Nilai	Bobot	Aksi
N/A	6	

Total Bobot: 6

Gambar 4.36 Wireframe Data KPI

o. Data vendor obat: Halaman ini menentukan vendor obat mana saja yang memang memiliki kerjasama secara sah dengan rumah sakit.

DATA VENDOR OBAT

Tambah Vendor

Simpan

#	Vendor	Aksi
1	Vendor A	
2	Vendor B	
3	Vendor C	

Gambar 4.37 Wireframe Data Vendor Obat

p. Data kelas tarif: Halaman ini bertujuan untuk menentukan kelas tarif rumah sakit.

21
Kelas Tarif

Tambah data

No	Kelas Tarif	Kategori Pendapatan	Aksi
1	Tanpa Kelas Tarif	Tanpa Kategori	<input type="checkbox"/>
2	Visite Konsul Dokter	Jasa Penunjang medis dan Jasa Tenaga Ahli	<input type="checkbox"/>
3	Tindakan Medis Perawatan	Jasa Penunjang medis dan Jasa Tenaga Ahli	<input type="checkbox"/>
4	Ruang Perawatan	Kamer Operasi	<input type="checkbox"/>

Gambar 4.38 Wireframe Kelas Tarif

- q. Pelayanan rumah sakit: Halaman ini untuk mengatur persentase pengalokasian dana untuk jasa sarana dan jasa pelayanan di setiap pelayanannya.

No	Layanan	JS (%)	JP (%)	Aksi
2	ABDOMEN 3 POSISI	60%	40%	⊙
3	ADMINISTRASI MEDOKAUSA	60%	40%	⊙
4	ADMINISTRASI PASIEN BARU	60%	40%	⊙
5	AFFID	60%	40%	⊙
6	AFF HEACTING DR SPESIA	60%	40%	⊙
7	ADD	60%	40%	⊙

Gambar 4.39 Wireframe Pelayanan Rumah Sakit

- r. Jasa sarana: Halaman ini menampilkan persentase jasa sarana, serta penjelasan mengenai jasa sarana.

Jasa Sarana (JS)

60%

Komponen jasa sarana merupakan imbalan yang diterima oleh Rumah Sakit atas pemeliharaan dan/atau sarana non medis, dalam keadaan barunya/ada kesesuaian habis pakai yang digunakan langsung dalam rangka Pelayanan Medis dan Pelayanan Penunjang Medis.

Gambar 4.40 Wireframe Jasa Sarana

- s. Jasa pelayanan: Halaman ini menampilkan persentase jasa pelayanan. Dimana sekaligus bisa mengubah persentasenya. Berlaku juga untuk penentuan jasa langsung dan tidak langsungnya.

Jasa Pelayanan (JP)

40%

Komponen jasa pelayanan merupakan imbalan yang diterima oleh pemberi pelayanan atas jasa yang diberikan kepada pasien/obat rawat inpatient Pelayanan Medis, Pelayanan Penunjang Medis dan/atau pelayanan lainnya.

60%

Jasa Pelayanan Langsung: Medis dan Penunjang Medis (Fungsional)

1. Tenaga Ahli/ Dokter Spesialis dan Umum
2. Tenaga Ahli Anestesi dan Lainnya

40%

Jasa Pelayanan Tidak Langsung: Non-Medis (Struktural) (Diluar bentuk Gaj, Honorarium, Tunjangan, Bonus di 1. Direktur Utama, Keuangan, Operasional, Pemasaran dst 2. Kepala Bagian setiap Divisi 3. Perawat, Apoteker, Pemasaran, Keuangan, Akuntansi, Perpajakan dan CS, Keamanan dan Kebersihan.

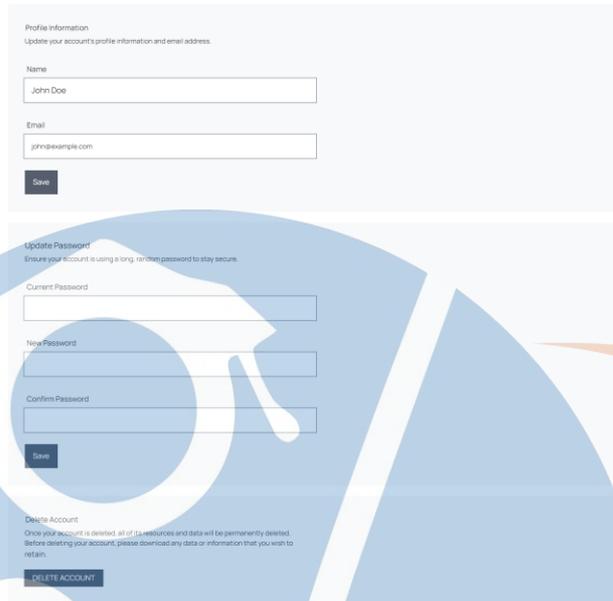
Gambar 4.41 Wireframe Jasa Pelayanan

- t. Tarif rumah sakit: Menentukan tarif setiap pelayanan rumah sakit.

No	Layanan	Persentase Pelayanan		Tarif Rumah Sakit
		Jasa Sarana (%)	Jasa Pelayanan (%)	
	ABDOMEN	80	20	0
2	ABDOMEN 3 POSISI	60	40	0

Gambar 4.42 Wireframe Tarif Rumah Sakit

- u. Profil: Menampilkan informasi akun pengguna aplikasi sekaligus bisa diperbarui.



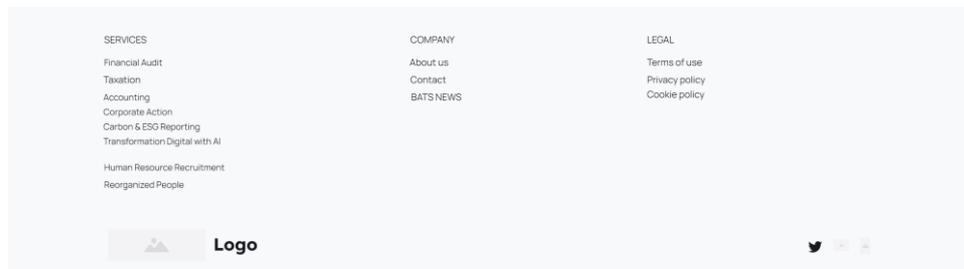
Gambar 4.43 Wireframe Profil

- v. Header: Berisi menu-menu aplikasi.



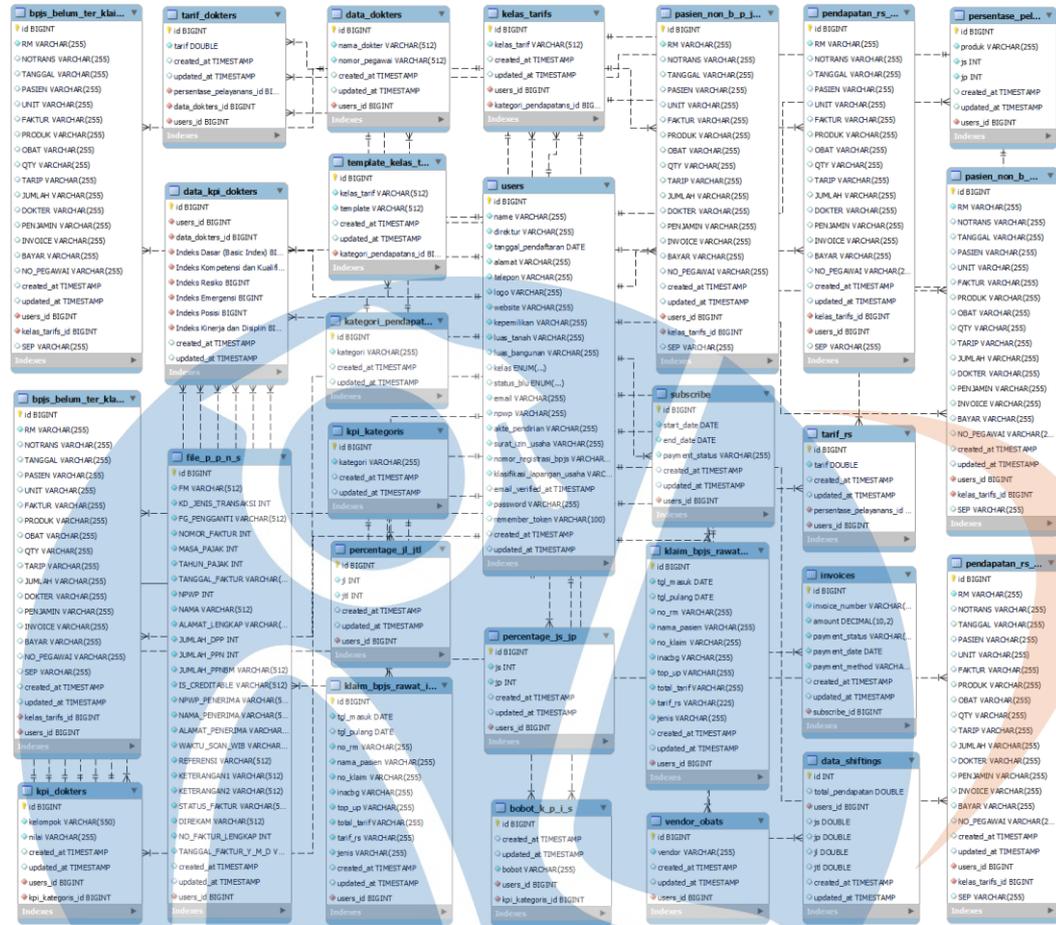
Gambar 4.44 Wireframe Header

- w. Footer: Informasi footer meliputi sumber utama informasi perusahaan pengembang, dan informasi layanan lainnya yang disediakan oleh perusahaan pengembang.



Gambar 4.45 Wireframe Footer

4.2.6 ERD Diagram



Gambar 4.46 ERD Diagram

Diagram ERD yang dikembangkan untuk aplikasi ini menggambarkan struktur data yang mendukung operasional Rumah Sakit XYZ dalam konteks klaim BPJS dan remunerasi dokter. Dalam ERD ini, penulis menunjukkan entitas penting seperti Dokter, Pasien, Klaim BPJS, dan Tarif, serta relasinya:

- Dokter dan Pasien: Tiap dokter dapat memiliki beberapa pasien. Baik itu pasien rawat inap, rawat jalan dari kategori pasien BPJS maupun pasien umum. Relasi ini mencerminkan bahwa dokter bertanggung jawab terhadap lebih dari satu pasien pada waktu yang sama, dan setiap pasien bisa mendapatkan pelayanan dari dokter yang sama atau berbeda tergantung pada kebutuhan medis mereka.
- Dokter dan Tarif: Setiap dokter dikaitkan dengan tarif yang bervariasi berdasarkan jasa yang diberikan. Tarif ini dapat berubah berdasarkan nilai

KPI (*Key Performance Indicator*). Selain itu dokter bisa dikaitkan berdasarkan dua jenis tarif, yakni tarif rumah sakit yang dikhususkan untuk setiap pelayanannya dan tarif dokter yang dikhususkan pada setiap dokter di setiap pelayanannya, dua hal ini tergantung dari kebijakan rumah sakit.

- c. Pasien dan Klaim BPJS: Setiap pasien memiliki satu atau lebih klaim BPJS yang merekam detail transaksi keuangan terkait perawatan yang diterima. Klaim ini menghubungkan pasien dengan dana yang dialokasikan oleh BPJS. Setiap pasien memiliki nomor rekam medis yang dimana akan dijadikan sebuah kunci identitas pengalokasian dana.
- d. Klaim BPJS dan Tarif: Klaim BPJS berhubungan dengan tarif yang ditentukan berdasarkan perawatan yang diberikan kepada pasien. Tarif ini ditentukan oleh INA-CBGs yang mencerminkan kompleksitas dan jenis perawatan yang diperlukan.

Diagram ini dirancang untuk memfasilitasi pemahaman yang lebih baik tentang bagaimana data antar tabel saling berhubungan, mendukung fungsi aplikasi dalam meningkatkan transparansi dan efisiensi pengelolaan keuangan rumah sakit.

