



**TUGAS AKHIR - KS141501**

**OPTIMASI PENJADWALAN STAF RUMAH SAKIT  
DENGAN MENGGUNAKAN ALGORITMA SIMULATED  
ANNEALING HYPER-HEURISTIC (STUDI KASUS :  
RSIA KENDANGSARI MERR SURABAYA)**

***HOSPITAL STAFF SCHEDULING OPTIMIZATION  
USING SIMULATED ANNEALING HYPER-HEURISTIC  
(CASE STUDY: RSIA KENDANGSARI MERR  
SURABAYA)***

**INDRIARTI KUSUMANITA  
NRP 5214100178**

**Dosen Pembimbing  
Ahmad Muklason, S.Kom, M.Sc, Ph. D**

**DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018**

**TUGAS AKHIR - KS141501**

**OPTIMASI PENJADWALAN STAF RUMAH SAKIT  
DENGAN MENGGUNAKAN ALGORITMA  
SIMULATED ANNEALING HYPER-HEURISTIC  
(STUDI KASUS : RSIA KENDANGSARI MERR  
SURABAYA)**

**INDRIARTI KUSUMANITA  
NRP 5214100178**

**Dosen Pembimbing  
Ahmad Muklason, S.Kom, M.Sc, Ph.D**

**DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018**

**FINAL PROJECT - KS141501**

**HOSPITAL STAFF SCHEDULING  
OPTIMIZATION USING SIMULATED  
ANNEALING HYPER-HEURISTIC (CASE  
STUDY: RSIA KENDANGSARI MERR  
SURABAYA)**

**INDRIARTI KUSUMANITA  
NRP 05214100178**

**Supervisor**

**Ahmad Muklason, S.Kom, M.Sc, Ph. D**

**INFORMATION SYSTEMS DEPARTMENT  
Faculty of Information and Communication Technology  
Institute Technology of Sepuluh Nopember  
Surabaya 2018**



**LEMBAR PENGESAHAN**

**OPTIMASI PENJADWALAN STAF RUMAH SAKIT  
DENGAN MENGGUNAKAN ALGORITMA  
SIMULATED ANNEALING HYPER-HEURISTIC  
(STUDI KASUS: RSIA KENDANGSARI MERR  
SURABAYA)**

**TUGAS AKHIR**

Disusun Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Departemen Sistem Informasi  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember

Oleh:

**INDRIARTI KUSUMANITA**

NRP. 5214100178

Surabaya, Januari 2018

**Plh Kepala  
Departemen Sistem Informasi**



**Edwin Riksakomara, S.Kom, M.T.**  
NIP.196907252003121001

**LEMBAR PERSETUJUAN**

**OPTIMASI PENJADWALAN STAF RUMAH SAKIT  
DENGAN MENGGUNAKAN ALGORITMA  
SIMULATED ANNEALING HYPER-HEURISTIC  
(STUDI KASUS: RSIA KENDANGSARI SURABAYA)**

**TUGAS AKHIR**

Disusun Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Departemen Sistem Informasi  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember

Oleh :

**INDRIARTI KUSUMANITA**

NRP. 5214100178

Disetujui Tim Penguji : Tanggal Ujian : 15 Januari 2018  
Periode Wisuda: Maret 2018

**Ahmad Muklason S. Kom, M.Sc, Ph.D**

(Pembimbing I)

**Wiwik Anggraeni, S.Si, M.Kom**

(Penguji I)

**Faizal Mahananto, S.Kom, M.Eng, Ph.D**

(Penguji II)

**OPTIMASI PENJADWALAN STAF RUMAH SAKIT  
DENGAN MENGGUNAKAN ALGORITMA  
SIMULATED ANNEALING HYPER-HEURISTIC  
(STUDI KASUS: RSIA KENDANGSARI MERR  
SURABAYA)**

**Nama Mahasiswa** : **INDRIARTI KUSUMANITA**  
**NRP** : **5214100178**  
**Jurusan** : **SISTEM INFORMASI FTIK-ITS**  
**Dosen Pembimbing 1** : **Ahmad Muklason, Ph. D**

**ABSTRAK**

*Rumah sakit Ibu dan Anak Kendangsari merupakan salah satu rumah sakit swasta yang berada di wilayah Surabaya. Salah satu hal yang menjadi perhatian dalam manajemen rumah sakit adalah penjadwalan staf rumah sakit yang optimal guna memenuhi kebutuhan pasien selama 24 jam penuh. Permasalahan ini menjadi kompleks karena terdapat berbagai batasan yang harus dipenuhi yang terdiri atas hard constraints dan soft constraints. Selama ini rumah sakit masih menggunakan metode penjadwalan secara manual yang memakan banyak waktu.*

*Maka dari itu dibutuhkan suatu penjadwalan otomatis yang membutuhkan waktu lebih sedikit untuk mencapai penjadwalan yang optimal. Penjadwalan yang optimal akan dilakukan dengan memaksimalkan nilai Jain's Fairness Index dari libur yang didapat oleh setiap staf pada setiap unit. Unit-unit yang menjadi ruang lingkup adalah Unit Farmasi, Unit Ruang Bayi dan NICU, Unit SIM Rekam Medis dan Registrasi, Unit Gizi, Unit Rawat Jalan dan IGD serta Kamar Operasi. Untuk dapat menyelesaikan permasalahan ini maka penulis akan menggunakan metode simulated annealing. Simulated Annealing merupakan sebuah metode dengan analogi seperti proses annealing pada logam. Algoritma simulated annealing akan fokus terhadap pencarian solusi optimal dengan*

*menerima adanya solusi yang lebih buruk ketika melakukan iterasi. Dalam pengerjaan metode ini, nantinya akan dilakukan bersama dengan metode hyper-heuristic. Metode hyper-heuristic berjalan pada lingkungan heuristic sehingga menghasilkan solusi yang lebih umum dibandingkan dengan penggunaan metode metaheuristics.*

*Hasil dari penelitian ini adalah meningkatnya nilai Jain's Fairness Index untuk setiap unit ketika dilakukan optimasi dengan menggunakan algoritma Simulated Annealing Hyper-Heuristic. Untuk Unit Farmasi, nilai JFI hasil manual meningkat dari 0.92 menjadi 0.93. Unit Ruang Bayi dan NICU hasilnya meningkat dari 0.89 menjadi 0.91. Unit SIM Rekam Medis dan Registrasi meningkat dari 0.90 menjadi 0.97. Unit Gizi meningkat dari 0.85 menjadi 0.86. Unit Rawat Jalan dan IGD meningkat dari 0.91 menjadi 0.93. Dan terakhir Unit Kamar Operasi meningkat dari 0.80 menjadi 0.97. Penggunaan algoritma Simulated Annealing Hyper-Heuristic diharapkan dapat memberikan manfaat dengan membuat jadwal yang optimal serta waktu yang dihabiskan untuk penjadwalan menjadi lebih cepat.*

***Kata kunci: Optimasi, Penjadwalan Staf Rumah Sakit, Algoritma Simulated Annealing, Hyper-Heuristic***



**HOSPITAL STAFF SCHEDULING OPTIMIZATION  
USING SIMULATED ANNEALING HYPER-  
HEURISTIC (CASE STUDY: RSIA KENDANGSARI  
MERR SURABAYA)**

**Name** : INDRIARTI KUSUMANITA  
**NRP** : 5214100178  
**Departement** : INFORMATION SYSTEM FTIK-ITS  
**Supervisor 1** : Ahmad Muklason, Ph. D

**ABSTRACT**

*RSIA Kendangsari is one of the private hospital located in Surabaya area. One of the main concerns in hospital management is the optimal hospital staff scheduling to meet the needs of patients for 24 hours. This problem becomes complex because there are various constraints that must be met consisting of hard constraints and soft constraints. During this time the hospital still uses manual scheduling which takes a lot of time. Therefore it requires an automated scheduling that takes less time to achieve optimal scheduling. Optimal scheduling will be done by maximizing the value of Jain's Fairness Index from the holidays earned by each staff on each unit. The units that are included in this research are Pharmacy Unit, Infant and NICU Unit, Registration and Medical Record Information System Unit, Nutrition Unit, Outpatient and Emergency Unit and Surgery Room.*

*To be able to solve this problem then the author will use simulated annealing method. Simulated Annealing is a method with analogies such as annealing processes in metals. The simulated annealing algorithm will focus on finding the optimal solution by accepting a worse solution when iterating. In this method work, it will be done together with hyper-heuristic method. Hyper-heuristic methods run on a heuristic environment resulting in a more general solution than the metaheuristics method.*

*The result of this research is the increase of Jain's Fairness Index value for each unit when optimized by using Simulated Annealing Hyper-Heuristic algorithm. For Pharmacy Unit, the JFI value of manual scheduling increased from 0.92 to 0.93. The Infant and NICU Unit resulted in an increase from 0.89 to 0.91. Registration and Medical Record Information System Unit increased from 0.90 to 0.97. The Nutrition Unit increased from 0.85 to 0.86. Outpatient and Emergency Unit increased from 0.91 to 0.93. And Surgery Room Unit increased from 0.80 to 0.97. The use of the Simulated Annealing Hyper-Heuristic algorithm is expected to provide benefits by creating an optimal schedule and time spent on scheduling to be faster.*

***Keywords: Optimization, Staff Scheduling, Simulated Annealing, Hyper-Heuristic***

## **KATA PENGANTAR**

Alhamdulillah atas karunia, rahmat, barakah, dan jalan yang telah diberikan Allah SWT selama ini sehingga penulis mendapatkan kelancaran dalam menyelesaikan tugas akhir dengan judul:

### **OPTIMASI PENJADWALAN PERAWAT DENGAN MENGUNAKAN SIMULATED ANNEALING HYPER- HEURISTIC (STUDI KASUS: RSIA KENDANGSARI MERR SURABAYA)**

Terima kasih atas pihak-pihak yang telah mendukung, memberikan saran, motivasi, semangat, dan bantuan baik materi maupun spiritual demi tercapainya tujuan pembuatan tugas akhir ini. Secara khusus penulis akan menyampaikan ucapan terima kasih yang sedalam-dalamnya kepada:

1. Allah SWT yang telah memberi segala rahmat dan hidayah-Nya untuk dapat menyelesaikan tugas belajar menjalani perkuliahan di Departemen Sistem Informasi ITS dan telah memberikan kemudahan, kelancaran dan kesehatan selama menyelesaikan Tugas Akhir ini.
2. Kedua orang tua, kakak dan seluruh keluarga yang selalu hadir dan senantiasa mendoakan dan memberikan kasih sayang serta semangat tiada henti untuk menyelesaikan Tugas Akhir ini.
3. Bapak Dr. Ir. Aris Tjahyanto, M.Kom, selaku Ketua Departemen Sistem Informasi, yang telah menyediakan fasilitas terbaik untuk kebutuhan penelitian mahasiswa.
4. Bapak Ahmad Muklason, S.Kom, M.Sc, Ph. D, selaku dosen pembimbing yang telah meluangkan banyak waktunya untuk membimbing, mengarahkan, memberikan ilmu, petunjuk dan motivasi untuk kelancaran Tugas Akhir ini.
5. Ibu Wiwik Anggraeni, S.Si, M.Kom selaku dosen wali penulis yang telah memberikan saran serta motivasi selama menempuh pendidikan di Departemen Sistem Informasi ITS.

6. Teman-teman sesama pejuang satu topik penjadwalan RSIA Kendangsari, Zuli Maulidati, Rizka Pordella, Fachrur Zafrinda, dan Fata Hirzi yang telah bersama-sama menyatukan pikiran dan pendapat demi terselesainya Tugas Akhir ini.
7. Teman-teman terbaik penulis di YSC, Cindy, Dhira, Rara, Patty, Fia, Opor, Tatan, Ninda, Risha, Roy, Depe, Nody, Septy, Rachel, dan Yunis yang selalu menemani dan mensupport penulis sejak awal perkuliahan.
8. Alden Delfian Wattimena yang telah membantu penulis dan teman-teman lain dalam menyelesaikan Tugas Akhir.

Penyusunan laporan ini masih jauh dari sempurna, untuk itu saya menerima adanya kritik dan saran yang membangun untuk perbaikan di masa mendatang. Semoga buku tugas akhir ini dapat memberikan manfaat bagi pembaca.

Surabaya, Januari 2018

Penulis

## DAFTAR ISI

LEMBAR PENGESAHAN.....	i
LEMBAR PERSETUJUAN.....	ii
ABSTRAK.....	iii
ABSTRACT.....	v
KATA PENGANTAR .....	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR .....	xiii
DAFTAR TABEL.....	xvii
BAB I PENDAHULUAN .....	1
1.1. Latar Belakang.....	1
1.2. Perumusan Masalah .....	3
1.3. Batasan Pengerjaan Tugas Akhir.....	3
1.4. Tujuan Tugas Akhir.....	3
1.5. Manfaat Tugas Akhir.....	4
1.6. Relevansi.....	4
BAB II TINJAUAN PUSTAKA.....	7
2.1. Studi Sebelumnya .....	7
2.2. Dasar Teori .....	11
2.2.1. RSIA Kendangsari .....	11
2.2.2. Penjadwalan .....	11
2.2.3. Nurse Rostering Problem.....	12
2.2.4. Simulated Annealing.....	15
2.2.5. Hyper-Heuristic.....	17
2.2.6. Simulated Annealing Hyper-Heuristic.....	18
2.2.7. Jain's Fairness Index .....	20
2.2.8. Hill Climbing .....	20
BAB III METODOLOGI PENELITIAN.....	23
3.1. Metodologi Penelitian.....	23
3.2. Uraian Metodologi.....	24
3.2.1. Identifikasi Permasalahan di Rumah Sakit Ibu dan Anak Kendangsari .....	24
3.2.2. Studi Literatur .....	24
3.2.3. Pengumpulan Data .....	25
3.2.4. Pemodelan Permasalahan.....	26

3.2.5.	Implementasi Algoritma Simulated Annealing Hyper-Heuristic .....	27
3.2.6.	Eksperimen .....	27
3.2.7.	Analisis Hasil dan Penarikan Kesimpulan .....	27
3.2.8.	Penyusunan Laporan Tugas Akhir .....	28
<b>BAB IV PERANCANGAN.....</b>		<b>31</b>
4.1.	Hasil Pengumpulan Data.....	31
4.2.	Aturan Penjadwalan Rumah Sakit .....	32
4.2.1.	Tipe Shift dan Tipe Skill pada Masing-Masing Unit.....	33
4.3.	Pembuatan Model untuk Masing-Masing Unit .....	37
4.3.1.	Asumsi dan Notasi.....	38
4.3.2.	Variabel Keputusan .....	38
4.3.3.	Pemodelan Unit Instalasi Farmasi .....	39
4.3.6	Pemodelan Unit Ruang Bayi dan NICU.....	41
4.3.7	Pemodelan Unit SIM Rekam Medis dan Registrasi .....	44
4.3.8	Pemodelan Unit Instalasi Gizi .....	46
4.3.9	Pemodelan Unit Instalasi Rawat Jalan dan IGD .....	49
4.3.10	Pemodelan Unit Kamar Operasi.....	52
4.3.11	Soft Constraint.....	54
4.3.12	Perumusan Soft Constraint.....	54
4.3.13	Fungsi Tujuan.....	56
4.3.14	Perumusan Fungsi Tujuan .....	56
4.4	Pemodelan Simulated Annealing Hyper-Heuristic .....	56
5.	<b>BAB V IMPLEMENTASI.....</b>	<b>61</b>
5.1	Lingkungan Uji Coba.....	61
5.2	Membuat Otomasi Penjadwalan .....	62
5.2.1	Memasukkan Data Set ke dalam Program JAVA .....	62
5.2.2	Memasukkan Jadwal Baru.....	63
5.2.3	Melakukan Perhitungan Jain's Fairness Index .....	73
5.2.4	Melakukan Pengecekan Constraint .....	75
5.3.	Membuat Optimasi Penjadwalan .....	77
5.3.1	Mendefinisikan Low Level Heuristic .....	77

5.3.2	Menerapkan Algoritma Simulated Annealing Hyper-Heuristic.....	80
6	BAB VI HASIL DAN PEMBAHASAN .....	83
6.1	Hasil Perhitungan Nilai Jain's Fairness Index Jadwal Manual .....	83
6.1.1	Jadwal Manual Unit Instalasi Farmasi .....	83
6.1.2	Jadwal Manual Unit Ruang Bayi dan NICU ...	84
6.1.3	Jadwal Manual Unit SIM Rekam Medis dan Registrasi.....	84
6.1.4	Jadwal Manual Unit Instalasi Gizi .....	85
6.1.5	Jadwal Manual Unit Instalasi Rawat Jalan dan IGD .....	86
6.1.6	Jadwal Manual Unit Kamar Operasi .....	87
6.2	Hasil Otomasi Penjadwalan dan Nilai Jain's Fairness Index dari Otomasi .....	87
6.2.1	Hasil Otomasi Unit Instalasi Farmasi .....	87
6.2.2	Hasil Otomasi Unit Ruang Bayi dan NICU ....	89
6.2.3	Hasil Otomasi Unit SIM Rekam Medis dan Registrasi.....	90
6.2.4	Hasil Otomasi Unit Instalasi Gizi .....	92
6.2.5	Hasil Otomasi Unit Instalasi Rawat Jalan dan IGD .....	94
6.2.6	Hasil Otomasi Unit Kamar Operasi .....	95
6.3	Hasil Optimasi dengan Menggunakan Algoritma Simulated Annealing Hyper-Heuristic.....	96
6.3.1	Hasil Penjadwalan dengan Optimasi.....	96
6.3.2	Diagram Trajectory .....	109
6.2.2	Perbandingan Algoritma Simulated Annealing Hyper-Heuristic dengan Hill Climbing.....	113
6.2.3	Perbandingan Statistik Algoritma Simulated Annealing Hyper-Heuristic dengan Algoritma Hill Climbing .....	119
6.2.4	Pemetaan Box Plot Algoritma Simulated Annealing Hyper-Heuristic dengan Algoritma Hill Climbing .....	123

6.3	Validasi Hasil Penjadwalan Otomatis dan Hasil Optimasi dengan Hard Constraints dari Masing-Masing Unit.....	129
6.4	Perbandingan Ketidaksesuaian Jadwal Manual, Otomasi dengan Optimasi.....	135
6.6	Perbandingan Nilai JFI Manual, Otomasi dengan Optimasi Jadwal Existing.....	137
6.6.1	Unit Farmasi.....	137
6.6.2	Unit Ruang Bayi dan NICU.....	137
6.6.3	Unit SIM Rekam Medis dan Registrasi.....	137
6.6.4	Unit Gizi.....	138
6.6.5	Unit Rawat Jalan dan IGD.....	138
6.6.6	Unit Kamar Operasi.....	138
7	BAB VII KESIMPULAN DAN SARAN.....	141
7.1	Kesimpulan.....	141
7.2	Saran.....	143
7.	DAFTAR PUSTAKA.....	145
8	BIODATA PENULIS.....	148
9	LAMPIRAN A – HASIL WAWANCARA.....	149
8.	LAMPIRAN B – JADWAL MANUAL BULAN NOVEMBER 2017.....	154
9.	LAMPIRAN C – JADWAL HASIL OPTIMASI BULAN DESEMBER 2017.....	166



## DAFTAR GAMBAR

Gambar 2.1 Algoritma Simulated Annealing Hyper-Heuristic .....	xv
Gambar 3.1 Metodologi Penelitian .....	23
Gambar 4.1 Algoritma Simulated Annnaling Hyper-Heuristic [22] .....	57
Gambar 6.1 Jadwal Manual November 2017 Unit Farmasi ...	83
Gambar 6.2 Jadwal Manual November 2017 Unit Ruang Bayi dan NICU .....	84
Gambar 6.3 Jadwal Manual November 2017 Unit SIM Rekam Medis dan Registrasi .....	85
Gambar 6.4 Jadwal Manual November 2017 Unit Instalasi Gizi .....	86
Gambar 6.5 Jadwal Manual November 2017 Unit Instalasi Rawat Jalan dan IGD .....	86
Gambar 6.6 Jadwal Manual November 2017 Unit Kamar Operasi .....	87
Gambar 6.7 Hasil Otomasi Unit Farmasi .....	88
Gambar 6.8 Hasil Otomasi Unit Ruang Bayi dan NICU .....	89
Gambar 6.9 Hasil Otomasi Unit SIM RM dan Registrasi .....	91
Gambar 6.10 Hasil Otomasi Unit Gizi .....	92
Gambar 6.11 Hasil Otomasi Unit Rawat Jalan dan IGD.....	94
Gambar 6.12 Hasil Otomasi Unit Kamar Operasi.....	95
Gambar 6.13 Hasil Jadwal Optimasi Unit Farmasi .....	97
Gambar 6.14 Hasil Jadwal Optimasi Unit Ruang Bayi dan NICU .....	99
Gambar 6.15 Hasil Jadwal Optimasi Unit SIM Rekam Medis .....	101
Gambar 6.16 Hasil Jadwal Optimasi Unit Gizi .....	103
Gambar 6.17 Hasil Jadwal Optimasi Unit Rawat Jalan dan IGD .....	106
Gambar 6.18 Hasil Jadwal Optimasi Unit Kamar Operasi ..	108
Gambar 6.19 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Farmasi .....	110

Gambar 6.20 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Ruang Bayi dan NICU .....	111
Gambar 6.21 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit SIM Rekam Medis dan Registrasi .....	111
Gambar 6.22 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Gizi .....	112
Gambar 6.23 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Rawat Jalan dan IGD .....	113
Gambar 6.24 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Kamar Operasi .....	113
Gambar 6.25 Perbandingan Algoritma pada Unit Farmasi ..	114
Gambar 6.26 Perbandingan Algoritma pada Unit Ruang Bayi dan NICU.....	115
Gambar 6.27 Perbandingan Algoritma pada Unit SIM Rekam Medis dan Registrasi .....	116
Gambar 6.28 Perbandingan Algoritma pada Unit Rawat Jalan dan IGD .....	117
Gambar 6.29 Perbandingan Algoritma Unit Gizi .....	118
Gambar 6.30 Perbandingan Algoritma Unit Kamar Operasi	119
Gambar 6.31 Box Plot Unit Farmasi .....	124
Gambar 6.32 Box Plot Unit Ruang Bayi dan NICU.....	125
Gambar 6.33 Box Plot Unit SIM Rekam Medis dan Registrasi .....	126
Gambar 6.34 Box Plot Unit Gizi .....	127
Gambar 6.35 Box Plot Unit Rawat Jalan dan IGD.....	128
Gambar 6.36 Box Plot Unit Kamar Operasi.....	129

## DAFTAR KODE PROGRAM

Kode Program 2.1 Pseudo-Code Hill Climbing .....	21
Kode Program 5.1 Memasukkan Data Set .....	62
Kode Program 5.2 Menambah Array Untuk Jadwal Baru .....	63
Kode Program 5.3 Pola Jadwal Kepala Unit Farmasi dan Petugas Gudang Obat .....	63
Kode Program 5.4 Pola Jadwal Staff Senior Farmasi .....	64
Kode Program 5.5 Pola Jadwal Staff Unit Farmasi.....	65
Kode Program 5.6 Pola Jadwal Kepala Unit Ruang Bayi dan NICU .....	66
Kode Program 5.7 Pola Jadwal Staf Unit Ruang Bayi dan NICU .....	66
Kode Program 5.8 Pola Jadwal Staf Rekam Medis dan Registrasi (1) .....	67
Kode Program 5.9 Pola Jadwal Staf Rekam Medis dan Registrasi (2) .....	67
Kode Program 5.10 Pola Jadwal Ahli Gizi .....	68
Kode Program 5.11 Pola Jadwal Chef .....	68
Kode Program 5.12 Pola Jadwal Staf nomor 6 .....	68
Kode Program 5.13 Pola Jadwal Helper .....	69
Kode Program 5.14 Pola Staf Gudang Makanan .....	69
Kode Program 5.15 Pola Staf Penyaji.....	69
Kode Program 5.16 Pola Staf Nomor 16 - 19 .....	70
Kode Program 5.17 Pola Staf nomor 17 dan 18.....	70
Kode Program 5.18 Pola Middle Shift .....	71
Kode Program 5.19 Pola Staf Unit Rawat Jalan dan IGD.....	71
Kode Program 5.20 Pola Staf Kamar Operasi.....	72
Kode Program 5.21 Pengisian Middle Shift.....	72
Kode Program 5.22 Pengisian Shift Hari Minggu Unit Kamar Operasi .....	73
Kode Program 5.23 Menghitung Nilai Jain's Fairness Index	74
Kode Program 5.24 Mengecek Constraint Unit Farmasi .....	76
Kode Program 5.25 Low-Level Heuristic Move.....	78
Kode Program 5.26 Low Level Heuristic Swap.....	79
Kode Program 5.27 Algoritma Simulated Annealing Hyper- Heuristic .....	80