4.2.6.1 Metode Spiral dalam Pengembangan Aplikasi

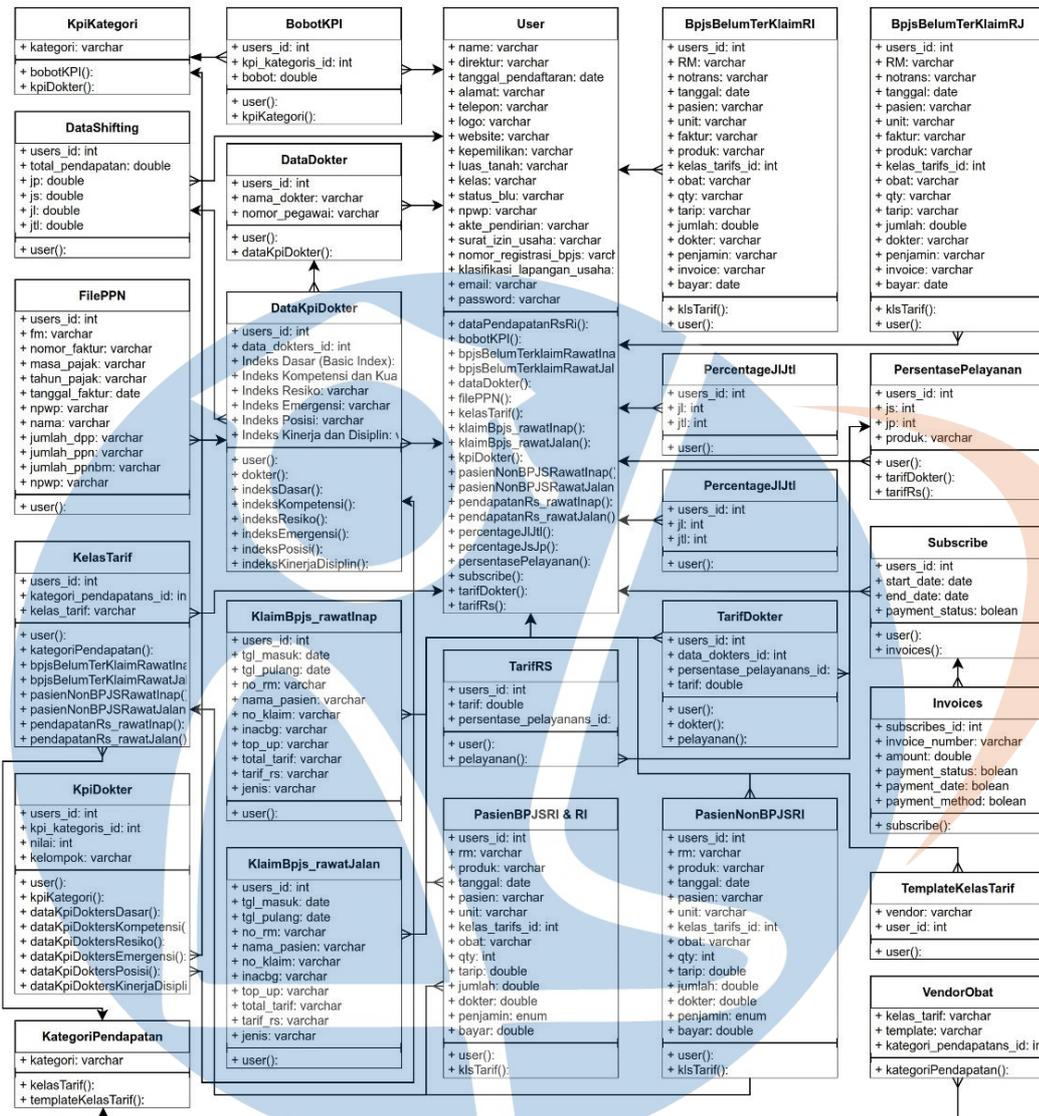
Metode Spiral telah diterapkan dalam pengembangan aplikasi *bridging* BPJS dan sistem remunerasi dokter ini untuk memfasilitasi proses iterasi yang berkelanjutan dan memungkinkan integrasi umpan balik pengguna secara efektif. Proses ini dilakukan melalui beberapa tahapan:

- a. Perencanaan Awal: Tahap ini melibatkan identifikasi kebutuhan dasar sistem dan perumusan tujuan umum aplikasi. Ini termasuk pemahaman mendalam mengenai proses klaim BPJS dan sistem remunerasi dokter di Rumah Sakit XYZ.
- b. Analisis Risiko: Setiap aspek dari aplikasi, seperti tahap konversi dari data klaim BPJS, penghitungan pembuatan jurnal, penghitungan pembuatan laporan keuangan dan penghitungan remunerasi dokter, dianalisis untuk risiko potensial yang mungkin mengganggu keakuratan data finansial.

- c. Pembuatan Prototipe Awal: Prototipe pertama aplikasi dibuat berdasarkan kebutuhan yang telah diidentifikasi. Prototipe ini mencakup fungsi inti seperti penginputan data klaim, penginputan data pendapatan rumah sakit, perhitungan remunerasi, pembuatan jurnal, dan laporan keuangan.
- d. Evaluasi Pengguna: Prototipe dievaluasi oleh konsultan dibidang keuangan pihak Rumah Sakit XYZ. Umpan balik yang diterima digunakan untuk menilai efektivitas dan kegunaan fitur-fitur yang ada.
- e. Iterasi Berikutnya: Berdasarkan umpan balik yang diterima, penulis melakukan revisi dan penyempurnaan aplikasi. Iterasi ini meliputi penyesuaian antarmuka pengguna, peningkatan akurasi perhitungan remunerasi, peningkatan akurasi perhitungan dalam pembuatan laporan keuangan, dan peningkatan akurasi perhitungan dalam pembuatan jurnal.
- f. Implementasi dan Evaluasi Lanjutan: Setelah iterasi, aplikasi diimplementasikan secara penuh dan terus dipantau untuk efektivitasnya dalam lingkungan operasional. Evaluasi terus-menerus ini memastikan bahwa aplikasi tetap relevan dan bermanfaat dalam mengatasi dinamika yang berubah dan tuntutan operasional rumah sakit.

Penerapan metode Spiral memungkinkan penulis mengembangkannya secara adaptif meningkatkan aplikasi dengan memperhatikan risiko, kebutuhan, dan realitas operasional yang berubah. Pendekatan ini membantu dalam menciptakan solusi yang tidak hanya teknis memadai tetapi juga sangat sejalan dengan operasional dan kebutuhan strategis Rumah Sakit XYZ, memastikan aplikasi ini memberikan nilai tambah nyata dalam manajemen keuangan dan operasional rumah sakit.

4.2.7 Class Diagram



Gambar 4.47 Class Diagram

Diagram kelas yang disajikan dalam Gambar 4.46 memberikan pandangan arsitektural yang detail tentang sistem pengembangan aplikasi *bridging* BPJS dan remunerasi dokter pada Rumah Sakit XYZ. Diagram ini secara khusus menggambarkan struktur kelas yang mendefinisikan bagaimana data dan fungsi-fungsi penting dikelola dalam sistem untuk mendukung operasi sehari-hari.

Komponen Diagram Kelas:

- Kelas: Setiap kelas diwakili oleh nama yang mencerminkan fungsinya dalam sistem. Misalnya, kelas Dokter, Pasien, dan KlaimBPJS

menunjukkan objek-objek utama yang berinteraksi dalam proses klaim dan remunerasi.

- b. Atribut: Atribut dalam kelas seperti namaDokter, tarifKlaim, dan statusPasien menyediakan informasi spesifik yang diperlukan untuk operasional sistem serta laporan yang dihasilkan.
- c. Metode: Metode seperti hitungRemunerasi() dan verifikasiKlaim() menggambarkan aksi-aksi yang dapat dijalankan terhadap objek-objek, berperan vital dalam logika bisnis aplikasi.

Hubungan Antar Kelas:

- a. Asosiasi: Hubungan antara kelas-kelas seperti Dokter dan KlaimBPJS menunjukkan interaksi langsung dalam proses pengajuan dan penanganan klaim remunerasi.
- b. Pewarisan: Digunakan untuk menunjukkan hubungan hierarki, seperti kelas DokterSpesialis yang mewarisi sifat dari kelas Dokter, mencerminkan struktur organisasi rumah sakit.
- c. Komposisi: Menunjukkan hubungan kepemilikan, seperti RumahSakit yang memiliki objek Dokter dan Pasien, penting untuk menggambarkan bagaimana data terorganisir dan dikelola dalam sistem.

Diagram kelas ini tidak hanya memfasilitasi pemahaman tentang desain sistem tetapi juga penting dalam fase implementasi dan pengujian, memastikan semua fungsi sistem terintegrasi dengan baik dan beroperasi sesuai dengan kebutuhan Rumah Sakit XYZ. Ini juga membantu tim pengembang dan manajemen rumah sakit dalam memahami dan mengelola alur kerja remunerasi dan klaim BPJS secara lebih efektif.

4.3 Hasil Implementasi

Pada bagian ini, akan diuraikan hasil dari proses implementasi aplikasi yang telah dikembangkan menggunakan metode pengembangan Spiral. Pendekatan iteratif ini memungkinkan tim pengembang untuk terus menerapkan penyesuaian dan perbaikan berdasarkan umpan balik yang diperoleh selama setiap siklus pengembangan. Hal ini penting untuk memastikan bahwa aplikasi tidak hanya memenuhi spesifikasi teknis yang telah ditetapkan, tetapi juga secara efektif menangani kebutuhan pengguna dan tantangan operasional yang ditemui di lapangan. Hasil implementasi yang akan dibahas mencakup berbagai aspek teknis dari aplikasi, termasuk fungsi-fungsi kunci, interaksi dengan sistem data BPJS, dan sistem kompensasi dokter, serta bagaimana aplikasi ini mendukung peningkatan transparansi dan akuntabilitas dalam pengelolaan kompensasi dokter di rumah sakit.

4.3.1 Menu Aplikasi:

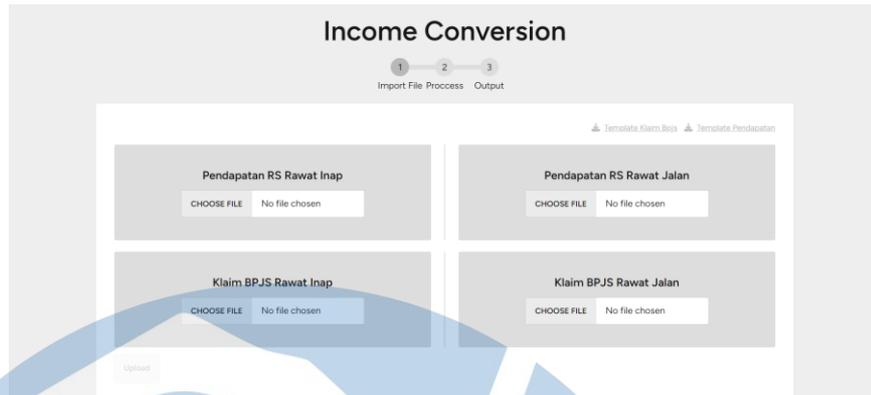
- a. **Dashboard:** Menampilkan gambaran besar terkait keunggulan aplikasi ini dan sebagian data besar seperti pembagian persentase untuk jasa sarana dan jasa pelayanan.



Gambar 4.48 Aplikasi - Dashboard

- b. **Income Conversion:** Fitur dimana pengguna bisa mengunggah dokumen pendapatan rumah sakit dan klaim BPJS yang akan dikonversi, setelah proses konversi selesai pengguna dapat melihat data yang sudah diolah seperti klaim BPJS mana saja yang belum terbayar, seluruh data hasil konversi, laporan keuangan bahkan sampai jurnal.

1) Tampilan sebelum mengunggah dokumen



Gambar 4.49 Aplikasi – *Income Conversion*

2) Tampilan sesudah mengunggah dokumen



Gambar 4.50 Aplikasi – *Income Conversion Uploaded*

c. Klaim BPJS yang belum terbayar: Berisi data-data pasien rumah sakit yang dimana dana klaim BPJS nya belum dibayar oleh pihak BPJS. Pengguna dapat memfilter data berdasarkan jenis pelayanan, yakni rawat inap dan rawat jalan.



Gambar 4.51 Aplikasi – *Piutang BPJS*

- d. Seluruh data hasil konversi: Menampilkan seluruh data pendapatan yang sudah dikonversi dengan tarif INA-CBGs. Data ini bisa difilter berdasarkan jenis pelayanan untuk rawat inap ataupun rawat jalan.

84 Pasien Rawat Jalan | 10 Pasien Rawat Inap

Tampilkan berdasarkan

Menampilkan Data Pasien Rawat Inap

1. GANDUNG HARTADI, TN-000020

Total Tarif RS kepada Pasien BPJS : Rp. 2.887.132
 Total Tarif RS yang diajukan ke BPJS : Rp. 2.432.200
 Total Penyermaan dari BPJS : Rp. 4.229.800

INACBG : I-4-79-1
 NoTrans : RBI-23-01-0043
 Tanggal : 2 Jan 2023

PRODUK	DOKTER	TARIF RS	PERSENTASE	SETELAH DI KONVERSI
JASA PARAMEDIS (HCU)	dr. Gadis Ranti Endamatrixa,	50.000	1,73%	73.253
KONSULTASI DR. SP (HCU)	dr. Gadis Ranti Endamatrixa,	65.000	2,25%	95.228
VISITE DR. UMUM (HCU)	dr. Gadis Ranti Endamatrixa,	30.000	1,04%	43.952
KONSULTASI DOKTER UMUM (M)	dr. Gadis Ranti Endamatrixa,	60.000	2,08%	87.903

Gambar 4.52 Aplikasi – Data Konversi

- e. Laporan keuangan: Menampilkan data laporan keuangan yang dimana tujuannya adalah untuk melihat total pendapatan.