## DAFTAR TABEL

Tabel 1.1 Research Roadmap Laboratorium RDIB .....	5
Tabel 2.1 Studi Literatur1 .....	7
Tabel 2.2 Studi Literatur 2 .....	8
Tabel 2.3 Studi Literatur 3 .....	9
Tabel 2.4 Studi Literatur 4 .....	10
Tabel 4.1 Tipe Shift RSIA Kendangsari .....	33
Tabel 4.2 Tipe Skill Unit Instalasi Farmasi.....	34
Tabel 4.3 Tipe Skill Unit Rawat Jalan dan IGD.....	34
Tabel 4.4 Tipe Skill Unit SIM Rekam Medis dan Registrasi	35
Tabel 4.5 Tipe Skill Unit Instalasi Gizi.....	35
Tabel 4.6 Tipe Skill Unit Rawat Jalan dan IGD.....	36
Tabel 4.7 Tipe Skill Unit Kamar Operasi.....	37
Tabel 5.1Perangkat Keras dan Lunak Yang Digunakan .....	61
Tabel 5.2 Pengujian Nilai Cooling Schedule pada Unit Farmasi .....	81
Tabel 6.1 Perhitungan Libur Hasil Otomasi Unit Farmasi.....	88
Tabel 6.2 Perhitungan Libur Hasil Otomasi Unit Ruang Bayi dan NICU .....	89
Tabel 6.3 Perhitungan Libur Hasil Otomasi Unit SIM RM dan Registrasi.....	91
Tabel 6.4 Perhitungan Libur Hasil Otomasi Unit Gizi.....	93
Tabel 6.5 Perhitungan Libur Hasil Otomasi Unit Rawat Jalan dan IGD.....	94
Tabel 6.6 Perhitungan Libur Hasil Otomasi Kamar Operasi .	96
Tabel 6.7 Hasil Perhitungan Libur Optimasi Unit Farmasi....	97
Tabel 6.8 Jumlah Shift Tiap Staf Hasil Optimasi Unit Farmasi .....	98
Tabel 6.9 Hasil Perhitungan Libur Optimasi Unit Ruang Bayi dan NICU .....	99
Tabel 6.10 Jumlah Shift Tiap Staf Hasil Optimasi Unit Ruang Bayi dan NICU.....	100
Tabel 6.11 Hasil Perhitungan Libur Optimasi Unit SIM Rekam Medis dan Registrasi .....	101

Tabel 6.12 Jumlah Shift Tiap Staf Hasil Optimasi Unit SIM Rekam Medis dan Registrasi .....	102
Tabel 6.13 Hasil Perhitungan Libur Optimasi Unit Gizi .....	104
Tabel 6.14 Jumlah Shift Tiap Staf Hasil Optimasi Gizi .....	104
Tabel 6.15 Hasil Perhitungan Libur Optimasi Unit Rawat Jalan dan IGD .....	106
Tabel 6.16 Jumlah Shif Tiap Staf Hasil Optimasi Unit Rawat Jalan dan IGD .....	107
Tabel 6.17 Hasil Perhitungan Libur Optimasi Unit Kamar Operasi.....	108
Tabel 6.18 Jumlah Shift Tiap Staf Hasil Optimasi Uit Kamar Operasi.....	109
Tabel 6.19 Statistik Perbandingan Algoritma Unit Farmasi.	120
Tabel 6.20 Statistik Perbandingan Algoritma Unit Ruang Bayi dan NICU.....	120
Tabel 6.21 Statistik Perbandingan Algoritma Unit SIM Rekam Medis dan Registrasi .....	121
Tabel 6.22 Statistik Perbandingan Algoritma Unit Gizi.....	121
Tabel 6.23 Statistik Perbandingan Algoritma Unit Rawat Jalan dan IGD .....	122
Tabel 6.24 Statistik Perbandingan Algoritma Unit Kamar Operasi.....	123
Tabel 6.25 Validasi Hasil Otomasi dan Hasil Optimasi dengan Hard Constraint .....	129
Tabel 6.26 Perbandingan Persentase Pelanggaran Jadwal Manual, Otomasi dan Optimasi Terhadap Pola M-M-LM-L135	
Tabel 6.27 Perbandingan JFI Unit Farmasi .....	137
Tabel 6.28 Perbandingan JFI Ruang Bayi dan NICU.....	137
Tabel 6.29 Perbandingan JFI SIM Rekam Medis dan Registrasi .....	137
Tabel 6.30 Perbandingan JFI Unit Gizi .....	138
Tabel 6.31 Perbandingan JFI Unit Rawat Jalan dan IGD.....	138
Tabel 6.32 Perbandingan JFI Unit Kamar Operasi.....	138

# **BAB I**

## **PENDAHULUAN**

Pada bab ini, akan dijelaskan mengenai latar belakang masalah, perumusan masalah, batasan masalah, tujuan tugas akhir, dan manfaat tugas akhir, serta relevansi penelitian tugas akhir dengan bidang keilmuan sistem informasi.

### **1.1. Latar Belakang**

RSIA Kendangsari merupakan salah satu Rumah Sakit Surabaya yang dirancang untuk berfokus pada kebutuhan ibu dan anak. RSIA Kendangsari dibentuk atas satu kesatuan visi dan misi yang sama oleh beberapa dokter spesialis kebidanan dan kandungan dalam upaya meningkatkan “kualitas kesehatan reproduksi kaum perempuan khususnya ibu hamil dan janin yang dikandungnya.” Atas dasar keinginan tersebut maka dibentuklah PT. Sandra Buana Medika yang mendapatkan izin untuk membuat rumah sakit swasta [1]. Sebagai Rumah Sakit yang memiliki fasilitas rawat inap tentunya diperlukan penjadwalan perawat yang dapat memenuhi kebutuhan pasien rumah sakit selama 24 jam penuh.

Adanya jumlah staf rumah sakit yang tidak sesuai dengan kebutuhan, akan membuat produktivitas rumah sakit menjadi menurun. Selain itu menurut beberapa penelitian, ketidakpuasan staf rumah sakit dalam bekerja sangat dipengaruhi oleh keseimbangan waktu bekerja dalam *shift* yang dijalankan [2]. Penjadwalan yang memenuhi keinginan staf tentunya juga akan berdampak positif terhadap lingkungan kerja.

Penyelesaian permasalahan penjadwalan perawat sampai dengan awal tahun 2000an masih banyak yang menggunakan penjadwalan secara manual atau biasa disebut dengan ‘*self-scheduling*’ [3]. Scheduling dengan menggunakan tenaga manual tentunya sangat menguras tenaga dan belum tentu pula hasil yang didapatkan bisa optimal. Maka dari itu diperlukan

suatu metode teknik otomasi yang optimal dalam membuat penjadwalan yang dapat memenuhi batasan-batasan yang ada seperti regulasi rumah sakit, jam kerja dan keinginan perawat dalam menjalankan jadwal tersebut. Terdapat berbagai macam pendekatan untuk menyelesaikan permasalahan tersebut antara lain dengan *mathematical programming*, metode *artificial intelligence*, metode *heuristic* dan penjadwalan *metaheuristic* [3].

Salah satu penelitian mengenai pemecahan masalah ini dengan menggunakan metode *simulated annealing* pernah dilakukan oleh Hidayatullah dan Santosa (2014) [4]. Dari hasil penelitian tersebut ditemukan bahwa dengan penggunaan metode *simulated annealing* dapat dihasilkan suatu penjadwalan yang jauh lebih cepat dibandingkan dengan penggunaan metode eksak. Namun hasil penjadwalan hanya mendekati nilai optimal saja. Penjadwalan dengan penggunaan metode eksak masih lebih baik.

Untuk memperoleh hasil yang lebih optimal kembali, maka metode *simulated annealing* ini akan dicoba digabungkan dengan konsep *hyper-heuristic*. Terdapat penelitian terdahulu yaitu *A Simulated Annealing Hyper-Heuristic Methodology for Flexible Decision Support* [5] yang menggabungkan metode ini dan menghasilkan penjadwalan yang lebih optimal dibandingkan dengan menggunakan pendekatan *meta-heuristic* lainnya.

Dengan berbasis dua penelitian sebelumnya, maka dalam tugas akhir ini yaitu penjadwalan perawat RSIA Kendangsari Surabaya akan dioptimalkan dengan menggunakan algoritma *simulated annealing hyper-heuristic*.



## 1.2. Perumusan Masalah

Berdasarkan uraian latar belakang yang telah dijelaskan, maka rumusan masalah dari tugas akhir ini adalah :

1. Bagaimana model matematis untuk mengatasi permasalahan penjadwalan staf di RSIA Kendangsari?
2. Bagaimana pengoptimalan penjadwalan staf di RSIA Kendangsari dengan menggunakan algoritma *simulated annealing hyper-heuristic*?
3. Bagaimana perbandingan hasil penjadwalan staf yang sudah diterapkan oleh RSIA Kendangsari saat ini dengan penjadwalan algoritma *simulated annealing hyper-heuristic*?

## 1.3. Batasan Pengerjaan Tugas Akhir

Batasan masalah dari penelitian ini adalah:

1. Tugas Akhir ini dilakukan di Rumah Sakit Ibu dan Anak Kendangsari dan berlaku untuk menjadwalkan seluruh staf di rumah sakit tersebut.
2. Data yang digunakan sebagai *hard constraint* adalah regulasi kerja staf rumah sakit yang ada di RSIA Kendangsari.
3. Data yang digunakan sebagai *soft constraint* adalah nilai *Jain's Fairness Index* dari libur setiap staf.

## 1.4. Tujuan Tugas Akhir

Tujuan dilakukannya tugas akhir ini antara lain:

1. Membuat model matematis yang sesuai dengan permasalahan penjadwalan staf di RSIA Kendangsari.
2. Implementasi algoritma *simulated annealing hyper-heuristic* yang dapat mengoptimalkan penjadwalan staf di RSIA Kendangsari.
3. Membandingkan hasil penjadwalan yang sudah diterapkan RSIA Kendangsari saat ini dengan hasil penjadwalan staf dengan menggunakan algoritma *simulated annealing hyper-heuristic*.

### 1.5. Manfaat Tugas Akhir

Manfaat dari dilakukannya penelitian Tugas Akhir ini adalah:

1. Manfaat Teoritis  
Adanya penelitian ini diharapkan dapat menambah khasanah keilmuan dalam bidang sistem informasi khususnya mengenai implementasi algoritma *Simulated Annealing Hyper-Heuristic* untuk membuat penjadwalan yang optimal.
2. Manfaat Praktis
  - a. Bagi Pihak Rumah Sakit Ibu dan Anak Kendangsari  
Tugas akhir ini diharapkan dapat membantu Rumah Sakit Ibu dan Anak Kendagsari dalam menyusun penjadwalan staf rumah sakit yang optimal sehingga dapat menghemat waktu dalam membuat penjadwalan optimal yang sesuai dengan keinginan staf.
  - b. Bagi Penelitian Berikutnya  
Tugas akhir ini diharapkan dapat memberikan motivasi bagi peneliti lain untuk melakukan pengembangan terhadap studi mengenai penjadwalan staf rumah sakit dengan menggunakan algoritma atau pendekatan lainnya.

### 1.6. Relevansi

Topik tugas akhir ini mengenai optimasi penjadwalan perawat merupakan pemenuhan dari pokok penelitian *Business Analytic* bagian *Optimizations* seperti pada tabel 1. Topik tersebut sesuai dengan bidang ilmu riset operasi yang menjadi cakupan *research roadmap* pada laboratorium Rekamaya Data dan Inteligensi Bisnis (RDIB).

Tugas akhir ini layak dijadikan sebagai tugas akhir pada tingkat S1, karena tugas ini dapat membantu RSIA Kendangsari dalam menentukan penjadwalan yang optimal dengan hanya membutuhkan waktu yang lebih sedikit dibandingkan dengan sebelumnya. Dalam bidang keilmuan, penelitian ini mendukung pengembangan ilmu dalam optimasi terutama optimasi dalam bidang penjadwalan staf rumah sakit.