Hasil Konversi Pendapatan BPJS untuk Menghilangkan Resiko Piutang Tak Tertagih

Jenis	Pengamin	Total Pendapatan (Rp)
RAWAT INAP	NON BPJS	29.413.724
RAWAT JALAN	NON BPJS	720.000
RAWAT INAP	BPJS	32.871.000
RAWAT JALAN	BPJS	15.098.600
Jumlah		78.103.324

Gambar 4.53 Aplikasi – Laporan Keuangan

- f. Jurnal pendapatan: Menampilkan buku jurnal pendapatan dari data yang telah dikonversi dan jurnal tersebut bisa diunduh.

Jurnal Pendapatan **Shifting**

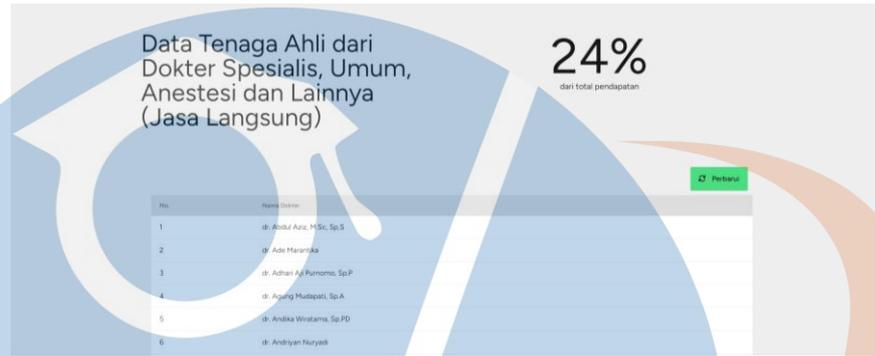
Download Jurnal

No	Tanggal Invoice	Tanggal Claim BPJS	No RM	Nama Pasien	Tarif RS	Claim BPJS		
1	RI - 2023-01-02	2022-09-02	000040	SUHARINI, NN	8.017.996	5.043.900	Bank	5.043.900
							RI - Jasa Penunjang medis dan Jasa Tenaga Ahli	245.338
							RI - Obat dan Perengkapan Medis	437.203
							RI - Fasilitas Rumah Sakit	3.713.414
							RI - Kamar Operasi	515.839
							RI - Pendapatan Administrasi dan Lainnya	132.105
2	RI - 2023-01-02	2022-09-02	000020	GANDUNG HARTADI, TN	2.887.132	4.229.800	Bank	4.229.800
							RI - Jasa Penunjang medis dan Jasa Tenaga Ahli	586.021

Gambar 4.54 Aplikasi – Jurnal

g. Dokter: Berisi seluruh data dokter yang ada dan melayani pasien, (*Key Performance Indicator*) dan data tarif dokter berdasarkan data pendapatan rumah sakit.

- 1) Data dokter: Berisi data nama-nama dokter berdasarkan data pendapatan rumahsakit. Data ini bisa disesuaikan secara berkala dengan data pendapatan yang baru diunggah.



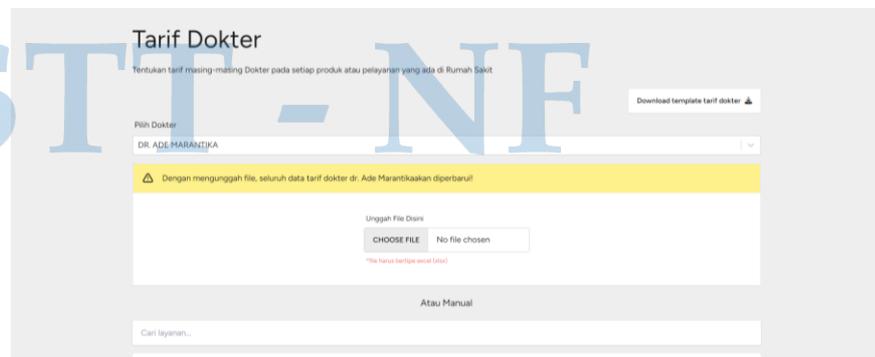
Gambar 4.55 Aplikasi – Data Dokter

- 2) KPI dokter: Fitur untuk menentukan KPI masing-masing dokter.



Gambar 4.56 Aplikasi – Jurnal

- 3) Tarif dokter: Fitur ini untuk menentukan tarif dokter dengan detail, setiap dokter dan setiap layanannya.



Gambar 4.57 Aplikasi – Tarif Dokter

- h. Pasien: Fitur ini untuk melihat data secara detail pasien mana saja yang termasuk pasien umum dan pasien BPJS.

- 1) Pasien Non BPJS: Terdapat nama-nama pasien non BPJS/umum, dan beserta data detail pada setiap pasiennya.

3
PASIEEN UMUM & ASURANSI RAWAT INAP

Pilih Kategori Pasien:
Rawat Inap

1. ALDA SABRINA, NN	RM: 003565
2. FAHRI RAHMAN SETIADI, AN	RM: 010842
3. HAPPY PARAMITHA, NY	RM: 009112

Gambar 4.58 Aplikasi – Pasien Non BPJS

- 2) Pasien BPJS: Terdapat nama-nama pasien BPJS, dan beserta data detail pada setiap pasiennya.

10
PASIEEN BPJS RAWAT INAP

Pilih Kategori Pasien:
Rawat Inap

1. ARDIONO, TN	RM: 000036
2. EKA MARIA UTAMA, NY	RM: 000033
3. GANDUNG HARTADI, TN	RM: 000020
4. MINAH, NY	RM: 000029
5. RIO CAHYOWIDI, TN	RM: 000012

Gambar 4.59 Aplikasi – Pasien BPJS

- i. Other: Berisi fitur-fitur master data yang dimana berguna untuk mengatur sekaligus pelengkap agar hasil data yang diharapkan sesuai dengan tujuan rumah sakit. Berikut adalah beberapa fiturmya:

- 1) Data KPI: Untuk mengatur jenis dan bobot KPI yang akan digunakan untuk pengalokasian pada penggajian dokter.

Indikator Kinerja Utama (KPI)

Nilai terukur yang berfungsi untuk menunjukkan seberapa efektif rumah sakit dalam mencapai tujuan utamanya.

[Tambah Data](#)

#	Indikator Kinerja Utama	Bobot	Aksi
1	Indeks Dasar (Basic Index)	6	✎
2	Indeks Kompetensi dan Kualifikasi	1	✎
3	Indeks Resiko	1	✎
4	Indeks Emergensi	1	✎
5	Indeks Posisi	1	✎
6	Indeks Kinerja dan Disiplin	1	✎

Gambar 4.60 Aplikasi – Data KPI

Kelompok	Nilai	Bobot	Aksi
SD	1	1	
SMP dan Setingkat SLTP	2	1	
SMA dan Setingkat SLTA	3	1	
D1 dan D2	4	1	
D3	5	1	
S1 Non Profesi dan D4	6	1	
Dokter Umum, Dokter Gigi, Nurse, Apoteker, Nutrisiisist dan Profesi Kesehatan Lain	7	1	
S2	8	1	
Dokter Spesialis (Sp1), Dokter Gigi Spesialis	9	1	
S3 dan Dokter Sub Spesialis (Sp2)	10	1	
Total Bobot		10	

Gambar 4.61 Aplikasi – Data KPI 2

- 2) Vendor obat: Fitur ini berguna untuk mengatur vendor obat mana saja yang memang sudah sah bekerjasama dengan rumah sakit, sehingga guna untuk mendeteksi obat mana saja yang pembeliannya diluar vendor obat rumah sakit.

#	Vendor	Aksi
1	Vendor A	
2	Vendor B	
3	Vendor C	

Gambar 4.62 Aplikasi – Vendor Obat

- 3) Kelas tarif: Fitur ini berfungsi untuk mengelompokkan setiap data pendapatan masuk kepada kelas tarif yang mana, misalnya seperti kelas tarif jasa pelayanan, dimana kelas tarif ini termasuk kedalam kategori kamar operasi. Hal ini memudahkan sistem untuk membuat laporan keuangan.

No	Kelas Tarif	Kategori Pendapatan	Aksi
1	Tanpa Kelas Tarif	Tanpa Kategori	
2	Visite/Konsul Dokter	Jasa Penunjang medis dan Jasa Tenaga Ahli	
3	Tindakan Medis Perawatan	Jasa Penunjang medis dan Jasa Tenaga Ahli	
4	Ruang Perawatan	Kamar Operasi	

Gambar 4.63 Aplikasi – Kelas Tarif

- 4) Persentase pelayanan: Fitur ini bertujuan untuk menentukan persentase setiap pelayanan yang ada dari segi berapa persen untuk jasa pelayanan dan jasa sarana. Guna untuk sebagai penentu pengalokasian biaya terhadap pendapatan rumah sakit.

Persentase Pelayanan

Tentukan persentase Jasa Pelayanan dan Jasa Sarana di setiap jasa

Atur ulang Tempas

Cari layanan...

No	Layanan	JS (%)	JP (%)	Aksi
1	ABDOMEN	80%	20%	<input type="checkbox"/>
2	ABDOMEN 3 POSISI	60%	40%	<input type="checkbox"/>
3	ADMINISTRASI HEMODIALISA	60%	40%	<input type="checkbox"/>
4	ADMINISTRASI PASIEN BARU	60%	40%	<input type="checkbox"/>
5	AFF DC	60%	40%	<input type="checkbox"/>
6	AFF HEACTING DR.SESIALIS	60%	40%	<input type="checkbox"/>

Gambar 4.64 Aplikasi – Persentase Pelayanan

- 5) Jasa sarana: Menampilkan informasi persentase serta penjelasan mengenai jasa sarana.

Jasa Sarana (JS) 60%

Komponen jasa sarana merupakan imbalan yang diterima oleh Rumah Sakit atas pemakaian akomodasi, bahan non medis, obat-obatan, bahan/alat kesehatan habis pakai yang digunakan langsung dalam rangka Pelayanan Medis dan Pelayanan Penunjang Medis.

Gambar 4.65 Aplikasi – Jasa Sarana

- 6) Jasa pelayanan: Fitur ini berguna untuk mengatur persentase untuk jasa pelayanan dan jasa sarana. Selain itu, dalam fitur ini bisa menentukan persentase untuk jasa langsung dan jasa tidak langsung.

Jasa Pelayanan (JP) 40%

Komponen jasa pelayanan merupakan imbalan yang diterima oleh pemberi pelayanan atas jasa yang diberikan kepada pasien dalam rangka Pelayanan Medis, Pelayanan Penunjang Medis dan/atau pelayanan lainnya.

Ubah Persentase

Gambar 4.66 Aplikasi – Jasa Pelayanan 1

60%

Jasa Pelayanan Langsung: Medis dan Penunjang Medis (Fungsional)

1. Tenaga Ahli Dokter Spesialis dan Umum
2. Tenaga Ahli Anestesi dan Lainnya

Ubah Persentase

40%

Jasa Pelayanan Tidak Langsung: Non Medis (Struktural)

Dalam bentuk Gaji, Honorarium, Tunjangan, Bonus dll

1. Direktur Utama, Keuangan, Operasional, Pemasaran dst
2. Kepala Bagian setiap Divisi
3. Perawat, Apoteker,
4. Pemasaran, Keuangan, Akuntansi, Perpajakan dan CS
5. Keamanan dan Kebersihan

Gambar 4.67 Aplikasi – Jasa Pelayanan 2

- 7) Tarif Rumah sakit: Fitur ini berguna untuk menentukan berapa tarif dasar rumah sakit untuk setiap pelayanannya. Karena akan ada beberapa kondisi dimana dokter akan dibayar sesuai tarif rumah sakit atau tarif dokter.



Gambar 4.68 Aplikasi – Tarif Rumah Sakit

- 8) Profil: Fitur ini untuk memperbarui informasi profil pengguna, serta bisa mengatur informasi akun seperti update password.



Gambar 4.69 Aplikasi – Profil

4.4 Hasil Pengujian

Setelah implementasi aplikasi selesai, langkah penting selanjutnya adalah pengujian komprehensif untuk memverifikasi kinerja dan fungsionalitas aplikasi sesuai dengan spesifikasi dan kebutuhan pengguna. Bagian ini akan membahas hasil dari berbagai tes yang dilakukan, termasuk *Blackbox Testing* dan *Load Testing*, yang bertujuan untuk menilai aplikasi dari perspektif pengguna akhir dan memastikan bahwa aplikasi dapat mengelola beban kerja di kondisi operasional sebenarnya. Pengujian ini krusial untuk menentukan keandalan dan efektivitas aplikasi dalam mengintegrasikan sistem BPJS dengan sistem kompensasi dokter, serta dalam memberikan transparansi dan akuntabilitas yang diperlukan. Selain itu, hasil pengujian ini juga akan membantu dalam mengidentifikasi area yang masih memerlukan perbaikan dan pengembangan lebih lanjut sebelum aplikasi sepenuhnya siap untuk diluncurkan dalam skala yang lebih besar.

4.4.1 *Blackbox Testing*

Blackbox Testing dilakukan untuk memverifikasi bahwa fitur-fitur dalam aplikasi berfungsi sesuai dengan persyaratan fungsional yang telah ditentukan. Pengujian ini fokus pada output dari aplikasi tanpa perlu mengetahui bagaimana fungsi internal aplikasi tersebut dijalankan. Berikut adalah tabel hasil pengujian *Blackbox* untuk beberapa skenario penting dalam aplikasi:

Table 4.2: Tabel *Blackbox Testing*

No	Skenario Pengujian	Input dan Aksi	Hasil yang Diharapkan	Hasil Aktual	Status
1	<i>Login Pengguna</i>	Masukkan <i>username</i> dan <i>password</i> yang valid	Pengguna berhasil login dan diarahkan ke <i>dashboard</i> utama	Sesuai ekspektasi	Lulus
2	Melihat Dashboard	Akses Dashboard utama	<i>Dashboard</i> utama ditampilkan dengan informasi terkini	Sesuai ekspektasi	Lulus
3	Melihat Halaman <i>Income Conversion</i>	Buka halaman <i>Income Conversion</i>	Halaman <i>Income Conversion</i> terbuka dengan data terkini	Sesuai ekspektasi	Lulus
4	Mengunggah Data Pendapatan dan Klaim BPJS	Unggah file data pendapatan dan klaim BPJS	Data diunggah dan terverifikasi dalam sistem	Sesuai ekspektasi	Lulus
5	Melihat Data Pasien BPJS yang Belum Tertagih	Akses daftar pasien BPJS belum tertagih	Daftar pasien BPJS yang belum tertagih ditampilkan	Sesuai ekspektasi	Lulus
6	Melihat Data Hasil Konversi	Buka halaman untuk melihat data hasil konversi	Data hasil konversi ditampilkan sesuai filter yang diterapkan	Sesuai ekspektasi	Lulus
7	Melihat Jurnal	Buka halaman jurnal	Jurnal keuangan ditampilkan	Sesuai ekspektasi	Lulus

No	Skenario Pengujian	Input dan Aksi	Hasil yang Diharapkan	Hasil Aktual	Status
8	Melihat Laporan Keuangan	Buka halaman laporan keuangan	Laporan keuangan ditampilkan sesuai periode yang dipilih	Sesuai ekspektasi	Lulus
9	Melihat Data Alokasi Remunerasi Dokter Metode <i>Shifting</i>	Lihat data alokasi remunerasi dokter dengan metode <i>shifting</i>	Data alokasi remunerasi dengan metode <i>shifting</i> ditampilkan	Sesuai ekspektasi	Lulus
10	Melihat Data Alokasi Remunerasi Dokter Metode <i>Allocation and Distribution</i>	Lihat data alokasi remunerasi dokter dengan metode <i>allocation and distribution</i>	Data alokasi remunerasi dengan metode <i>allocation and distribution</i> ditampilkan	Sesuai ekspektasi	Lulus
11	Melihat Data Dokter	Akses data dokter yang tersimpan	Data dokter ditampilkan dengan detail	Sesuai ekspektasi	Lulus
12	CRUD Data KPI Dokter	Tambah, ubah, hapus, lihat data KPI dokter	Data KPI dokter dapat ditambahkan, diubah, atau dihapus	Sesuai ekspektasi	Lulus
13	Menentukan Tarif Dokter	Atur tarif gaji untuk dokter	Tarif dokter dapat diatur dan disimpan	Sesuai ekspektasi	Lulus
14	Menentukan Tarif Rumah Sakit	Atur tarif yang berlaku di rumah sakit	Tarif rumah sakit dapat diatur dan disimpan	Sesuai ekspektasi	Lulus
15	Melihat Data Pasien BPJS	Akses data pasien BPJS yang tersedia	Data pasien BPJS ditampilkan	Sesuai ekspektasi	Lulus
16	Melihat Data Pasien Non-BPJS	Akses data pasien non-BPJS yang tersedia	Data pasien non-BPJS ditampilkan	Sesuai ekspektasi	Lulus
17	CRUD Data KPI	Tambah, ubah, hapus, lihat data KPI	Data KPI dapat ditambahkan, diubah, atau dihapus	Sesuai ekspektasi	Lulus

No	Skenario Pengujian	Input dan Aksi	Hasil yang Diharapkan	Hasil Aktual	Status
18	CRUD Vendor Obat	Tambah, ubah, hapus, lihat data vendor obat	Data vendor obat dapat ditambahkan, diubah, atau dihapus	Sesuai ekspektasi	Lulus
19	CRUD Kelas Tarif	Tambah, ubah, hapus, lihat data kelas tarif	Data kelas tarif dapat ditambahkan, diubah, atau dihapus	Sesuai ekspektasi	Lulus
20	CRUD Pelayanan Rumah Sakit	Tambah, ubah, hapus, lihat data pelayanan rumah sakit	Data pelayanan rumah sakit dapat ditambahkan, diubah, atau dihapus	Sesuai ekspektasi	Lulus
21	Melihat Halaman Informasi Jasa Sarana	Buka halaman informasi jasa sarana	Informasi jasa sarana ditampilkan	Sesuai ekspektasi	Lulus
22	Update Persentase Jasa Pelayanan dan Sarana	Update persentase jasa pelayanan dan sarana	Persentase jasa diperbaharui dan disimpan dalam sistem	Sesuai ekspektasi	Lulus
23	Update Persentase Jasa Langsung dan Tidak Langsung	Update persentase jasa langsung dan tidak langsung	Persentase jasa diperbaharui dan disimpan dalam sistem	Sesuai ekspektasi	Lulus
24	<i>Logout</i>	Proses <i>logout</i> dari sistem	Pengguna berhasil <i>logout</i> dan kembali ke halaman <i>login</i>	Sesuai ekspektasi	Lulus

4.4.2 Load Testing

- a. Pendahuluan: Dalam pengujian load testing ini, penulis menggunakan tool K6. K6 adalah alat open-source yang dirancang untuk melakukan pengujian beban (*load testing*) dan kinerja (*performance testing*) pada aplikasi web. Alat ini memungkinkan pengembang untuk mensimulasikan lalu lintas pengguna yang tinggi untuk menguji batas kapasitas dan stabilitas aplikasi.

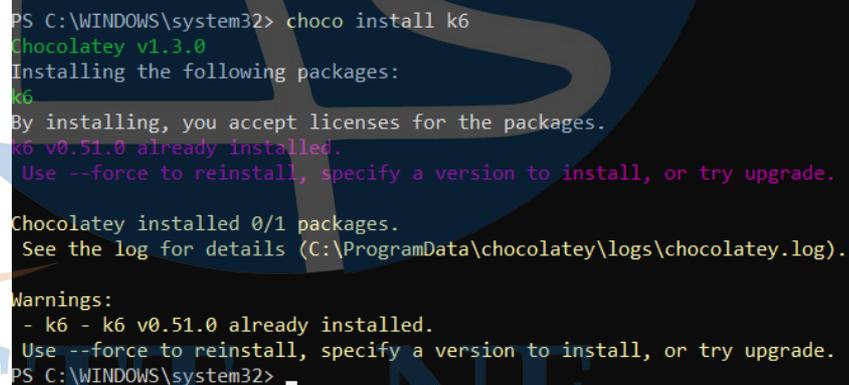
b. Alasan memilih K6:

Penulis memilih K6 karena beberapa alasan utama:

- 1) Open Source dan Gratis: K6 merupakan alat *open-source* yang dapat digunakan tanpa biaya, sangat sesuai untuk kebutuhan pengembangan dan pengujian.
- 2) Mudah Digunakan: K6 memiliki sintaksis yang mudah dimengerti dan ditulis menggunakan JavaScript, bahasa pemrograman yang umum digunakan.
- 3) Ringan dan Efisien: K6 dirancang untuk efisiensi, mampu mensimulasikan ribuan pengguna virtual dengan konsumsi sumber daya yang minimal.
- 4) Integrasi dengan CI/CD: K6 dapat dengan mudah diintegrasikan dengan pipeline CI/CD untuk otomatisasi pengujian kinerja.

c. Langkah-langkah Pengujian dengan K6

- 1) Instalasi K6: Untuk memulai, penulis menggunakan sistem operasi *windows* lalu menginstal K6 di lingkungan pengembangan dengan perintah berikut:



```
PS C:\WINDOWS\system32> choco install k6
Chocolatey v1.3.0
Installing the following packages:
k6
By installing, you accept licenses for the packages.
k6 v0.51.0 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.

Chocolatey installed 0/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Warnings:
- k6 - k6 v0.51.0 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.
PS C:\WINDOWS\system32>
```

Gambar 4.70 Instalasi K6

- 2) Membuat Skrip Pengujian: Penulis membuat skrip pengujian dalam file *load-test.js* yang mendefinisikan skenario pengujian. Berikut adalah contoh skrip dasar:

```

import http from 'k6/http';
import { check, sleep } from 'k6';

export let options = {
  stages: [
    { duration: '1m', target: 3 },
    { duration: '1m', target: 3 },
    { duration: '1m', target: 0 }
  ],
  thresholds: {
    http_req_duration: ['p(95)<500'],
  }
};

export default function () {
  let res = http.get('http://127.0.0.1:8000/shifting-journal', { timeout: '600s' });
  check(res, {
    'status was 200': (r) => r.status === 200,
  });
  sleep(1);
}

```

Gambar 4.71 Skrip Pengujian K6

- 3) Menjalankan Pengujian: Penulis menjalankan pengujian dengan menggunakan perintah berikut di terminal:

```
> k6 run tests/load/load-test.js
```

Gambar 4.72 Menjalankan Pengujian K6

Perintah ini akan mengeksekusi skrip pengujian yang telah dibuat dan menghasilkan output yang mencakup metrik kinerja seperti waktu respons, throughput, dan jumlah kegagalan.

- 4) Seleksi URL untuk Pengujian: Dalam rangka fokus pengujian ini, penulis memilih enam URL yang krusial dalam aplikasi, dimana enam di antaranya adalah:
- /shifting-journal: Menangani pembuatan jurnal keuangan.
 - /piutang-bpjs-tak-tertagih: Mengelola data piutang BPJS yang tidak tertagih, penting untuk keuangan dan audit internal.
 - /ppn: Membuat laporan keuangan.
 - /data-konversi: Menampilkan data pendapatan seluruh pasien.
 - /shifting-remunerasi: Menghitung gaji setiap dokter.
 - /allocation-distribution-remunerasi: Menghitung gaji setiap dokter.

Url-url ini dipilih karena mereka mewakili fitur utama aplikasi dan memiliki intensitas pemrosesan data serta volume permintaan yang tinggi.

d. Deskripsi Skenario Pengujian

Dalam pengujian beban (*Load Testing*) ini, penulis menguji aplikasi dengan jumlah data yang signifikan untuk menilai kemampuan sistem dalam mengelola volume data yang besar dalam kondisi operasional nyata. Penulis menggunakan skenario pengujian dinamis dengan k6 untuk mensimulasikan penggunaan aplikasi di bawah beban yang realistis. Skenario ini melibatkan peningkatan jumlah pengguna secara bertahap, stabilisasi pengguna pada beban moderat, dan pengurangan jumlah pengguna, seperti berikut:

- 1) Peningkatan Pengguna: Jumlah pengguna (*Virtual Users, VUs*) ditingkatkan secara bertahap dari 0 hingga 3 pengguna dalam waktu 1 menit.
- 2) Pemeliharaan Beban: Beban dijaga pada 3 pengguna selama 1 menit untuk menilai performa sistem di bawah beban yang sesuai dengan penggunaan sehari-hari.
- 3) Pengurangan Pengguna: Jumlah pengguna secara bertahap dikurangi kembali ke 0 dalam 1 menit.

Total data yang diuji adalah 17.213 data, yang terdiri dari:

- 1) 9.819 data klaim rawat jalan: Data ini mencakup semua klaim yang dibuat oleh pasien yang menerima perawatan tanpa perlu menginap di rumah sakit.
- 2) 1.347 data klaim rawat inap: Ini mencakup klaim yang dibuat untuk pasien yang memerlukan perawatan intensif atau perawatan yang membutuhkan mereka untuk tinggal di rumah sakit satu malam atau lebih.
- 3) 3.046 data pendapatan rawat inap: Pendapatan dari layanan rawat inap, mencakup semua biaya yang terkait dengan perawatan pasien inap.
- 4) 3.001 data pendapatan rawat jalan: Pendapatan dari layanan rawat jalan, yang meliputi biaya perawatan yang tidak memerlukan pasien untuk menginap.