**Tabel 1.1 Research Roadmap Laboratorium RDIB**

<b>COMPUTERIZED DECISION SUPPORT</b>	<b>DATA MANAGEMENT</b>	<b>BUSINESS ANALYTICS</b>
<ul style="list-style-type: none"> <li>- Decision Support System</li> <li>- System Modelling and Analysis</li> </ul>	<ul style="list-style-type: none"> <li>- Database and Database Management Systems (DBMS)</li> <li>- Extraction, Transformation and Load System (ETL)</li> <li>- Data Warehouse (DW)</li> <li>- Real-time DW Data Mart</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Optimization</b></li> <li>- Data/Web/Text Mining</li> <li>- Web Analytic</li> <li>- Peramalan</li> </ul>
<b>KNOWLEDGE MANAGEMENT</b>	<b>INTELLIGENT SYSTEMS</b>	
<ul style="list-style-type: none"> <li>- Knowledge Management System</li> <li>- Expert Locating System</li> <li>- Ontology</li> </ul>	<ul style="list-style-type: none"> <li>- Expert System</li> <li>- Artificial Neural Network</li> <li>- Fuzzy Logic</li> <li>- Genetic Algorithm</li> <li>- Intelligent Agent, Automated Decision System</li> </ul>	

*(Halaman ini sengaja dikosongkan)*

## BAB II TINJAUAN PUSTAKA

Bab ini berisikan tinjauan pustaka yang akan digunakan dalam penelitian tugas akhir ini, mencakup studi sebelumnya, dasar teori dan metode yang digunakan.

### 2.1. Studi Sebelumnya

Dalam pengerjaan tugas akhir ini terdapat beberapa penelitian yang terkait untuk bisa dijadikan sebagai bahan studi literatur untuk menyelesaikan tugas akhir ini. Beberapa penelitian sebelumnya yang dijadikan acuan dalam pengerjaan tugas akhir disajikan dalam tabel berikut.

**Tabel 2.1 Studi Literatur1**

<b>Judul Paper</b>	A SIMULATED ANNEALING HYPER-HEURISTIC UNIVERSITY COURSE TIMETABLING PROBLEM [6]
<b>Penulis; Tahun</b>	Bai, R., Burke, E.K., Kendall, G. and McCollum, B., ; 2006
<b>Deskripsi Umum Penelitian</b>	Penelitian ini dilakukan untuk melakukan optimasi terhadap permasalahan penjadwalan kuliah dengan menggunakan metode <i>Simulated Annealing Hyper-heuristic</i> . Paper ini menjabarkan mengenai tahap-tahap atau <i>framework</i> dalam menerapkan metode <i>Simulated Annealing Hyper-heuristic</i> .
<b>Keterkaitan Penelitian</b>	Penerapan metode yang sama yaitu <i>Simulated Annealing Hyper-heuristic</i> dapat menjadikan literatur ini sebagai acuan pengerjaan tugas akhir. Terdapat pula langkah dalam berupa <i>pseudo-code</i> yang dapat mempermudah pengerjaan. Namun literatur ini tidak memberitahukan hasil akhir dari penerapan metode tersebut.

Tabel 2.2 Studi Literatur 2

<b>Judul Paper</b>	A SIMULATED ANNEALING HYPER-HEURISTIC METHODOLOGY FOR FLEXIBLE DECISION SUPPORT [5]
<b>Penulis; Tahun</b>	Bai, R., Blazewicz, J., Burke, E. K., Kendall, G., & McCollum, B. ; 2007
<b>Deskripsi Umum Penelitian</b>	<p>Penelitian ini menjabarkan secara mendalam metodologi <i>Simulated Annealing Hyper-Heuristic</i> sebagai salah satu alternatif dalam menentukan keputusan yang lebih fleksibel. Paper ini menawarkan 3 fitur baru untuk meningkatkan penerapan metode <i>hyper-heuristic</i> yang sudah ada sebelumnya, yaitu kriteria penerimaan <i>simulated annealing</i>, strategi pemilihan stokastik heuristic dan <i>short-term memory</i>. Dalam pembuktiannya, dilakukan penerapan algoritma tersebut untuk 3 permasalahan yang sangat berbeda dengan menggunakan parameter yang sama. Penerapan dilakukan untuk permasalahan penjadwalan perawat, penjadwalan jadwal di universitas dan <i>bin packing</i>. Dengan menerapkan metode ini, hasil akhirnya menjadi lebih baik dibandingkan dengan menggunakan metode <i>tabu search hyper-heuristic</i> maupun metode dengan pendekatan <i>meta-heuristic</i>.</p>

<b>Keterkaitan Penelitian</b>	Literatur ini dapat digunakan sebagai acuan dalam melakukan pengerjaan tugas akhir karena memiliki penjelasan yang dalam terkait metode <i>simulated annealing hyper-heuristic</i> . Selain itu paper ini juga menerapkan metode pada kasus yang sama yaitu <i>nurse rostering</i> .
-------------------------------	--

**Tabel 2.3 Studi Literatur 3**

<b>Judul Paper</b>	A SIMULATED ANNEALING BASED HYPER HEURISTIC FOR DETERMINING SHIPPER SIZES FOR STORAGE AND TRANSPORTATION [6]
<b>Penulis; Tahun</b>	Dowland, K.A., Soubeiga, E. and Burke, E. ; 2007
<b>Deskripsi Umum Penelitian</b>	Penelitian ini menerapkan metode <i>simulated annealing hyper-heuristic</i> untuk menentukan ukuran dalam penyimpanan dan transportasi ketika melakukan sebuah pengiriman. Dalam melakukan perhitungan, paper ini memperkenalkan metode <i>tabu search-hyper heuristic</i> yang diterapkan bersamaan dengan <i>simulated annealing-hyper heuristic</i> . Namun ditemukan kesulitan dalam menentukan solusi mana yang paling baik untuk diterapkan. Perlu ditemukan adanya suatu ukuran atau <i>measure</i> lagi agar penerapan metode ini menjadi lebih optimal.

<b>Keterkaitan Penelitian</b>	Literatur ini dapat digunakan sebagai acuan dalam melakukan pengerjaan tugas akhir karena memiliki metode yang sama yaitu <i>simulated annealing hyper-heuristic</i> .
-------------------------------	--

**Tabel 2.4 Studi Literatur 4**

<b>Judul Paper</b>	OPTIMASI NURSE SCHEDULING PROBLEM (STUDI KASUS : RSUD DR. SUTOMO SURABAYA) [4]
<b>Penulis; Tahun</b>	Hidayatullah, Aditya P. Santosa Budi ; 2014
<b>Deskripsi Umum Penelitian</b>	Penelitian ini menjelaskan mengenai penerapan metode meta-heuristik <i>Simulated Annealing</i> untuk melakukan optimasi penjadwalan perawat dengan menggunakan aplikasi LINGO. Paper ini membandingkan hasil perhitungan antara metode meta-heuristik dengan metode eksak. Hasil perhitungan menunjukkan bahwa perhitungan dengan menggunakan metode <i>Simulated Annealing</i> hanya membutuhkan waktu yang lebih sedikit dibanding menggunakan metode eksak, namun hasil yang didapat hanya mendekati nilai optimal saja. Sedangkan perhitungan dengan menggunakan metode eksak dapat mencapai nilai yang optimal namun membutuhkan waktu yang relatif lebih lama apabila diterapkan untuk skala yang lebih besar.



<p><b>Keterkaitan Penelitian</b></p>	<p>Literatur ini dapat digunakan sebagai acuan dalam melakukan pengerjaan tugas akhir karena memiliki metode yang sama yaitu <i>Simulated Annealing</i> dan juga beberapa variabel yang serupa karena mengambil contoh kasus yang sama yaitu <i>nurse rostering problem</i> di sebuah rumah sakit.</p>
--------------------------------------	--

## 2.2. Dasar Teori

### 2.2.1. RSIA Kendangsari

RSIA Kendangsari merupakan rumah sakit yang dikhususkan untuk melayani kebutuhan Ibu dan Anak. Pada awalnya rumah sakit ini dibangun atas kesamaan visi dan misi oleh beberapa dokter spesialis kebidanan dan kandungan dalam upaya meningkatkan kualitas kesehatan reproduksi kaum perempuan khususnya ibu hamil. Maka dari itu dibentuklah PT. Sandra Buana Medika yang mengajukan izin untuk mendirikan Rumah Sakit Ibu dan Anak pada tanggal 2009 dengan lingkup bidang usaha jasa rumah sakit swasta. Pada bulan Januari 2011, rumah sakit ini resmi dibuka dengan mulai menerima pasien untuk Poli Obygn dan Poli Anak [1].

Rumah Sakit Ibu dan Anak Kendangsari ini sendiri memiliki berbagai macam fasilitas mulai dari UGD hingga layanan rawat inap dengan pilihan berbagai macam kelas untuk ruangnya [6]. Selain itu, rumah sakit ini juga mempunyai berbagai macam fasilitas unggulan seperti *Screening 4D*, senam hamil, rooming in, pijat bayi, inisiasi menyusu dini, dan pijat bayi [7]. Untuk tenaga medis sendiri, saat ini terdapat 4 dokter yang melayani rumah sakit ini..

### 2.2.2. Penjadwalan

Penjadwalan merupakan salah satu hal yang krusial dalam bidang usaha produksi maupun jasa suatu perusahaan.

Penjadwalan berhubungan dengan alokasi sumber daya kepada pekerjaan atau *task* dalam sebuah periode waktu dan tujuan dari penjadwalan adalah untuk mengoptimasi satu atau lebih tujuan yang diinginkan [8].

Penjadwalan di bidang jasa layanan tidak semudah penjadwalan dalam bidang produksi. Ada banyak masalah serta batasan yang harus dihadapi dan penjadwalan harus dikoordinasikan dengan sistem pendukung keputusan lain demi tercapainya suatu penjadwalan yang optimal.

### **2.2.3. Nurse Rostering Problem**

Perawat merupakan salah satu komponen penting dalam rumah sakit karena bertugas untuk merawat dan melayani pasien yang menggunakan jasa rumah sakit. Maka dari itu keberadaan perawat yang selalu berjaga selama 24 jam menjadi hal yang wajib ada dalam suatu rumah sakit. Tingkat urgensi terhadap pendekatan sistematis dalam membuat penjadwalan menjadi tinggi karena dalam lingkungan layanan kesehatan, rumah sakit harus bisa mendukung penuh kebutuhan pasien dan juga kebutuhan staff yang diperlukan. Hal inilah yang biasa disebut sebagai *nurse rostering problem* atau permasalahan penjadwalan perawat.

*Nurse rostering* kepada alokasi perawat ke dalam sejumlah periode selama beberapa waktu [3]. *Nurse rostering* biasanya mempertimbangkan hari libur dan juga *shift* kerja secara bersamaan. Tujuan dari penyelesaian *nurse rostering problem* adalah untuk memasukkan perawat ke dalam *shift* yang telah disediakan rumah sakit dalam memenuhi kebutuhan perawat yang berjaga dengan mematuhi batasan-batasan yang ada [9]. Batasan yang menjadi pertimbangan dalam *nurse rostering problem* terbagi menjadi dua bagian yaitu *hard constraint* dan *soft constraint*. *Hard constraints* merupakan semua batasan yang harus dipenuhi sedangkan *soft constraints* merupakan

batasan yang diinginkan tetapi boleh dilanggar untuk mendapatkan hasil solusi yang bisa diterapkan [3].

Beberapa contoh hal yang termasuk ke dalam *hard constraint* adalah spesifikasi *shift* dimana setiap perawat tidak boleh bekerja lebih dari 1 *shift* per hari dan semua pasien harus terlayani oleh perawat atau tidak boleh ada *shift* yang kosong [10]. Sedangkan untuk *soft constraint* termasuk di dalamnya adalah [11]:

1. Batas waktu kerja perawat (minimal/maksimal);
2. *Shift* kerja berurutan yang sama (minimal/maksimal/angka pasti);
3. *Shift* kerja yang berurutan (minimal/maksimal/angka pasti);
4. Preferensi perawat;
5. Hari libur perawat (minimal/maksimal/hari libur berurutan);
6. Waktu senggang di antara *shift* kerja (minimal);
7. Penugasan tipe *shift* (tipe *shift* maksimal, kebutuhan untuk setiap tipe *shift*);
8. Liburan dan cuti;
9. Batasan antar perawat (perawat tidak boleh bekerja atau harus bekerja dengan perawat tertentu);
10. Pola *shift*;
11. Rekam historis penjadwalan sebelumnya;
12. Kebutuhan lain dalam periode waktu yang lebih pendek atau panjang dari periode perencanaan;
13. Batasan di dalam *shift* (*shift* tidak dapat ditugaskan ke perawat yang sama);

Berikut ini adalah model matematis yang digunakan untuk menyelesaikan *nurse rostering problem* [12]:

### 2.2.3.1 Sets

$E$  = Perawat yang akan dijadwalkan,  $e \in E$

$T$  = Tipe *shift* untuk setiap perawat,  $t \in T$

$D$  = Hari dalam jangka perencanaan,  $d \in \{ 1, \dots | D \}$

$R_e$  = Regular expressions untuk perawat  $e$ ,  $r \in R_e$

$W_e$  = Batas waktu kerja untuk perawat  $e$ ,  $w \in W_e$

### 2.2.3.2. Parameters

$r_{er}^{max}$  = Angka maksimal kecocokan antara *regular expression*  $r$  di dalam jadwal kerja perawat  $e$ .