Setiap endpoint dalam aplikasi diuji di bawah skenario ini untuk memastikan konsistensi dan keandalan hasil. Tujuan pengujian ini adalah untuk memverifikasi bahwa sistem dapat efisien dalam memproses dan menangani permintaan secara stabilitas, bahkan di bawah beban kerja yang sesuai dengan ekspektasi penggunaan nyata.

1) *Endpoint* /shifting-journal: Menangani Pembuatan Jurnal Keuangan

- Aksi yang Dilakukan: Pengujian dilakukan dengan mengirim permintaan *POST* ke endpoint untuk mencatat transaksi keuangan baru ke dalam jurnal.
- Data yang Digunakan: Data transaksi termasuk tanggal, jumlah, dan keterangan dari transaksi yang terjadi.
- Ekspektasi dari Pengujian: *Endpoint* harus mampu menerima data dan mengembalikan respons sukses dengan status HTTP 200 dalam waktu yang wajar, serta memastikan data tersebut tercatat dengan benar di *database*.

2) *Endpoint* /piutang-bpjs-tak-tertagih: Mengelola Data Piutang BPJS yang Tidak Tertagih

- Aksi yang Dilakukan: Pengujian ini melibatkan pengambilan data piutang dari BPJS yang belum terbayar melalui permintaan *GET*.
- Data yang Digunakan: Permintaan tidak memerlukan data khusus selain autentikasi yang valid.
- Ekspektasi dari Pengujian: *Endpoint* ini diharapkan memberikan respons dalam format yang benar dan data yang akurat tentang piutang yang belum terbayar, dengan status HTTP 200.

3) *Endpoint* /ppn: Membuat Laporan Keuangan

- Aksi yang Dilakukan: Membuat laporan PPN dengan mengirimkan permintaan *GET* ke endpoint ini.
- Data yang Digunakan: Periode waktu untuk laporan yang diminta (misalnya, bulan dan tahun).

- Ekspektasi dari Pengujian: *Endpoint* harus mengembalikan laporan keuangan yang detail dan akurat untuk periode yang diminta dalam format PDF atau *Excel*, dengan status HTTP 200.
- 4) *Endpoint* /data-konversi: Menampilkan Data Pendapatan Seluruh Pasien
- Aksi yang Dilakukan: Menampilkan data pendapatan dari semua pasien melalui permintaan *GET*.
 - Data yang Digunakan: Filter tanggal atau kriteria spesifik lainnya untuk mengkustomisasi data yang diambil.
 - Ekspektasi dari Pengujian: Data yang ditampilkan harus lengkap dan akurat, mencerminkan semua pendapatan yang telah tercatat, dan ditampilkan dengan cepat dan efisien.
- 5) *Endpoint* /shifting-remunerasi: Menghitung Gaji Setiap Dokter
- Aksi yang Dilakukan: Menghitung gaji dokter berdasarkan jumlah *shift* yang dilakukan.
 - Data yang Digunakan: Jumlah *shift* yang dilakukan oleh dokter selama periode tertentu.
 - Ekspektasi dari Pengujian: Perhitungan harus akurat dan sesuai dengan aturan remunerasi yang berlaku, dan respons harus diterima dengan status HTTP 200.
- 6) *Endpoint* /allocation-distribution-remunerasi: Menghitung Gaji Setiap Dokter
- Aksi yang Dilakukan: Menghitung gaji berdasarkan sistem alokasi dan distribusi yang telah ditetapkan.
 - Data yang Digunakan: Data alokasi yang mencakup berbagai parameter seperti jumlah pasien yang ditangani, waktu kerja, dan lainnya.
 - Ekspektasi dari Pengujian: Sistem harus dapat melakukan perhitungan secara cepat dan akurat, menghasilkan detail gaji yang bisa dipercaya dan sesuai dengan data input.

e. Tabel Hasil Pengujian

1) *Endpoint* /shifting-journal:

Table 4.3 Tabel Hasil Pengujian /shifting-journal

Pengujian	Status HTTP	Waktu Respons				Permintaan Gagal
		Avg	Min	Med	Max	
1	200	50.07s	9.47s	52.68s	1m26s	0
2	200	22.99s	10.01s	23.65s	31.09s	0
3	200	23.85s	9.73s	24.21s	40.83s	0
4	200	22.67s	9.7s	22.2s	32.67s	0
5	200	22.08s	10.56s	23.04s	28.89s	0
Avg		28.33s	9.89s	29.16s	43.89s	

Untuk visualisasi lebih detail dari distribusi waktu respons ini, lihat gambar di bawah ini:

```

execution: local
script: tests/load/load-test.js
output: -

scenarios: (100.00%) 1 scenario, 3 max VUs, 3m30s max duration (incl. graceful stop):
  * default: Up to 3 looping VUs for 3m0s over 3 stages (gracefulRampDown: 30s, gracefulStop: 30s)

✓ status was 200

checks.....: 100.00% ✓ 8 X 0
data_received.....: 555 kb 2.9 kb/s
data_sent.....: 664 B 4.46238178051853 B/s
http_req_blocked.....: avg=770.71µs min=0s med=642µs max=1.5ms p(90)=1.24ms p(95)=1.37ms
http_req_connecting.....: avg=703.80µs min=0s med=642µs max=1.5ms p(90)=1.24ms p(95)=1.37ms
X http_req_duration.....: avg=50.07s min=9.47s med=52.68s max=1m26s p(90)=1m23s p(95)=1m24s
  { expected_response:true } : avg=50.07s min=9.47s med=52.68s max=1m26s p(90)=1m23s p(95)=1m24s
http_req_failed.....: 0.00% ✓ 0 X 8
http_req_receiving.....: avg=8.31ms min=0s med=1.63ms max=50.2ms p(90)=19.63ms p(95)=24.92ms
http_req_sending.....: avg=3.33µs min=0s med=0s max=96.7µs p(90)=152µs p(95)=229.35µs
http_req_tls_handshaking.....: avg=0s min=0s med=0s max=0s p(90)=0s p(95)=0s
http_req_waiting.....: avg=50.06s min=9.46s med=52.67s max=1m26s p(90)=1m23s p(95)=1m24s
http_reqs.....: 8 0.041345/s X 8
iteration_duration.....: avg=51.1s min=10.48s med=53.69s max=1m27s p(90)=1m24s p(95)=1m25s
iterations.....: 8 0.041345/s
vus.....: 1 min=1 max=3
vus_max.....: 3 min=3 max=3

running (3m13.5s), 0/3 VUs, 8 complete and 1 interrupted iterations
default ✓ [=====] 0/3 VUs 3m0s
ERROR [0195] thresholds on metrics "http_req_duration" have been crossed
  
```

Gambar 4.73 Contoh Hasil Pengujian Pembuatan Jurnal

2) *Endpoint* /piutang-bpjs-tak-tertagih

Table 4.4: Tabel Hasil Pengujian /piutang-bpjs-tak-tertagih

Pengujian	Status HTTP	Waktu Respons				Permintaan Gagal
		Avg	Min	Med	Max	
1	200	381ms	211ms	371ms	732ms	0
2	200	745ms	228ms	734ms	1.49s	0
3	200	347ms	197ms	328ms	802ms	0
4	200	370ms	196ms	350ms	929ms	0
5	200	762ms	217ms	692ms	1.66s	0
Avg		521ms	210ms	495ms	1.12s	

Untuk visualisasi lebih detail dari distribusi waktu respons ini, lihat gambar di bawah ini:

```

output:
scenarios: (100.00%) 1 scenario, 3 max VUs, 3m30s max duration (incl. graceful stop):
 * default: Up to 3 looping VUs for 3m0s over 3 stages (gracefulRampDown: 30s, gracefulStop: 30s)

✓ status was 200

checks.....: 100.00% ✓ 282 X 0
data_received.....: 121 MB 669 kB/s
data_sent.....: 30 kB 164 B/s
http_req_blocked.....: avg=1.12ms min=0s med=995.34µs max=34.28ms p(90)=1.61ms p(95)=2.26ms
http_req_connecting.....: avg=995.63µs min=0s med=661.1µs max=34.28ms p(90)=1.49ms p(95)=1.78ms
X http_req_duration.....: avg=381.35ms min=210.85ms med=370.79ms max=732.36ms p(90)=494.19ms p(95)=544.37ms
 { expected_response:true } : avg=381.35ms min=210.85ms med=370.79ms max=732.36ms p(90)=494.19ms p(95)=544.37ms
http_req_failed.....: 0.00% ✓ 0 X 282
http_req_receiving.....: avg=4.16ms min=0s med=3.1ms max=37.6ms p(90)=7.36ms p(95)=11.41ms
http_req_sending.....: avg=129.64µs min=0s med=0s max=9ms p(90)=510.05µs p(95)=544.17µs
http_req_tls_handshaking.....: avg=0s min=0s med=0s max=0s p(90)=0s p(95)=0s
http_req_waiting.....: avg=377.05ms min=206.7ms med=366.67ms max=732.36ms p(90)=489.54ms p(95)=539.83ms
http_reqs.....: 282 1.59856/s
iteration_duration.....: avg=1.39s min=1.22s med=1.38s max=1.74s p(90)=1.5s p(95)=1.55s
iterations.....: 282 1.59856/s
vus.....: 1 min=1 max=3
vus_max.....: 3 min=3 max=3

running (3m00.8s), 0/3 VUs, 282 complete and 0 interrupted iterations

```

Gambar 4.74 Contoh Hasil Pengujian Piutang BPJS

3) Endpoint /ppn

Table 4.5: Tabel Hasil Pengujian /ppn

Pengujian	Status HTTP	Waktu Respons				Permintaan Gagal
		Avg	Min	Med	Max	
1	200	357ms	209ms	317ms	1.44s	0
2	200	553ms	217ms	576ms	1.32s	0
3	200	245ms	167ms	240ms	599ms	0
4	200	227ms	171ms	219ms	547ms	0
5	200	263ms	170ms	250ms	648ms	0
Avg		329ms	186ms	320	911ms	

Untuk visualisasi lebih detail dari distribusi waktu respons ini, lihat gambar di bawah ini:

```

output:
scenarios: (100.00%) 1 scenario, 3 max VUs, 3m30s max duration (incl. graceful stop):
 * default: Up to 3 looping VUs for 3m0s over 3 stages (gracefulRampDown: 30s, gracefulStop: 30s)

✓ status was 200

checks.....: 100.00% ✓ 287 X 0
data_received.....: 13 MB 69 kB/s
data_sent.....: 24 kB 131 B/s
http_req_blocked.....: avg=1.62ms min=0s med=917.8µs max=42.81ms p(90)=2.31ms p(95)=4.66ms
http_req_connecting.....: avg=1.48ms min=0s med=713.2µs max=42.81ms p(90)=1.85ms p(95)=3.86ms
X http_req_duration.....: avg=357.09ms min=209.6ms med=317.81ms max=1.44s p(90)=471.06ms p(95)=546.21ms
 { expected_response:true } : avg=357.09ms min=209.6ms med=317.81ms max=1.44s p(90)=471.06ms p(95)=546.21ms
http_req_failed.....: 0.00% ✓ 0 X 287
http_req_receiving.....: avg=2.09ms min=0s med=1.51ms max=40.26ms p(90)=3.7ms p(95)=4.78ms
http_req_sending.....: avg=113.31µs min=0s med=0s max=6.09ms p(90)=503.00µs p(95)=535.39µs
http_req_tls_handshaking.....: avg=0s min=0s med=0s max=0s p(90)=0s p(95)=0s
http_req_waiting.....: avg=354.88ms min=208.6ms med=317.35ms max=1.44s p(90)=466.52ms p(95)=544.23ms
http_reqs.....: 287 1.58276/s
iteration_duration.....: avg=1.36s min=1.21s med=1.32s max=2.48s p(90)=1.48s p(95)=1.56s
iterations.....: 287 1.58276/s
vus.....: 1 min=1 max=3
vus_max.....: 3 min=3 max=3

running (3m01.3s), 0/3 VUs, 287 complete and 0 interrupted iterations

```

Gambar 4.75 Contoh Hasil Pengujian PPN

4) *Endpoint /data-konversi*

Table 4.6: Tabel Hasil Pengujian /data-konversi

Pengujian	Status HTTP	Waktu Respons				Permintaan Gagal
		Avg	Min	Med	Max	
1	200	1.01s	579ms	1s	1.72s	0
2	200	1.38s	586ms	1.56s	3.01s	0
3	200	1.18s	610ms	1.19s	3.24s	0
4	200	1s	608ms	1.14s	1.64s	0
5	200	1.04s	617ms	1.2s	1.49s	0
Avg		1.12s	600ms	1.2s	2.22s	

Untuk visualisasi lebih detail dari distribusi waktu respons ini, lihat gambar di bawah ini:

```

output: -
scenarios: (100.00%) 1 scenario, 3 max VUs, 3m30s max duration (incl. graceful stop):
  * default: Up to 3 looping VUs for 3m0s over 3 stages (gracefulRampDown: 30s, gracefulStop: 30s)

v status was 200

checks .....: 100.00% ✓ 195 X 0
data_received.....: 863 MB 4.6 MB/s
data_sent.....: 18 kB 100 B/s
http_req_blocked.....: avg=937.39µs min=0s med=513.29µs max=14.65ms p(90)=1.5ms p(95)=1.9ms
http_req_connecting.....: avg=849.2µs min=0s med=505.49µs max=14.65ms p(90)=1.29ms p(95)=1.67ms
X http_req_duration.....: avg=1.01s min=579.99ms med=1s max=1.72s p(90)=1.51s p(95)=1.55s
  { expected_response:true }.....: avg=1.01s min=579.99ms med=1s max=1.72s p(90)=1.51s p(95)=1.55s
http_req_failed.....: 0.00% ✓ 0 X 195
http_req_receiving.....: avg=11.45ms min=4.90ms med=9.86ms max=39.21ms p(90)=18.43ms p(95)=20.9ms
http_req_sending.....: avg=60.31µs min=0s med=0s max=1.1ms p(90)=112.26µs p(95)=503.71µs
http_req_tls_handshaking.....: avg=0s min=0s med=0s max=0s p(90)=0s p(95)=0s
http_req_waiting.....: avg=998.51ms min=573.75ms med=993.62ms max=1.72s p(90)=1.5s p(95)=1.55s
http_reqs.....: 195 1.076995/s
iteration_duration.....: avg=2.02s min=1.58s med=2.01s max=2.74s p(90)=2.52s p(95)=2.56s
iterations.....: 195 1.076995/s
vus.....: 1 min=1 max=3
vus_max.....: 3 min=3 max=3
  
```

Gambar 4.76 Contoh Hasil Pengujian Data Konversi

5) *Endpoint /shifting-remunerasi*

Table 4.7: Tabel Hasil Pengujian /shifting-remunerasi

Pengujian	Status HTTP	Waktu Respons				Permintaan Gagal
		Avg	Min	Med	Max	
1	200	1.16s	541ms	1.1s	2.92s	0
2	200	1.23s	666ms	1.37s	2.19s	0
3	200	1.16s	545ms	1.42s	1.71s	0
4	200	1.19s	511ms	1.03s	2.86s	0
5	200	1.21s	517ms	1.38s	1.69s	0
Avg		1.19s	556ms	1.26s	2.274s	

Untuk visualisasi lebih detail dari distribusi waktu respons ini, lihat gambar di bawah ini:

```

script: tests/load/load-test.js
output: -

scenarios: (100.00%) 1 scenario, 3 max VUs, 3m30s max duration (incl. graceful stop):
 * default: Up to 3 looping VUs for 3m0s over 3 stages (gracefulRampDown: 30s, gracefulStop: 30s)

√ status was 200

checks.....: 100.00% ✓ 180 X 0
data_received.....: 8.1 MB 45 kB/s
data_sent.....: 18 kB 98 B/s
http_req_blocked.....: avg=1.45ms min=0s med=518.45µs max=21.54ms p(90)=1.88ms p(95)=6.42ms
http_req_connecting.....: avg=1.21ms min=0s med=503.25µs max=21.44ms p(90)=1.5ms p(95)=5.31ms
X http_req_duration.....: avg=1.16s min=541.62ms med=1.1s max=2.92s p(90)=1.8s p(95)=2.03s
 { expected_response:true }...: avg=1.16s min=541.62ms med=1.1s max=2.92s p(90)=1.8s p(95)=2.03s
http_req_failed.....: 0.00% ✓ 0 X 180
http_req_receiving.....: avg=1.58ms min=0s med=1.37ms max=14.83ms p(90)=3.08ms p(95)=3.65ms
http_req_sending.....: avg=46.49µs min=0s med=0s max=1.5ms p(90)=66.65µs p(95)=246.66µs
http_req_tls_handshaking.....: avg=0s min=0s med=0s max=0s p(90)=0s p(95)=0s
http_req_waiting.....: avg=1.16s min=539.07ms med=1.1s max=2.92s p(90)=1.8s p(95)=2.03s
http_reqs.....: 180 0.9932157/s
iteration_duration.....: avg=2.17s min=1.54s med=2.11s max=3.92s p(90)=2.8s p(95)=3.05s
iterations.....: 180 0.9932157/s
vus.....: 1 min=1 max=3
vus_max.....: 3 min=3 max=3

running (3m01.2s), 0/3 VUs, 180 complete and 0 interrupted iterations

```

Gambar 4.77 Contoh Hasil Pengujian Shifting Remunerasi

6) Endpoint /allocation-distribution-remunerasi

Table 4.8: Tabel Hasil Pengujian /allocation-distribution-remunerasi

Pengujian	Status HTTP	Waktu Respons				Permintaan Gagal
		Avg	Min	Med	Max	
1	200	934ms	511ms	876ms	2.11s	0
2	200	772ms	506ms	854ms	1.27s	0
3	200	793ms	508ms	885ms	1.04s	0
4	200	798ms	516ms	909ms	1.23s	0
5	200	3s	763ms	2.1s	7.81s	0
Avg		1.26s	561ms	1.12s	2.69s	

Untuk visualisasi lebih detail dari distribusi waktu respons ini, lihat gambar di bawah ini:

```

output: -

scenarios: (100.00%) 1 scenario, 3 max VUs, 3m30s max duration (incl. graceful stop):
 * default: Up to 3 looping VUs for 3m0s over 3 stages (gracefulRampDown: 30s, gracefulStop: 30s)

√ status was 200

checks.....: 100.00% ✓ 202 X 0
data_received.....: 9.1 MB 50 kB/s
data_sent.....: 23 kB 127 B/s
http_req_blocked.....: avg=924.4µs min=0s med=503.7µs max=17.53ms p(90)=1.62ms p(95)=2.81ms
http_req_connecting.....: avg=870.86µs min=0s med=503.29µs max=17.53ms p(90)=1.54ms p(95)=2.5ms
X http_req_duration.....: avg=934.2ms min=511.85ms med=876.12ms max=2.11s p(90)=1.39s p(95)=1.52s
 { expected_response:true }...: avg=934.2ms min=511.85ms med=876.12ms max=2.11s p(90)=1.39s p(95)=1.52s
http_req_failed.....: 0.00% ✓ 0 X 202
http_req_receiving.....: avg=1.51ms min=0s med=1ms max=16.04ms p(90)=2.98ms p(95)=3.63ms
http_req_sending.....: avg=68.52µs min=0s med=0s max=1.5ms p(90)=172.42µs p(95)=504.15µs
http_req_tls_handshaking.....: avg=0s min=0s med=0s max=0s p(90)=0s p(95)=0s
http_req_waiting.....: avg=932.62ms min=511.04ms med=873.52ms max=2.11s p(90)=1.39s p(95)=1.52s
http_reqs.....: 202 1.1124737/s
iteration_duration.....: avg=1.94s min=1.51s med=1.88s max=3.12s p(90)=2.39s p(95)=2.52s
iterations.....: 202 1.1124737/s
vus.....: 1 min=1 max=3
vus_max.....: 3 min=3 max=3

running (3m01.6s), 0/3 VUs, 202 complete and 0 interrupted iterations

```

Gambar 4.78 Contoh Hasil Pengujian Shifting Remunerasi

f. Analisis Hasil Pengujian *Load*

1) *Endpoint* /shifting-journal

Hasil pengujian menggambarkan performa sistem di bawah beban yang diberikan dengan 3 pengguna virtual. Kelima pengujian dilakukan untuk memahami konsistensi dan keandalan respons sistem terhadap permintaan HTTP.

Rangkuman Hasil Pengujian:

- Status HTTP: Semua pengujian menghasilkan status HTTP 200, yang menunjukkan bahwa sistem berhasil menangani permintaan tanpa mengalami *error*.
- Waktu Respons: Waktu respons rata-rata bervariasi secara signifikan antar pengujian:
 1. Pengujian 1: Waktu respons rata-rata sangat tinggi, mencapai rata-rata 50.07 detik dengan puncak pada 1 menit 26 detik.
 2. Pengujian 2-5: Waktu respons rata-rata menurun menjadi sekitar 22-23 detik, dengan waktu maksimum berkisar antara 28.89 detik hingga 40.83 detik.
- Konsistensi: Terdapat variasi yang signifikan dalam waktu respons minimum, menunjukkan bahwa beberapa permintaan ditangani jauh lebih cepat daripada lainnya. Ini menunjukkan adanya potensi fluktuasi performa dalam sistem yang perlu diinvestigasi lebih lanjut.
- *Threshold*: Hasil pengujian secara konsisten melampaui threshold performa yang telah ditetapkan ($p(95) < 500\text{ms}$), menandakan bahwa sistem tidak memenuhi target performa yang diharapkan.

2) *Endpoint* /piutang-bpjs-tak-tertagih

Analisis Hasil Pengujian:

- Status HTTP: Semua permintaan berhasil dengan status HTTP 200, menunjukkan bahwa *endpoint* tersebut berhasil memproses semua permintaan tanpa *error*.

- Waktu Respons:
 1. Rata-Rata (*Avg*): Waktu respons rata-rata selama lima pengujian adalah 521ms. Ini menunjukkan bahwa *endpoint* dapat menangani permintaan dengan waktu respons yang relatif cepat.
 2. Minimum (*Min*): Waktu respons tercepat yang tercatat adalah 196ms, menunjukkan bahwa sistem dapat responsif di bawah kondisi optimal.
 3. Median (*Med*): Waktu respons median adalah 495ms, menunjukkan responsivitas yang konsisten.
 4. Maksimum (*Max*): Waktu respons maksimum mencapai 1.66 detik pada pengujian kelima, menunjukkan variasi yang signifikan dalam waktu respons yang mungkin disebabkan oleh faktor eksternal seperti beban *server* atau *latency* jaringan.
 - Konsistensi dan Keandalan: Walaupun terjadi variasi dalam waktu respons maksimum, kegagalan permintaan tidak tercatat pada pengujian manapun, menunjukkan keandalan tinggi dari *endpoint* ini dalam memproses permintaan.
 - *Threshold*: Dengan rata-rata waktu respons di bawah satu detik, sistem tampaknya memenuhi ekspektasi performa untuk operasi real-time. Namun, peningkatan pada waktu respons maksimal menunjukkan kebutuhan akan optimisasi lanjutan atau peningkatan sumber daya untuk mengurangi latensi pada permintaan puncak.

3) *Endpoint* /ppn

Analisis Hasil Pengujian:

- Status HTTP: Semua pengujian berhasil dengan status HTTP 200, menunjukkan bahwa semua permintaan diproses dengan sukses tanpa *error*.
- Waktu Respons:

1. Rata-Rata (*Avg*): Waktu respons rata-rata untuk kelima pengujian adalah 329ms. Ini menunjukkan waktu respons yang relatif cepat dalam pemrosesan permintaan.
 2. Minimum (*Min*): Waktu respons tercepat tercatat adalah 167ms, yang mencerminkan performa optimal sistem di bawah kondisi yang ideal.
 3. Median (*Med*): Waktu respons median dari kelima pengujian adalah sekitar 320ms, yang menunjukkan konsistensi dalam waktu respons.
 4. Maksimum (*Max*): Waktu respons tertinggi adalah 1.44 detik, terjadi pada pengujian pertama. Ini menunjukkan adanya kemungkinan beban atau kondisi tertentu yang mempengaruhi performa.
 - Konsistensi dan Keandalan: Meskipun terdapat variasi dalam waktu respons maksimum, tidak ada permintaan yang gagal dalam kelima pengujian, yang menunjukkan keandalan sistem dalam mengelola permintaan.
- 4) *Endpoint* /data-konversi
- Analisis Hasil Pengujian:
- Status HTTP: Semua pengujian berhasil menghasilkan status HTTP 200, menunjukkan bahwa endpoint ini berhasil menangani semua permintaan tanpa mengalami *error*.
 - Waktu Respons:
 1. Rata-Rata (*Avg*): Waktu respons rata-rata keseluruhan adalah 1.12 detik, menunjukkan responsivitas yang relatif stabil meskipun ada variasi antara pengujian.
 2. Minimum (*Min*): Waktu respons tercepat yang tercatat adalah 579ms, mencerminkan kondisi optimal di mana sistem menanggapi dengan cepat.
 3. Median (*Med*): Waktu respons median stabil di sekitar 1.2 detik, mengindikasikan performa yang konsisten di tengah pengujian.