$r_{er}^{min}$  = Angka minimal kecocokan antara *regular expression*  $r$  di dalam jadwal kerja perawat  $e$ .

$a_{er}$  = Bobot yang diasosiasikan dengan *regular expression*  $r$  untuk perawat  $e$ .

$v_{ew}^{max}$  = Angka maksimal dari jam yang dialokasikan untuk perawat  $e$  selama dalam periode waktu yang didefinisikan oleh batasan waktu kerja.

$v_{ew}^{min}$  = Angka minimal dari jam yang dialokasikan untuk perawat  $e$  selama dalam periode waktu yang didefinisikan oleh Batasan waktu kerja.

$b_{ew}$  = Bobot yang diasosiasikan dengan Batasan waktu kerja  $w$  untuk perawat  $e$ .

$s_{td}^{max}$  = Angka maksimum dari tipe *shift*  $t$  yang dibutuhkan pada hari  $d$ .

$s_{td}^{min}$  = Angka minimum dari tipe *shift*  $t$  yang dibutuhkan pada hari  $d$ .

$c_{td}$  = Bobot yang diasosiasikan dengan kebutuhan minimal tipe *shift*  $t$  dalam hari  $d$ .

### 2.2.3.3. Variables

$x_{etd}$  = bernilai 1 jika perawat  $e$  diberikan tipe *shift*  $t$  pada hari  $d$ , dan bernilai 0 bila tidak ada pemberian kerja pada *shift* tersebut.

$n_{er}$  = Angka kecocokan dari *regular expression*  $r$  di dalam jadwal kerja dari perawat  $e$ .

$p_{ew}$  = Jumlah jam kerja yang dialokasikan untuk perawat  $e$  selama periode waktu yang didefinisikan dengan batas kerja  $w$ .

$q_{td}$  = Jumlah tipe *shift*  $t$  yang dialokasikan untuk hari  $d$ .

#### 2.2.3.4. Constraints

Perawat hanya bisa dialokasikan ke dalam satu *shift* per hari.

$$\sum_{t \in T} x_{etd} \leq 1, \forall e \in E, d \in D \quad (1)$$

#### 2.2.3.5. Objective function

$$\text{Min } f(s) = \sum_{e \in E} \sum_{i=1}^4 f_{e,i}(x) + \sum_{t \in T} \sum_{d \in D} \sum_{i=5}^6 f_{t,d,i}(x) \quad (2a)$$

*where*

$$f_{e,1}(x) = \sum_{e \in R_e} \max \{0, (n_{er} - r_{er}^{\max}) a_{er}\} \quad (2b)$$

$$f_{e,2}(x) = \sum_{e \in R_e} \max \{0, (r_{er}^{\min} - n_{er}) a_{er}\} \quad (2c)$$

$$f_{e,3}(x) = \sum_{e \in R_e} \max \{0, (p_{ew} - v_{ew}^{\max}) b_{ew}\} \quad (2d)$$

$$f_{e,4}(x) = \sum_{e \in R_e} \max \{0, (v_{ew}^{\min} - p_{ew}) b_{ew}\} \quad (2e)$$

$$f_{e,5}(x) = \max \{0, (s_{td}^{\min} - q_{td}) c_{td}\} \quad (2f)$$

$$f_{e,6}(x) = \max \{0, (q_{td} - s_{td}^{\max}) c_{td}\} \quad (2g)$$

#### 2.2.4. Simulated Annealing

*Simulated Annealing* adalah sebuah teknik pencarian acak yang menggunakan analogi ketika sebuah baja mengalami pendinginan menjadi struktur kristal dengan energi minimal (*annealing process*); hal ini membangun sebuah dasar teknik optimasi untuk kombinasi dari berbagai masalah [13]. Jadi teknik ini meniru perilaku proses pemanasan baja yang dipanaskan sampai suhu tertentu yang kemudian baja tersebut didinginkan perlahan. Ketika baja dipanaskan hingga suhu yang

mendidih maka atom pada baja akan bergerak bebas, namun ketika pendinginan atom tersebut menjadi tidak dapat bergerak bebas dan membentuk suatu kristal [4].

*Simulated Annealing* merupakan sebuah metode untuk permasalahan dengan *combinatorial optimization*, seperti meminimalkan fungsi dari banyak variabel yang ada. *Simulated Annealing* menggunakan analogi operasi “pendinginan” untuk masalah optimasi dengan mentransformasikan solusi yang buruk dan tidak teratur ke dalam solusi yang memiliki nilai optimal tinggi dan sesuai dengan keinginan [14].

Penyelesaian permasalahan *combinatorial optimization* dilakukan dengan mencari beberapa konfigurasi dari parameter  $\bar{X} = (X_1, X_2, \dots, X_3)$  yang meminimalkan beberapa fungsi  $f(\bar{X})$ . Fungsi ini biasanya merujuk pada fungsi *cost* atau *objective* yang merupakan sebuah ukuran konfigurasi parameter yang baik [14].

Algoritma dasar yang digunakan untuk *simulated annealing* adalah [15]:

1. Inisialisasi dengan menggunakan solusi acak atau *random*.
2. Hitung *cost* yang dihasilkan dengan fungsi *cost* yang telah didefinisikan sebelumnya.
3. Buat sebuah solusi acak untuk *neighbor*.
4. Hitung *cost* dari solusi yang baru.
5. Bandingkan antara *cost* yang baru dengan *cost* yang lama. Apabila nilai *cost* baru lebih kecil daripada *cost* lama, maka pindahkan ke solusi yang baru. Bila nilai *cost* baru lebih besar daripada *cost* maka boleh dipindahkan ke solusi yang baru ataupun tetap pada solusi yang lama.
6. Ulangi langkah 3-5 hingga sebuah solusi yang dapat diterima tercapai atau hingga meraih nilai iterasi yang maksimum.

### 2.2.5. Hyper-Heuristic

Ide mendasar dari *hyper-heuristic* ini adalah sebuah gagasan untuk mengembangkan algoritma yang lebih generic atau umum dibandingkan dengan pendekatan *meta-heuristic* standar yang umumnya memerlukan *tuning* parameter empiris berbasis intesif untuk memecahkan masalah di domain yang berbeda [16]. Perbedaan fitur yang membedakan *hyper-heuristic* dengan metode lainnya ialah bahwa metode ini berjalan dalam lingkungan pencarian heuristik (atau komponen heuristic) dibanding dengan berjalan secara langsung pada lingkungan solusi pada masalah yang ditunjukkan [17].

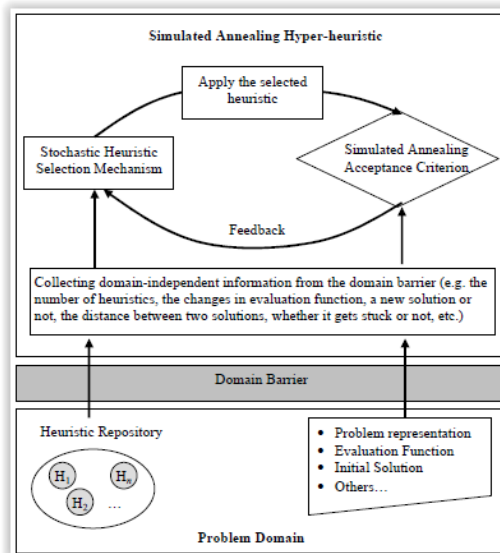
Dua kategori utama dari *hyper-heuristic* adalah *heuristic selection* dan *heuristic generation*. *Heuristic selection* merupakan metodologi untuk memilih heuristic yang sudah ada sedangkan *heuristic generation* merupakan metodologi untuk mendapatkan *heuristic* yang baru dari komponen yang sudah ada sebelumnya [17].

Dalam sumber *feedback* ketika proses pembelajaran berlangsung, *hyper-heuristic* memiliki tiga sumber yang berbeda yaitu [18]:

- *Online learning hyper-heuristic* : mempelajari sembari menyelesaikan sebuah *instance* dari masalah. Contohnya adalah penggunaan *reinforcement learning* untuk *heuristic selection*.
- *Offline learning hyper-heuristic* : mempelajari, dari sekumpulan *training instance*, sebuah metode yang dapat menggeneralisasi *instance* yang tidak terlihat. Contohnya adalah sistem klasifikasi dan pemrograman genetika.
- *No-learning hyper-heuristic* : Tidak menggunakan *feedback* dari proses pencarian.

### 2.2.6. Simulated Annealing Hyper-Heuristic

Berikut adalah *framework* yang digunakan oleh Bai Ruibin [5] dalam penggunaan metode *simulated annealing hyper-heuristic*. Sama seperti *framework hyper-heuristic* sebelumnya, terdapat *domain barrier* yang terletak di antara *hyper-heuristic* dan lingkup masalah.



**Gambar 2.1** Algoritma *Simulated Annealing Hyper-Heuristic*



Untuk memfasilitasi nilai kepuasan dari tingkat generalitas, informasi *domain-dependent* tidak diperbolehkan untuk ditransfer ke algoritma *hyper-heuristic*. Namun, informasi non-domain diperbolehkan melewati batas atau *barrier* sehingga *hyper-heuristic* dapat mengeksploitasi data ini.

Pertama, model ini mengadopsi kriteria penerimaan *simulated annealing* untuk mengurangi kelemahan dari dua kriteria penerimaan sederhana yang telah digunakan di pendekatan *hyper-heuristic* yang lainnya. Kedua, mekanisme pemilihan stokastik heuristic digunakan karena memiliki peringkat yang lebih tinggi dibanding metode lain dalam hal algoritma evolusioner untuk optimasi yang memiliki *constraint*. Mekanisme pemilihan *heuristic* akan secara dinamis mengatur prioritas yang berbeda selama pencarian. Pada awalnya tentu saja, mekanisme pemilihan ini tidak mengetahui mana *heuristic* yang memiliki performa lebih baik. Ketika melakukan pencarian, mekanisme pemilihan *heuristic* akan memulai menerapkan preferensi di antara berbagai *low-level heuristic* yang berbeda dengan mempelajari dan beradaptasi pada performa historikalnya. Untuk memilih *heuristic* yang tepat, maka *simulated annealing acceptance criterion* juga harus terpenuhi. Ketika keputusan sudah dibuat oleh mekanisme pemilihan *heuristic*, maka *heuristic* yang terpilih akan diterapkan terhadap solusi yang ada [5].

Sebagai tambahan, terdapat pula beberapa fitur yang ditambah ke dalam *simulated annealing hyper-heuristic* [5]:

1. Pemanfaatan *short-term memories*. Panjang dari *memories (learning period)* lebih pendek dibandingkan dengan keseluruhan periode pencarian. Dengan asumsi bahwa setiap *low-level heuristic* memiliki derajat performa yang berbeda di lingkungan pencarian atau periode yang berbeda. Jika sebuah *heuristic* meningkatkan solusi secara teratur pada awal pencarian, hal ini bukan berarti bahwa *heuristic* tersebut akan efektif di tengah atau bagian akhir dari proses *annealing*. Maka *memory hyper-heuristic* harus dibatasi.

2. Algoritma harus dapat membedakan situasi dimana *heuristic* gagal untuk membuat solusi yang baru dan *heuristic* yang mengembalikan solusi baru namun gagal untuk meningkatkan fungsi *objective*. *Hyper-heuristic* harus mampu untuk mendukung pemanggilan tipe *heuristic* yang lain sebari mengurangi tipe pertama, terutama ketika algoritma terhenti pada nilai *local optimum*.