4. Maksimum (*Max*): Waktu respons maksimum mencapai 3.24 detik pada pengujian ketiga, menunjukkan potensi beban puncak yang mempengaruhi performa.

- Performa di Bawah Beban: Meskipun waktu respons rata-rata berada di atas satu detik, adanya waktu respons maksimum yang tinggi menunjukkan kebutuhan untuk mengoptimalkan sistem lebih lanjut, terutama dalam menangani beban yang lebih berat atau data yang lebih kompleks.

5) *Endpoint* /shifting-remunerasi

Analisis Hasil Pengujian:

- Status HTTP: Semua pengujian berhasil dengan status HTTP 200, yang menunjukkan bahwa semua permintaan berhasil diproses tanpa *error*.
- Waktu Respons:
 1. Rata-Rata (*Avg*): Waktu respons rata-rata adalah 1.19 detik, menunjukkan waktu pemrosesan yang moderat untuk permintaan yang dilakukan.
 2. Minimum (*Min*): Waktu respons tercepat yang tercatat adalah 511ms, yang menunjukkan performa optimal sistem dalam kondisi tertentu.
 3. Median (*Med*): Waktu respons median adalah 1.26 detik, menunjukkan waktu respons tipikal untuk operasi ini.
 4. Maksimum (*Max*): Waktu respons maksimal adalah 2.92 detik pada pengujian pertama, menunjukkan adanya kasus-kasus ketika waktu pemrosesan dapat meningkat secara signifikan.
- Konsistensi: Walaupun sistem secara keseluruhan menunjukkan konsistensi dalam menangani permintaan, variasi waktu respons maksimal menunjukkan adanya kasus ketika permintaan mungkin memerlukan waktu lebih lama untuk diproses. Hal ini dapat dipengaruhi oleh kompleksitas perhitungan atau beban sistem saat itu.

- Performa di Bawah Beban: Waktu respons yang lebih lama pada beberapa kasus menunjukkan potensi area untuk peningkatan performa, terutama dalam mengoptimalkan perhitungan atau manajemen sumber daya.

6) *Endpoint* /allocation-distribution-remunerasi

Analisis Hasil Pengujian:

- Status HTTP: Semua pengujian berhasil dengan status HTTP 200, yang menunjukkan bahwa semua permintaan berhasil diproses tanpa mengalami *error*.
- Waktu Respons:
 1. Rata-Rata (Avg): Waktu respons rata-rata adalah 1.26 detik, menunjukkan waktu pemrosesan yang moderat dengan variasi signifikan antar pengujian.
 2. Minimum (Min): Waktu respons tercepat yang tercatat adalah 506ms, mencerminkan kondisi optimal di mana sistem menanggapi dengan cepat.
 3. Median (Med): Waktu respons *median* adalah 1.12 detik, mengindikasikan waktu respons tipikal untuk operasi ini.
 4. Maksimum (Max): Waktu respons maksimal mencapai 7.81 detik pada pengujian kelima, menunjukkan adanya kondisi yang sangat mempengaruhi performa.
- Konsistensi: Pengujian ke-5 menunjukkan waktu respons yang jauh lebih tinggi dibandingkan dengan empat pengujian lainnya. Variabilitas ini menunjukkan bahwa sistem mungkin mengalami kesulitan dengan beban yang lebih berat atau permintaan yang lebih kompleks.

g. Kesimpulan

Dari hasil pengujian load pada berbagai *endpoint*, sistem secara keseluruhan menunjukkan kemampuan yang baik dalam menangani permintaan dengan berbagai beban operasional. Berikut adalah rangkuman kritis dan rekomendasi untuk masing-masing *endpoint* berdasarkan hasil pengujian:

- 1) *Endpoint* /shifting-journal: Waktu respons yang sangat tinggi pada pengujian pertama dan variasi signifikan antar pengujian menunjukkan perlunya optimisasi dan stabilisasi performa lebih lanjut. Dianjurkan untuk menyelidiki dan mengatasi fluktuasi performa ini, terutama dalam kondisi beban yang tinggi.
- 2) *Endpoint* /piutang-bpjs-tak-tertagih: Memiliki waktu respons yang sangat baik dengan performa yang konsisten. Namun, peningkatan waktu respons maksimal menunjukkan perlunya penyelidikan lebih lanjut untuk menangani kasus-kasus beban puncak.
- 3) *Endpoint* /ppn: Menunjukkan waktu respons yang cepat dan performa yang stabil. Keandalan ini penting untuk operasi yang memerlukan respons cepat, namun diperlukan penyelidikan lebih lanjut untuk menangani lonjakan waktu respons.
- 4) *Endpoint* /data-konversi: Menunjukkan variasi dalam waktu respons, terutama pada pengujian ketiga dengan waktu respons maksimal yang tinggi. Optimalisasi pada pengolahan data dan penggunaan sumber daya diperlukan untuk meningkatkan performa.
- 5) *Endpoint* /shifting-remunerasi: Menunjukkan waktu respons yang moderat dengan adanya kasus di mana waktu pemrosesan meningkat secara signifikan. Pengujian menunjukkan kebutuhan akan optimisasi dalam perhitungan dan manajemen sumber daya.
- 6) *Endpoint* /allocation-distribution-remunerasi: Pengujian menunjukkan variabilitas yang signifikan, khususnya dengan peningkatan waktu respons yang ekstrem pada pengujian terakhir. Perlunya peningkatan performa dan stabilitas sistem sangat kritis untuk mengelola kasus beban yang berat.

Berikut adalah analisis detail berdasarkan rata-rata waktu respons untuk setiap *endpoint*, dengan menyoroti mana yang memiliki waktu respons paling cepat dan paling lama:

- 1) *Endpoint* Paling Cepat: *Endpoint* /ppn dengan rata-rata waktu respons 329ms. Ini menunjukkan performa yang sangat efisien dan responsif, menjadi *benchmark* untuk efisiensi sistem.
- 2) *Endpoint* Paling Lambat: *Endpoint* /shifting-journal dengan rata-rata waktu respons 28.33 detik. Ini mengindikasikan bahwa *endpoint* ini memerlukan investigasi lebih lanjut dan optimasi untuk mengurangi latensi.

Detail rata-rata waktu load test untuk setiap *endpoint* adalah sebagai berikut:

- 1) /shifting-journal: Rata-rata waktu respons 28.33 detik
- 2) /piutang-bpjs-tak-tertagih: Rata-rata waktu respons 521ms
- 3) /ppn: Rata-rata waktu respons 329ms
- 4) /data-konversi: Rata-rata waktu respons 1.12 detik
- 5) /shifting-remunerasi: Rata-rata waktu respons 1.19 detik
- 6) /allocation-distribution-remunerasi: Rata-rata waktu respons 1.26 detik

Dari data ini, penulis dapat melihat bahwa sementara sebagian besar *endpoint* mengelola permintaan dengan kecepatan yang memadai, /shifting-journal menonjol sebagai area yang memerlukan perhatian serius untuk peningkatan. Upaya harus difokuskan pada optimisasi proses dan infrastruktur untuk *endpoint* ini. Selanjutnya, review mendalam terhadap arsitektur dan implementasi sistem disarankan untuk meningkatkan performa keseluruhan.

4.5 Wawancara Dengan Pengguna

a. Deskripsi Responden

Responden dalam penelitian ini adalah seorang konsultan keuangan profesional yang bekerja dengan rumah sakit XYZ. Konsultan ini bertanggung jawab atas pengelolaan dan supervisi operasi keuangan rumah sakit menggunakan aplikasi yang dikembangkan. Responden telah memenuhi kriteria berikut:

- 1) Konsultan keuangan dengan pengalaman luas dalam manajemen keuangan sektor kesehatan.
- 2) Terlibat secara langsung dalam implementasi dan penggunaan aplikasi sebagai pengguna utama.
- 3) Memberikan persetujuan untuk berpartisipasi dalam penelitian ini dan menyediakan *feedback* yang komprehensif.

b. Metode Pengumpulan Data

Data dikumpulkan melalui serangkaian wawancara mendalam yang dilakukan untuk mendapatkan insight tentang pengalaman penggunaan sistem. Detail sesi wawancara adalah sebagai berikut:

- 1) Tanggal: 22 Juni 2024
- 2) Waktu: 14:00 - 15:00
- 3) Metode: Virtual melalui Zoom

Pertanyaan dan Respons Wawancara

- 1) Pertanyaan: Bagaimana Anda menggambarkan pengalaman Anda secara keseluruhan dengan menggunakan aplikasi ini? *Respons* “Secara keseluruhan, saya merasa aplikasi ini sangat membantu. *Interface* penggunanya *intuitif*, yang memudahkan saya dalam menavigasi berbagai fungsi. Namun, ada beberapa laporan yang memuat waktu loading yang agak lama, terutama saat mengakses data transaksi yang besar.”.
- 2) Pertanyaan: Apa tantangan terbesar yang Anda hadapi saat menggunakan sistem? *Respons* “Tantangan terbesar adalah saat harus melakukan rekonsiliasi data secara manual ketika terjadi kesalahan *input*. Sistem ini membantu dalam banyak hal, tetapi masih membutuhkan beberapa penyesuaian untuk otomatisasi proses rekonsiliasi yang lebih efektif.”.
- 3) Pertanyaan: Dari aspek mana Anda merasa sistem ini paling menguntungkan? *Respons* “Keuntungan terbesar adalah dalam pengelolaan data keuangan dan penggajian. Sistem memungkinkan saya untuk memonitor aliran kas secara real-time dan membuat

laporan bulanan dengan cepat. Ini sangat mengurangi waktu yang biasanya saya butuhkan untuk tugas-tugas ini.”.

- 4) Pertanyaan: Adakah fitur yang Anda pikir perlu ditambahkan ke dalam sistem untuk meningkatkan kinerja? *Respons* “Saya sangat menyarankan penambahan fitur prediksi keuangan yang bisa membantu dalam merencanakan anggaran dan pengeluaran di masa depan. Fitur ini bisa sangat bermanfaat untuk perencanaan keuangan jangka panjang.”.

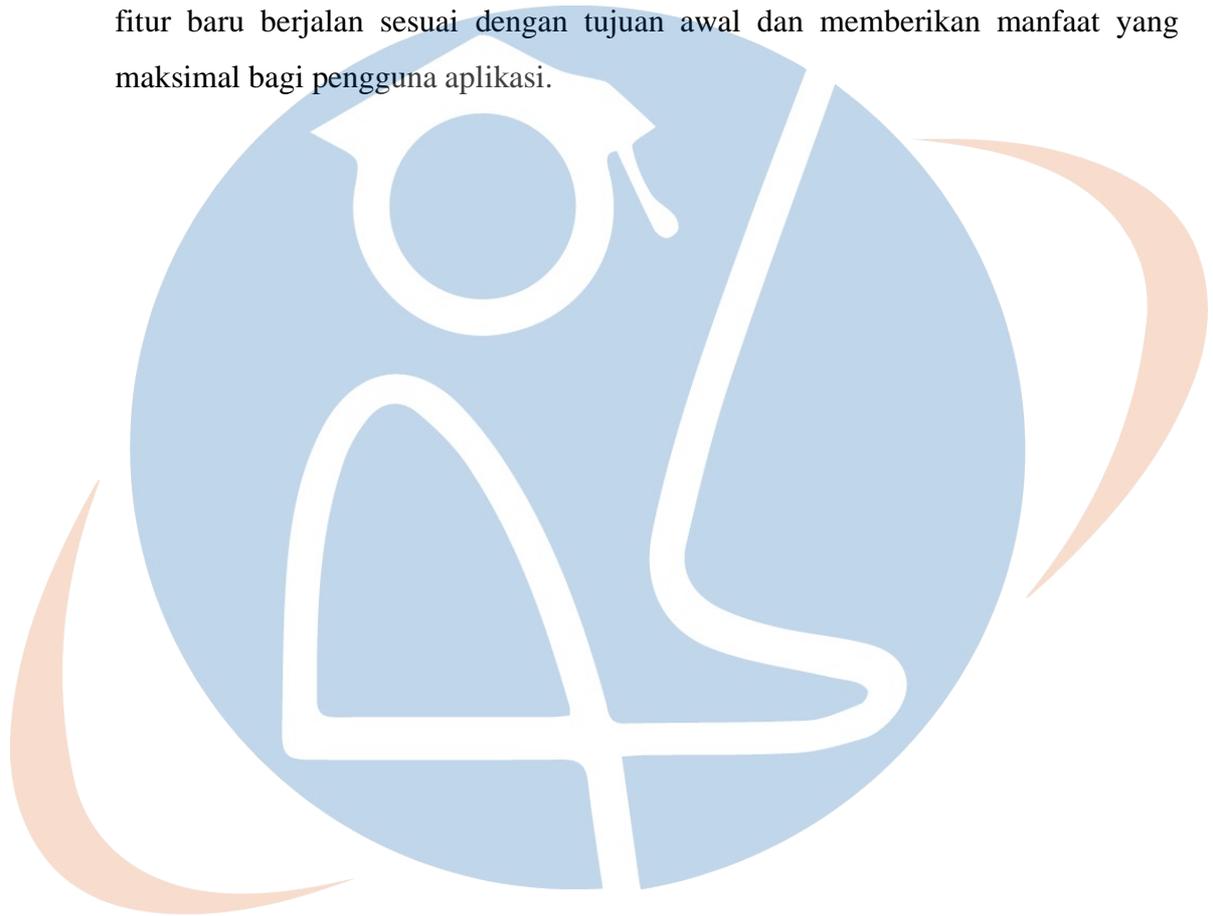
4.6 Fitur Inovatif dan Rencana Pengembangan Selanjutnya