### 2.2.7. Jain's Fairness Index

*Fairness* atau keadilan merupakan pertimbangan yang penting dalam studi pengujian performa. Khususnya dalam sistem distribusi dimana sebuah kumpulan sumber daya digunakan secara bersamaan oleh sekelompok pengguna maka alokasi yang adil menjadi penting. Jain menawarkan sebuah perhitungan dalam mengukur *fairness* [19].

$$f(\mathbf{x}) = \frac{(\sum_{i=1}^n x_i)^2}{\sum_{i=1}^n x_i^2} \quad \mathbf{x}_i \geq \mathbf{0} \quad (2.1)$$

Nilai  $n$  menyatakan *user* atau pengguna yang ingin dihitung dan *user* ke- $i$  menerima alokasi  $x_i$ . Indeks ini mengukur kesetaraan antara *user* dari alokasi  $x$ . Jika semua *user* mendapatkan nilai yang sama (nilai  $x_i$  sama), maka indeks *fairness* adalah 1, dan sistem berarti 100% adil. Selama perbedaan meningkat, keadilan menjadi berkurang dan memiliki indeks *fairness* mendekati 0. Nilai JFI terbatas antara 0 dan 1 dengan nilai yang semakin besar menyatakan bahwa solusi menjadi semakin adil. Nilai 1 dapat disamakan dengan keadilan yang merata [20].

### 2.2.8. Hill Climbing

Hill Climbing merupakan salah satu metode pencarian heuristik dan teknik optimasi yang merupakan sebuah algoritma iteratif yang dimulai dengan dengan sebuah inisiasi solusi untuk sebuah masalah kemudian berusaha mencari solusi yang lebih baik dengan mengubah salah satu variabel. Apabila perubahan menghasilkan hasil atau solusi yang lebih baik maka akan

ditetapkan perubahan tersebut dan diulang hingga tidak ada lagi solusi terbaik yang bisa dihasilkan.

Akan dijelaskan algoritma *hill climbing* dalam aturan yang umum. Semisalkan kita memiliki masalah optimasi kombinasi yang kita ingin cari solusinya. Untuk setiap masalah  $I$ , terdapat kumpulan *feasible solution*  $F(I)$  dimana setiap *feasible solution*  $X$  memiliki nilai cost  $c(X)$ . Solusi optimal adalah *feasible solution*  $X$  yang memiliki nilai paling minimum. Semisalkan kita telah mendefinisikan heuristic  $H$  dan kita juga memiliki beberapa metode untuk membuat inisiasi *feasible solution*  $X$ . Algoritma *hill climbing* bisa dilihat pada Kode Program 2.1 [21].

```
Generate initial feasible solution X;  
While x is not a local minimum do  
  Begin  
    Choose any  $Y \in N(X)$  such that  $c(Y) \leq c(X)$ ;  
    X:=Y  
  End;
```

**Kode Program 2.1 Pseudo-Code Hill Climbing**

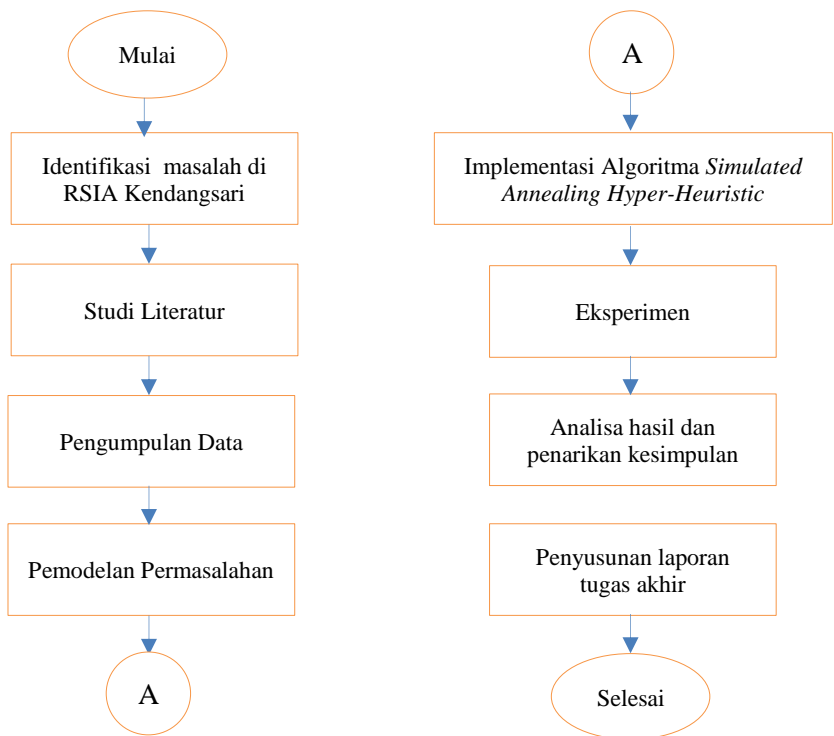
*(Halaman ini sengaja dikosongkan)*

### BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan terkait metodologi yang akan digunakan sebagai panduan untuk menyelesaikan penelitian tugas akhir ini.

#### 3.1. Metodologi Penelitian

Diagram Metodologi dari Tugas Akhir ini dapat dilihat pada gambar 3.1



Gambar 3.1 Metodologi Penelitian

### **3.2. Uraian Metodologi**

Di bawah ini adalah penjelasan dari setiap proses alur metodologi berdasarkan diagram alur metodologi berdasarkan diagram alur metodologi pada sub bab sebelumnya.

#### **3.2.1. Identifikasi Permasalahan di Rumah Sakit Ibu dan Anak Kendangsari**

Tahap ini merupakan tahap pertama yang dilakukan dalam pengerjaan tugas akhir ini. Pada tahapan ini, dilakukan identifikasi terhadap permasalahan pada objek yang akan diangkat menjadi topik pada tugas akhir ini. Identifikasi masalah dilakukan dengan mengetahui proses bisnis serta regulasi mengenai penjadwalan perawat pada RSIA Kendangsari.

#### **3.2.2. Studi Literatur**

Pada tahapan ini dilakukan pengumpulan berbagai data dan informasi serta mengkaji pustaka tentang konsep serta metode yang digunakan untuk menyelesaikan permasalahan. Selain itu dilakukan pula pengumpulan informasi terkait objek yang diambil. Sumber informasi yang bisa digunakan yaitu dengan mencari paper, jurnal, laporan penelitian ataupun tugas akhir terkait permasalahan yang ada. Dari studi literatur yang dilakukan maka didapatkan faktor-faktor yang mempengaruhi permasalahan dan juga batasan dari permasalahan. Setelah mengetahui permasalahan, faktor-faktor yang mempengaruhi permasalahan dan juga batasannya, maka langkah berikutnya yaitu menentukan metode penyelesaian permasalahan. Untuk memilih metode yang sesuai dengan permasalahan, maka diperlukan analisis terhadap metode-metode yang telah ada yang berasal dari laporan penelitian dengan permasalahan yang sama.

### 3.2.3. Pengumpulan Data

Selanjutnya dilakukan pengumpulan data yang berkaitan dengan permasalahan yang dijelaskan untuk menjawab tujuan dari tugas akhir. Adapun data yang diperlukan dalam penyelesaian permasalahan yang didapat sebagai berikut :

1. Regulasi atau peraturan penjadwalan staf di RSIA Kendangsari  
Peraturan yang dimaksud adalah aturan-aturan yang harus dipenuhi oleh perawat yang bekerja pada RSIA Kendangsari dalam hal penjadwalan. Peraturan yang dimaksud dapat berupa jumlah perawat yang harus berjaga, minimal lama jam kerja untuk setiap perawat, jumlah *shift* yang harus diambil oleh tiap perawat, dan lain-lain. Nantinya aturan-aturan tersebut akan diubah menjadi *hard constraint* pada penelitian ini.
2. Jadwal staf RSIA Kendangsari saat ini  
Jadwal perawat yang sudah ada saat ini akan dicatat dan nantinya akan dijadikan sebagai data pembanding ketika jadwal baru dengan menggunakan metode *simulated annealing hyper-heuristic* telah dibuat dalam penelitian ini.
3. Daftar staf RSIA Kendangsari  
Daftar perawat akan digunakan untuk mencari tahu jumlah dan hierarki serta jenis perawat yang ada di RSIA Kendangsari. Data ini nantinya akan digunakan sebagai bahan untuk mengetahui perbedaan antara satu jenis perawat dan jenis perawat yang lain.
4. Preferensi jadwal oleh masing-masing staf RSIA Kendangsari  
Preferensi perawat merupakan batasan yang dibuat berdasarkan keinginan perawat itu sendiri dikarenakan kondisi saat ini para perawat tidak menyukai jadwal yang sudah ada. Data tersebut merupakan nilai jaja favorit dari masing-masing perawat yang ada.

### 3.2.4. Permodelan Permasalahan

Setelah mendapatkan seluruh data yang diperlukan untuk menyelesaikan permasalahan, maka selanjutnya dilakukan pembuatan model permasalahan dari data yang sudah didapatkan. Data-data berupa peraturan perawat dan preferensi perawat nantinya akan dikonversikan menjadi model matematis. Masalah yang telah didefinisikan juga akan dibentuk dalam model otomatis. Berikut adalah beberapa model matematis yang akan dibuat :

#### 3.2.4.1. C

##### onstraint

Batasan yang digunakan sebagai acuan dalam membuat model matematisnya terbagi menjadi dua yaitu *hard constraint* dan *soft constraint*. Untuk *hard constraint* acuannya adalah regulasi yang dimiliki oleh rumah sakit. Sedangkan untuk *soft constraint* maka acuannya adalah nilai optimalitas Jain's Fairness Index yang dimiliki oleh tiap unit. Model matematis untuk membuat batasan diacu berdasarkan sub bab 2.2.3.4.

#### 3.2.4.2. V

##### ariabel Keputusa n

Variabel keputusan merupakan seperangkat variabel yang belum diketahui nilainya, yang akan dicari nilainya dalam permasalahan yang didapatkan. Model matematis dalam membuat variabel keputusan diacu berdasarkan sub bab 2.2.3.3.

#### 3.2.4.3. F

##### ungsi Tujuan

Adapun fungsi tujuan dalam melakukan tugas akhir ini adalah memaksimalkan nilai Jain's Fairness Index dalam hal hari libur tiap karyawannya dalam tiap unit. Model matematis dalam membuat fungsi tujuan diacu berdasarkan sub bab 2.2.3.5.



### **3.2.5. Implementasi Algoritma *Simulated Annealing Hyper-Heuristic***

Setelah pembuatan model matematis berdasarkan permasalahan penjadwalan perawat, akan dilakukan implementasi algoritma *simulated annealing hyper-heuristic*. Algoritma yang akan diimplementasikan mengacu pada sub bab 2.6.6. Implementasi algoritma *simulated annealing hyper-heuristic* dilakukan dengan menggunakan bahasa pemrograman JAVA.

### **3.2.6. Eksperimen**

Apabila algoritma *simulated annealing hyper-heuristic* sudah dibuat, maka langkah selanjutnya adalah melakukan eksperimen terhadap algoritma yang telah dibuat. Proses yang ada di dalam eksperimen ini adalah memastikan bahwa hasil eksperimen sudah memenuhi semua *hard constraints* dan dapat meminimalkan jumlah pelanggaran yang ada terhadap *soft constraints*. Bila sudah sesuai maka akan dilanjutkan ke proses berikutnya. Bila hasil eksperimen belum memenuhi nilai optimal, maka akan dilakukan peninjauan ulang.

### **3.2.7. Analisis Hasil dan Penarikan Kesimpulan**

Pada tahap ini dilakukan penarikan kesimpulan yang dilakukan dengan membandingkan hasil analisis dari perhitungan yang telah dilakukan. Analisis yang dilakukan yaitu dengan melakukan perbandingan antara penjadwalan yang sudah ada dengan penjadwalan optimal yang telah dihasilkan dengan metode *simulated annealing hyper-heuristic*. Dari analisa yang dilakukan akan terlihat kekurangan ataupun kelebihan dari penjadwalan yang telah dilakukan oleh perusahaan. Kemudian akan muncul mengenai saran untuk pengembangan penelitian selanjutnya agar bisa mendapatkan hasil yang lebih optimal dibandingkan dengan penelitian yang sekarang.

### 3.2.8. Penyusunan Laporan Tugas Akhir

Penyusunan laporan merupakan tahap akhir dari proses-proses yang telah dilakukan sebelumnya. Disini dilakukan dokumentasi terhadap proses-proses yang telah dilakukan dan kesimpulan dari permasalahan yang didapatkan. Seluruh pelaksanaan ataupun pengerjaan tugas akhir ini akan didokumentasikan dengan mengikuti format yang telah ditetapkan oleh laboratorium Rekayasa Data dan Intelegensia Bisnis (RDIB) serta yang berlaku di Jurusan Sistem Informasi ITS.

Di dalam laporan Tugas Akhir akan mencakup:

**a. BAB I Pendahuluan**

Pada bab ini akan dijelaskan mulai dari latar belakang, rumusan permasalahan, batasan permasalahan, tujuan dan manfaat pengerjaan tugas akhir.

**b. BAB II Tinjauan Pustaka**

Pada bab ini akan dijelaskan mengenai penelitian-penelitian sebelumnya yang telah dilakukan serta teori-teori yang menunjang permasalahan yang dibahas pada tugas akhir ini

**c. BAB III Metodologi**

Pada bab ini akan dijelaskan alur proses dari pengerjaan tugas akhir mulai dari identifikasi permasalahan sampai pembuatan laporan tugas akhir.

**d. BAB IV Perancangan**

Pada bab ini akan dijelaskan proses pengumpulan dan juga deskripsi data-data yang digunakan dalam penyelesaian tugas akhir.

**e. BAB V Implementasi**

Bab ini berisi tentang implementasi dan penjelasan setiap alur proses yang telah dijabarkan sebelumnya pada metodologi yang digunakan dalam tugas akhir.

**f. BAB VI Hasil dan Pembahasan**

Bab ini berisi analisis dan pembahasan dalam penyelesaian permasalahan yang dibahas pada pengerjaan tugas akhir.

**g. BAB VII Kesimpulan dan Saran**

Bab ini berisi kesimpulan dan saran yang ditujukan untuk kelengkapan penyempurnaan tugas akhir ini.

*(Halaman ini sengaja dikosongkan)*

## **BAB IV PERANCANGAN**

Pada bab empat ini akan membahas bagaimana rancangan dari penelitian tugas akhir yang meliputi subyek dan obyek dari Tugas Akhir, pemilihan subyek dan obyek Tugas Akhir dan bagaimana Tugas Akhir akan dilakukan.

### **4.1. Hasil Pengumpulan Data**

Pada RSIA Kendangsari, penjadwalan perawat ditentukan oleh setiap Kepala Unit dari bagian tersebut. RSIA Kendangsari memiliki total staff sebanyak 124 orang, dimana staff tersebut dibagi dalam 14 Unit. Dari total 14 Unit tersebut terdapat 6 unit yang memiliki pola unik dalam melakukan penjadwalan staf baik medis maupun non medis, yaitu:

1. Instalasi Farmasi
2. Ruang bayi dan NICU
3. SIM Rekam Medis dan Registrasi
4. Instalasi Gizi
5. Instalasi Rawat Jalan dan IGD
6. Kamar Operasi

Secara umum kepala unit dari setiap bagian RSIA Kendangsari melakukan penjadwalan sebelum tanggal 25 setiap bulannya. Nantinya pada tanggal 25, jadwal akan diberikan ke bagian Unit Sumber Daya Manusia untuk dilakukan rekap. Untuk perawat atau staff yang ingin mengajukan rikues cuti atau libur harus kepada Kepala Unit harus disampaikan sebelum tanggal 25. Kecuali apabila terdapat hal-hal yang sifatnya urgent atau mendadak seperti terdapat salah satu keluarga yang meninggal atau terkena musibah lain, maka dapat dikurangi dengan jatah cuti yang masih ada. Kemudian orang lain akan menggantikan *shift* dimana staff atau perawat tersebut seharusnya masuk.

Dalam satu bulan, perawat diberikan jatah cuti sebanyak 1 hari. Hal ini berarti untuk setiap tahunnya, perawat mendapatkan jatah cuti sebanyak 12 hari. Untuk hari liburnya sendiri, tiap

minggunya staff atau perawat mendapatkan jatah libur 1 hari. Hal ini berarti dalam 1 tahun yang terdiri dari 52 minggu, perawat mendapatkan jatah libur sebanyak 52 hari ditambah 12 jatah cuti menjadi 64 hari libur.

Beberapa masalah yang dihadapi saat ini salah satunya adalah selama ini pihak SDM merasa kesulitan untuk mengontrol jalannya penjadwalan dikarenakan banyak dari perawat yang berganti *shift* kerja tanpa adanya izin terlebih dahulu kepada pihak SDM. Masalah lain yang muncul adalah adanya jam jam dimana rumah sakit sepi atau sangat ramai. Ketika rumah sakit keadaannya sangat ramai, biasanya diselesaikan dengan mengoper perawat yang bekerja pada *shift* tersebut, dari unit yang renggang ke unit yang padat akan pasien. Namun perpindahan perawat ini tidak dimasukkan ke dalam penjadwalan karena sifatnya yang kondisional. Apabila rumah sakit keadaannya sepi, maka bagian SDM akan meliburkan perawatnya. Misalnya dokter yang bertugas di RS tersebut harus mengikuti pecan ilmiah di luar negeri. Maka SDM berinisiatif untuk meliburkan perawat yang berada pada *shift* dan unit tersebut apabila masih terdapat jatah libur yang tersisa.

#### **4.2. Aturan Penjadwalan Rumah Sakit**

Aturan penjadwalan Rumah Sakit dibuat oleh pihak rumah sakit sesuai dengan peraturan pemerintah. Berikut merupakan beberapa aturan penjadwalan Rumah Sakit yang bersifat umum :

1. *Shift* kerja pada Rumah Sakit Kendangsari secara garis besar terbagi atas 3 *shift* yaitu *shift* pagi yang dimulai pada jam 07.00-14.00 WIB, kemudian *shift* sore yang dimulai pada jam 14.00-21.00 WIB serta *shift* malam yang dimulai pada jam 21.00-07.00 WIB.
2. Staff yang memperoleh *shift* malam selama berturut-turut harus mendapatkan libur di hari berikutnya. Pola untuk *shift* malam yang sering digunakan secara berturut-turut adalah *Shift* Malam – *Shift* Malam – *Shift* Lepas Malam – Libur.

3. Setiap staf memiliki maksimal jam kerja 7 jam per hari dalam 6 hari kerja per minggunya atau setara dengan 42 jam kerja selama 1 minggu.
4. Setiap unit harus memiliki staf yang mendapatkan *shift* malam berjumlah minimal 1.
5. Untuk mengatasi jam kerja yang sibuk pada jam-jam tertentu, RSIA menerapkan *middle shift* yang berlangsung dari jam 10.00-17.00 WIB atau jam 12.00-19.00 WIB.
6. Setiap staf hanya boleh mengisi 1 shift per harinya.

#### 4.2.1. Tipe Shift dan Tipe Skill pada Masing-Masing Unit

Tipe shift pada RSIA Kendangsari terbagi menjadi 10 tipe yang terdiri dari 8 jenis tipe shift dengan jam kerja. Tipe *shift* dan jam kerja dapat dilihat pada Tabel 4.1.

Tabel 4.1 Tipe Shift RSIA Kendangsari

Tipe Shift	Notasi	Kode Shift	Jam Kerja
Pagi	P	1	07.00 – 14.00 WIB
Pagi 1	P1	2	04.30 – 11.30 WIB
Pagi 2	P2	3	05.00 – 12.00 WIB
Pagi-Sore	PS	4	06.00 – 20.00 WIB
Siang	S	5	14.00 – 21.00 WIB
Malam	M	6	21.00 – 07.00 WIB
Lepas Malam	LM	7	-
Middle Shift	MS	8	10.00 – 17.00 WIB
Middle Shift 2	MD	9	12.00 – 19.00 WIB
Libur	L	10	-

Beberapa staf di dalam RSIA Kendangsari dibagi menjadi beberapa tipe skill. Pembagian tipe skill ini didasarkan atas skill atau jabatan dari tiap staf dalam unit tersebut.

##### 1. Unit Instalasi Farmasi

Unit Farmasi memiliki 8 staff yang terdiri dari 1 Apoteker, 1 Staff Senior, 5 Anggota dan 1 Petugas Gudang Obat. Kode staf dan tipe skill lebih rinci dapat dilihat pada Tabel 4.2

**Tabel 4.2 Tipe Skill Unit Instalasi Farmasi**

No	Kode Staff	Skill	Tipe Skill
1	101	Apoteker	1
2	102	Staff Senior	2
3	103	Anggota	4
4	104	Anggota	4
5	105	Anggota	4
6	106	Anggota	4
7	107	Anggota	4
8	108	Petugas Gudang Obat	3

2. Unit Ruang Bayi dan NICU

Unit Ruang Bayi dan NICU memiliki 13 staf yang terdiri dari 1 Kepala Unit dan 12 Anggota. Kode staf dan tipe skill lebih rinci dapat dilihat pada Tabel 4.3.

**Tabel 4.3 Tipe Skill Unit Rawat Jalan dan IGD**

No	Kode Staff	Skill	Tipe Skill
1	201	Kepala Unit	1
2	202	Anggota	4
3	203	Anggota	4
4	204	Anggota	4
5	205	Anggota	4
6	206	Anggota	4
7	207	Anggota	4
8	208	Anggota	4
9	209	Anggota	4



No	Kode Staff	Skill	Tipe Skill
10	210	Anggota	4
11	211	Anggota	4
12	212	Anggota	4
13	213	Anggota	4

3. Unit SIM Rekam Medis dan Registrasi  
Unit SIM Rekam Medis dan Registrasi memiliki 6 staf yang terdiri dari 6 anggota. Kode staf dan tipe skill lebih rinci dapat dilihat pada Tabel 4.4

**Tabel 4.4 Tipe Skill Unit SIM Rekam Medis dan Registrasi**

No	Kode Staff	Skill	Tipe Skill
1	301	Anggota	4
2	302	Anggota	4
3	303	Anggota	4
4	304	Anggota	4
5	305	Anggota	4
6	306	Anggota	4

4. Unit Instalasi Gizi

Unit Instalasi Gizi memiliki 19 staf yang terdiri dari terdiri 1 ahli gizi, 4 *chef*, 4 *helper*, 6 penyaji, 2 petugas *café*, dan 2 *driver*.

**Tabel 4.5 Tipe Skill Unit Instalasi Gizi**

No	Kode Staff	Skill	Tipe Skill
1	401	Ahli Gizi	1
2	402	Chef	5
3	403	Chef	5

No	Kode Staff	Skill	Tipe Skill
4	404	Chef	5
5	405	Chef	5
6	406	Helper	6
7	407	Helper	6
8	408	Helper	6
9	409	Helper	6
10	410	Penyaji	4
11	411	Penyaji	4
12	412	Penyaji	4
13	413	Penyaji	4
14	414	Penyaji	4
15	415	Penyaji	4
16	416	Café	7
17	417	Café	7
18	418	Driver	8
19	419	Driver	8

Dari Tabel 4.5 diketahui bahwa terdapat 19 staf di dalam Unit Gizi yang terdiri 1 ahli gizi, 4 *chef*, 4 *helper*, 6 penyaji, 2 petugas *café*, dan 2 *driver*.

##### 5. Unit Rawat Jalan dan IGD

Tabel 4.6 Tipe Skill Unit Rawat Jalan dan IGD

No	Kode Staff	Skill	Tipe Skill
1	501	Anggota	4
2	502	Anggota	4

No	Kode Staff	Skill	Tipe Skill
3	503	Anggota	4
4	504	Anggota	4
5	505	Anggota	4
6	506	Anggota	4
7	507	Anggota	4
8	508	Anggota	4

Dari Tabel 4.6 diketahui bahwa terdapat 8 staf di dalam Unit Rawat Jalan dan IGD yang terdiri dari 8 anggota.

#### 6. Unit Kamar Operasi

**Tabel 4.7 Tipe Skill Unit Kamar Operasi**

No	Kode Staff	Skill	Tipe Skill
1	601	Anggota	4
2	602	Anggota	4
3	603	Anggota	4
4	604	Anggota	4
5	605	Anggota	4
6	606	Anggota	4

Dari Tabel 4.7 diketahui bahwa terdapat 6 staf di dalam Kamar Operasi yang terdiri dari 6 anggota.

### 4.3. Pembuatan Model untuk Masing-Masing Unit

Masalah penjadwalan perawat pada Rumah Sakit Ibu dan Anak Kendangsari Merr Surabaya ini adalah menjadwalkan staf dan perawat dalam kurun waktu 1 bulan, tepatnya bulan Desember 2017 yang berjumlah sebanyak 31 hari. Penjadwalan yang akan

dilakukan harus memenuhi waktu kerja serta batasan-batasan perawat lainnya. Pembuatan jadwal dilakukan untuk 6 unit yang terdiri atas unit kamar operasi, ruang bayi dan NICU, SIM Rekam Medis dan Registrasi, Instalasi Farmasi, Instalasi Rawat Jalan dan IGD dan Instalasi Gizi. Setiap unit ini memiliki batasan masing-masing tergantung kebutuhan di unitnya.