Dalam pengembangan aplikasi ini, terdapat beberapa fitur utama yang telah berhasil diimplementasikan, serta rencana untuk inovasi dan peningkatan fitur yang akan dilakukan di masa mendatang. Berikut adalah beberapa fitur yang direncanakan:

- a. Pemilihan Metode Penggajian Dokter: Aplikasi ini telah memungkinkan penggajian dokter berdasarkan tarif INA-CBG BPJS. Sebagai pengembangan selanjutnya, aplikasi ini akan ditambahkan kemampuan untuk rumah sakit menentukan metode penggajian dokter, apakah mengikuti tarif BPJS, tarif internal rumah sakit, atau kombinasi keduanya. Fitur ini bertujuan untuk memberikan fleksibilitas kepada manajemen rumah sakit dalam menyesuaikan kompensasi dokter dengan kebijakan internal dan kondisi keuangan rumah sakit.
- b. Integrasi KPI Dokter: Fitur saat ini sudah mencakup dasar untuk mengatur dan menginput KPI setiap dokter. Rencana pengembangan selanjutnya adalah memperluas fitur ini agar rumah sakit dapat memilih menggunakan KPI sebagai acuan utama dalam menentukan gaji dokter. Fitur ini diharapkan bisa mendukung peningkatan kualitas pelayanan medis serta kinerja dokter secara keseluruhan.
- c. Manajemen Vendor Obat: Fitur CRUD (*Create, Read, Update, Delete*) untuk manajemen vendor obat telah dibuat. Ke depannya, fitur ini akan dikembangkan lebih lanjut untuk memastikan bahwa pembelian obat oleh rumah sakit dilakukan sesuai dengan ketentuan dan kontrak yang telah

disepakati dengan vendor yang telah diverifikasi dan dipilih secara resmi. Hal ini bertujuan untuk memastikan kualitas dan efisiensi pengadaan obat.

Setiap fitur inovatif ini direncanakan untuk diintegrasikan dengan sistem yang ada, dengan memperhatikan skalabilitas, keamanan, dan kemudahan penggunaan. Evaluasi akan dilakukan secara berkala untuk memastikan bahwa implementasi fitur baru berjalan sesuai dengan tujuan awal dan memberikan manfaat yang maksimal bagi pengguna aplikasi.



STT - NF

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini berhasil menggunakan metode pengembangan Spiral untuk menciptakan aplikasi yang menjembatani BPJS dan sistem kompensasi dokter yang efisien, transparan, dan akuntabel. Implementasi metode Spiral terbukti efektif dalam mengelola iterasi yang berulang dengan evaluasi dan pengelolaan risiko yang efisien pada setiap fase, mendukung pengembangan aplikasi yang kompleks dengan memastikan peningkatan berkelanjutan berdasarkan umpan balik dari pengguna awal. Aplikasi yang dikembangkan berhasil mengintegrasikan data dari BPJS dengan sistem internal rumah sakit untuk mengelola kompensasi dokter, menyediakan sistem yang efisien, transparan, dan akuntabel yang memenuhi kebutuhan rumah sakit sekaligus memastikan kepatuhan terhadap kebijakan BPJS. Selain itu, aplikasi meningkatkan transparansi dan akuntabilitas dalam pengelolaan kompensasi dokter dengan audit trail yang otomatis dan tercatat, memudahkan penelusuran dan verifikasi, dan memastikan bahwa setiap elemen dalam sistem dapat diandalkan dan dipertanggungjawabkan, menjawab secara langsung pertanyaan penelitian yang diajukan.

5.2 Saran

Saran untuk Penelitian Selanjutnya:

1. Pengembangan Fitur Prediktif: Untuk meningkatkan fungsionalitas aplikasi, penelitian selanjutnya bisa mengeksplorasi pengintegrasian fitur-fitur prediktif yang menggunakan data historis untuk meramalkan kebutuhan kompensasi dokter di masa mendatang, membantu rumah sakit dalam perencanaan keuangan yang lebih baik.
2. Integrasi dengan Sistem Keuangan Lain: Mengintegrasikan aplikasi ini dengan sistem keuangan lain di rumah sakit dapat memberikan gambaran yang lebih holistik tentang keuangan rumah sakit, meningkatkan efisiensi pengelolaan keuangan secara keseluruhan.
3. Studi Multi-situs: Melakukan pengujian aplikasi di lebih banyak rumah sakit akan membantu memahami keefektifan aplikasi ini dalam berbagai

setting operasional dan membantu menyempurnakan fitur sesuai dengan kebutuhan yang lebih luas.

4. Evaluasi Dampak Jangka Panjang: Menilai dampak jangka panjang dari penggunaan aplikasi ini pada operasional rumah sakit dan kepuasan dokter bisa memberikan wawasan lebih lanjut tentang keberlanjutan dan skalabilitas solusi ini.

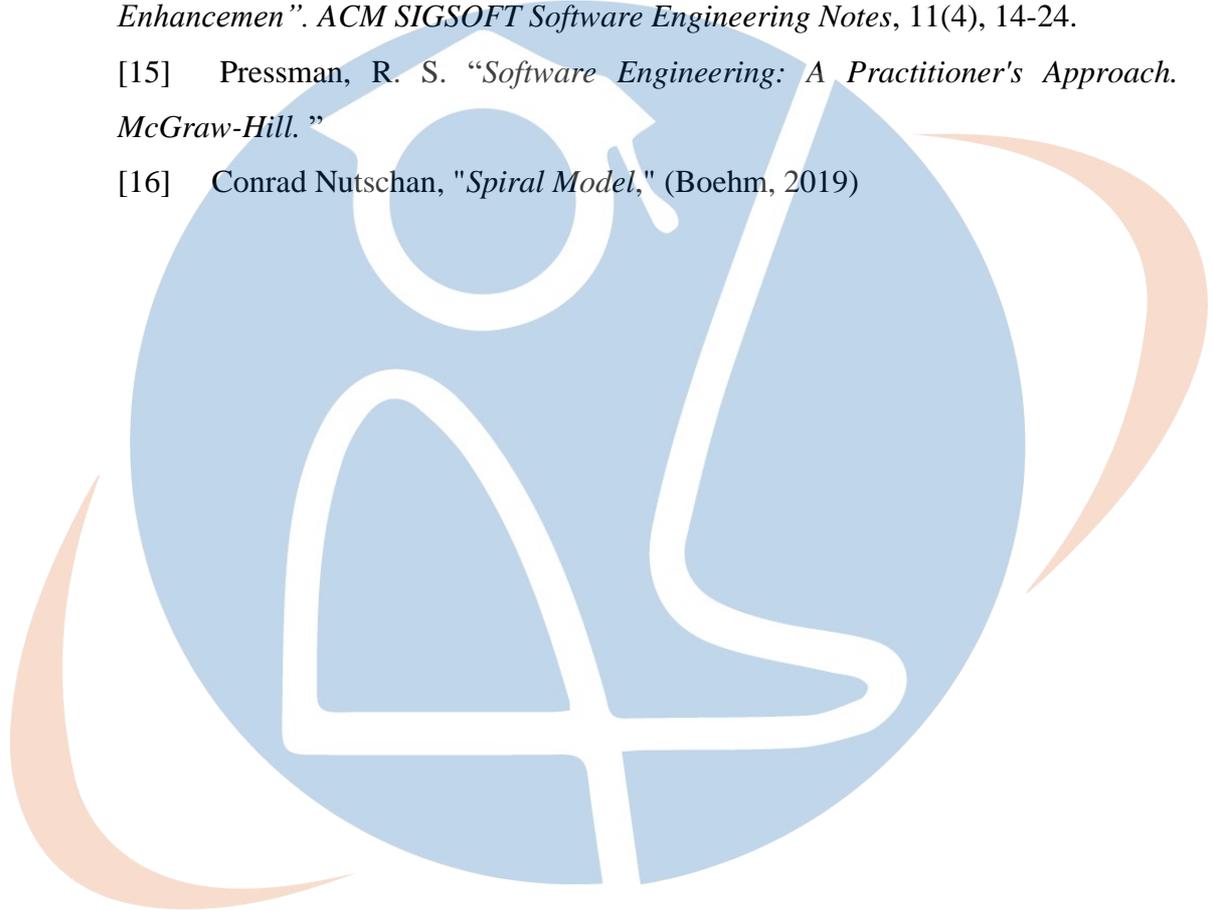


STT - NF

DAFTAR PUSTAKA

- [1] N. Medika, S. Wahyuni, I. Saputra, and F. Hanum, "Komparasi Biaya Riil Rumah Sakit dengan Tarif INA-CBGs pada Penderita Thalasemia di Rumah Sakit Umum Zainoel Abidin Banda Aceh Tahun 2018," *Kedokteran Nanggroe Medika*, vol. 2, no. 3, 2019.
- [2] I. Ananta, "Penerapan Pola Pembayaran INA-CBGs BPJS Kesehatan dalam Tinjauan Regulasi dan Implementasi," in *Seminar Nasional dan Call for Papers "Tantangan Pengembangan Ilmu Akuntansi, Inklusi Keuangan, dan Kontribusinya Terhadap Pembangunan Ekonomi Berkelanjutan,"* [Online]. Available: <http://www.depkes.go.id/resources/down>
- [3] F. Idris *et al.*, "Sistem Pembayaran Mixed Method INA-CBGs dan Global Budget di Rumah Sakit: Tahap 1 Uji Coba Mixed Method INA-CBGs-Global Budget di Indonesia Health Service Payment System in Hospital Mixed INA-CBGs and Global Budget."
- [4] J. Sosial, K. Di, R. Sakit, K. Pelalawan, and R. Amalia, "Analisis Penerapan Indonesia Case Based Groups (Ina-Cbg's) Dalam Pelayanan Badan Penyelenggara."
- [5] "Peraturan Menteri Kesehatan Republik Indonesia Nomor 27 Tahun 2014 Tentang Petunjuk Teknis Sistem Indonesian Case Base Groups (INA-CBGs)."
- [6] J. Kesehatan, N. Statistik Jkn, D. Jaminan, S. Nasional, and B. Kesehatan, "Fakta dan Data Capaian Program Fakta dan Data Capaian Program Jaminan Kesehatan Nasional Fakta dan Data Capaian Program Jaminan Kesehatan Nasional," 2015.
- [7] "Peraturan Menteri Kesehatan Nomor 28 Tahun 2014 Tentang Pedoman Pelaksanaan Program Jaminan Kesehatan Nasional."
- [8] "PERATURAN MENTERI KESEHATAN REPUBLIK INDONESIA."
- [9] "Peraturan Menteri Kesehatan Republik Indonesia Nomor 63 Tahun 2016 Tentang Pedoman Pengelolaan Keuangan Badan Layanan Umum di Lingkungan Kementerian Kesehatan." [Online]. Available: www.peraturan.go.id
- [10] "Peraturan Menteri Dalam Negeri Nomor 61 Tahun 2007 tentang Pedoman Teknis Pengelolaan Keuangan Badan Layanan Umum Daerah."

- [11] T. M. dr. Hani, *Model pembagian jasa pelayanan di rumah sakit dengan metode konversi dan proporsi*, 1st ed. Yogyakarta: Deepublish, 2019.
- [12] “25 PMK No. 12 ttg Pola Tarif BLU RS”.
- [13] “Peraturan Menteri Kesehatan Republik Indonesia Nomor 85 Tahun 2015 Tentang Pola Tarif Nasional Rumah Sakit.”
- [14] Boehm, B. W. “*A Spiral Model of Software Development and Enhancemen*”. *ACM SIGSOFT Software Engineering Notes*, 11(4), 14-24.
- [15] Pressman, R. S. “*Software Engineering: A Practitioner's Approach. McGraw-Hill.*”
- [16] Conrad Nutschan, "*Spiral Model*," (Boehm, 2019)



STT - NF