Langkah awal yang akan dilakukan adalah pembuatan jadwal Desember 2017 secara manual berdasarkan beberapa batasan-batasan yang dimiliki oleh setiap unitnya. Pembuatan jadwal Desember 2017 manual akan mempertimbangkan jadwal November 2017.

#### 4.3.1. Asumsi dan Notasi

Jadwal diasumsikan dimulai pada hari pertama bulan Desember 2017 yaitu pada hari Jumat dimana bulan Desember 2017 terdiri atas 31 hari. Berikut adalah notasi-notasi yang digunakan.

K = Jumlah hari dalam 1 bulan

N = Jumlah Total Karyawan

C = Jumlah hari Minggu dalam 1 bulan

i = Staff ke-i

z = Jumlah Tipe Shift

M = Minggu

S = Sabtu

W = Weekdays

#### 4.3.2. Variabel Keputusan

$X_{idt}$  = bernilai 1 jika staf i ditugaskan pada tipe shift t pada hari d, bernilai 0 jika tidak

Dimana

$i = 1, 2, 3, \dots, n$ ; i adalah urutan staff ke-

$d = 1, 2, 3, \dots, k$ ; d adalah urutan hari ke-

$t = 1, 2, 3, \dots, z$ ; t adalah tipe shift (lihat pada **Tabel 4.1**)

### 4.3.3. Pemodelan Unit Instalasi Farmasi

Unit Instalasi Farmasi di RSIA Kendangsari Merr Surabaya ini terdiri atas 8 orang personil yang terbagi menjadi 1 orang Apoteker, 1 orang staff senior, 5 orang anggota dan 1 orang petugas gudang obat. Dalam penjadwalan instalasi farmasi terdapat 6 jenis shift baik shift kerja maupun shift libur, yaitu shift pagi, shift sore, shift malam, *middle shift*, libur, dan lepas malam.

#### 4.3.3.1. Batasan

Penyelesaian permasalahan penjadwalan staf pada unit Instalasi Farmasi ini didasarkan atas batasan sesuai dengan aturan dan regulasi yang berlaku. Berikut merupakan batasan-batasan yang berlaku pada unit instalasi Farmasi RSIA Kendangsari.

*Hard Constraint* :

1. Jumlah staf pada shift pagi tiap harinya harus sama dengan 3
2. Jumlah staf pada shift siang tiap harinya harus sama dengan 2
3. Jumlah staf pada shift malam tiap harinya harus sama dengan 1
4. Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM) pada *shift* sebelumnya untuk anggota.

#### 4.3.5.2 Perumusan Batasan

Batasan yang dirumuskan berdasarkan batasan sebagai berikut:

##### **Hard Constraint 1**

Jumlah staf pada shift pagi tiap harinya harus sama dengan 3.

Dirumuskan untuk setiap nilai  $d$ , dan  $t=1$

$$\sum_{i=1}^n X_{id1} = 3 \quad (4.1)$$

##### **Hard Constraint 2**

Jumlah staf pada shift siang tiap harinya harus sama dengan 2. Dirumuskan untuk setiap nilai  $d$ , dan  $t=5$

$$\sum_{i=1}^n X_{id5} = 2 \quad (4.2)$$

### Hard Constraint 3

Jumlah staf pada shift malam tiap harinya harus sama dengan 1. Dirumuskan untuk setiap nilai  $d$ , dan  $t=6$

$$\sum_{i=1}^n X_{id6} = 1 \quad (4.3)$$

### Hard Constraint 4

Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM) pada *shift* sebelumnya khusus untuk anggota. Setelah 2 kali *shift* Malam maka, *shift* selanjutnya harus Lepas Malam, kemudian baru dilanjutkan dengan *shift* Libur.

#### 4.3.5.3 Langkah-langkah Penjadwalan Unit Instalasi Farmasi

Sebelum menggunakan algoritma *Simulated Annealing Hyper-Heuristic* untuk membuat optimasi penjadwalan maka perlu dilakukan pembuatan pola jadwal untuk bulan Desember 2017. Berikut ini adalah langkah-langkah dalam membuat jadwal manual yang memenuhi syarat sebagai *feasible solution* untuk unit instalasi farmasi yang nantinya akan diterapkan pada JAVA.

1. Kepala Unit dan Petugas Gudang selalu memiliki shift pagi dari hari Senin-Sabtu dan libur setiap hari Minggu.
2. Untuk staf no 2 memiliki pola Shift Siang,
3. Lihat pola M-M-LM-L yang belum lengkap pada bulan sebelumnya
4. Isikan pola M-M-LM-L melanjutkan pola bulan sebelumnya untuk Staff 3-7

5. Setiap hari harus memiliki paling tidak 1 orang shift malam sehingga jika salah satu staf sudah shift LM maka harus ada staf yang lain shift M
6. Untuk staf no 3 sampai 7 memiliki pola sehabis shift libur adalah P-P-S-S
7. Untuk staf no 3 sampai 7 sebelum mendapatkan shift malam harus mendapatkan shift siang
8. Dalam satu hari shift pagi harus terdiri dari 3, shift siang terdiri dari 2
9. Khusus hari minggu P=1 S=2 M=1 tidak ada middle shift
10. Alokasikan request Libur dan Cuti,
11. Jika jumlah Libur, Cuti, dan Lepas malam kurang dari sama dengan satu maka shift sisa merupakan shift middle.

#### **4.3.6 Pemodelan Unit Ruang Bayi dan NICU**

Unit Ruang Bayi dan NICU memiliki staff sejumlah 12 orang personil yang terdiri atas 1 Kepala Unit dan 11 anggota. Dalam penjadwalan unit Ruang Bayi dan NICU terdapat 7 jenis shift baik shift kerja maupun shift libur, yaitu shift pagi, shift sore, shift malam, *middle shift*, cuti, libur, dan lepas malam.

##### **4.3.6.1 Batasan**

Penyelesaian permasalahan penjadwalan staf pada unit Ruang Bayi dan NICU ini didasarkan atas batasan sesuai dengan aturan dan regulasi yang berlaku. Berikut merupakan batasan-batasan yang berlaku pada unit Ruang Bayi dan NICU RSIA Kendangsari.

##### *Hard Constraints*

1. Jumlah staf pada shift pagi setiap hari harus lebih dari sama dengan dua
2. Jumlah staf pada shift pagi setiap hari kurang dari sama dengan 4
3. Jumlah staf pada shift siang setiap hari harus lebih dari sama dengan satu
4. Jumlah staf pada shift siang setiap hari harus kurang dari sama dengan dua

5. Jumlah staf pada shift malam setiap hari harus lebih dari sama dengan dua dan kurang dari sama dengan tiga
6. Jumlah staf pada shift malam setiap hari harus kurang dari sama dengan tiga
7. Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM) pada *shift* sebelumnya khusus untuk anggota.

#### 4.3.6.2 Perumusan Batasan

##### Hard Constraint 1

Jumlah staf pada shift pagi setiap hari harus lebih dari sama dengan dua. Dirumuskan untuk setiap nilai  $d$ , dan  $t=1$

$$\sum_{i=1}^n X_{id1} \geq 2 \quad (4.4)$$

##### Hard Constraint 2

Jumlah staf pada shift pagi setiap hari kurang dari sama dengan empat. Dirumuskan untuk setiap nilai  $d$ , dan  $t=1$

$$\sum_{i=1}^n X_{id1} \leq 4 \quad (4.5)$$

##### Hard Constraint 3

Jumlah staf pada shift siang setiap hari harus lebih dari sama dengan satu. Dirumuskan untuk setiap nilai  $d$ , dan  $t=5$

$$\sum_{i=1}^n X_{id5} \geq 1 \quad (4.6)$$

##### Hard Constraint 4

Jumlah staf pada shift siang setiap hari harus kurang dari sama dengan dua. Dirumuskan untuk setiap nilai  $d$ , dan  $t=5$

$$\sum_{i=1}^n X_{id5} \leq 2 \quad (4.7)$$



**Hard Constraint 5**

Jumlah staf pada shift malam setiap hari harus lebih dari sama dengan dua. Dirumuskan untuk setiap nilai  $d$ , dan  $t=6$

$$\sum_{i=1}^n X_{id6} \geq 2 \quad (4.8)$$

**Hard Constraint 6**

Jumlah staf pada shift malam setiap hari harus kurang dari sama dengan tiga. Dirumuskan untuk setiap nilai  $d$ , dan  $t=6$

$$\sum_{i=1}^n X_{id6} \leq 3 \quad (4.9)$$

**Hard Constraint 7**

Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM) pada *shift* sebelumnya khusus untuk anggota. Setelah 2 kali *shift* Malam maka, *shift* selanjutnya harus Lepas Malam, kemudian baru dilanjutkan dengan *shift* Libur.

### 4.3.6.3 Langkah-langkah Penjadwalan Unit Ruang Bayi dan NICU

Sebelum menggunakan algoritma *Simulated Annealing Hyper-Heuristic* untuk membuat optimasi penjadwalan maka perlu dilakukan pembuatan pola jadwal untuk bulan Desember 2017. Berikut ini adalah langkah-langkah dalam membuat jadwal manual yang memenuhi syarat sebagai *feasible solution* untuk unit ruang bayi dan NICU yang nantinya akan diterapkan pada JAVA.

1. Isikan Pola M-M-LM-L pada setiap anggotanya, kecuali pada jadwal kepala anggota, karena kepala anggota yang hanya memiliki pola P dan L dalam satu minggunya, setiap minggunya kepala anggota libur pada hari minggu dan memiliki shift pagi dari hari senin sampai Sabtu

2. Selanjutnya untuk anggota kedua sampai anggota ke tiga belas isikan pola {"P","P","S","S","Ms","Ms"} setelah mengisi pola M-M-LM-L
3. Isi juga secara menurun geser dua kotak kekanan secara berulang

### **4.3.7 Pemodelan Unit SIM Rekam Medis dan Registrasi**

Unit SIM Rekam Medis dan Registrasi memiliki staff sejumlah 6 orang personil yang semuanya merupakan staff anggota. Dalam penjadwalan unit SIM Rekam Medis dan Registrasi terdapat 7 jenis shift baik shift kerja maupun shift libur, yaitu shift pagi, shift sore, shift malam, *middle shift*, cuti, libur, dan lepas malam.

#### **4.3.7.1 Batasan**

Penyelesaian permasalahan penjadwalan staf pada unit SIM Rekam Medis dan Registrasi ini didasarkan atas batasan sesuai dengan aturan dan regulasi yang berlaku. Berikut merupakan batasan-batasan yang berlaku pada unit SIM Rekam Medis dan Registrasi RSIA Kendangsari.

#### *Hard Constraints*

1. Jumlah staf pada shift pagi tiap harinya lebih dari sama dengan 1
2. Jumlah staf pada shift pagi tiap harinya kurang dari sama dengan 2
3. Jumlah staf pada shift siang tiap harinya lebih dari sama dengan 1
4. Jumlah staf pada shift siang tiap harinya kurang dari sama dengan 2
5. Jumlah staf pada shift malam tiap harinya harus sama dengan 1
6. Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM) pada *shift* sebelumnya.

### 4.3.7.2 Perumusan Batasan

Batasan yang dirumuskan berdasarkan batasan sebagai berikut:

#### Hard Constraint 1

Jumlah staf pada shift pagi tiap harinya lebih dari sama dengan

1. Dirumuskan untuk setiap nilai  $d$ , dan  $t=1$

$$\sum_{i=1}^n X_{id1} \geq 1 \quad (4.10)$$

#### Hard Constraint 2

Jumlah staf pada shift pagi tiap harinya kurang dari sama dengan 2. Dirumuskan untuk setiap nilai  $d$ , dan  $t=1$

$$\sum_{i=1}^n X_{id1} \leq 2 \quad (4.11)$$

#### Hard Constraint 3

Jumlah staf pada shift siang tiap harinya lebih dari sama dengan

1. Dirumuskan untuk setiap nilai  $d$ , dan  $t=5$

$$\sum_{i=1}^n X_{id5} \geq 1 \quad (4.12)$$

#### Hard Constraint 4

Jumlah staf pada shift siang tiap harinya kurang dari sama dengan 2. Dirumuskan untuk setiap nilai  $d$ , dan  $t=1$

$$\sum_{i=1}^n X_{id5} \leq 2 \quad (4.13)$$

#### Hard Constraint 5

Jumlah staf pada shift malam tiap harinya harus sama dengan 1.

Dirumuskan untuk setiap nilai  $d$ , dan  $t=6$

$$\sum_{i=1}^n X_{id6} = 1 \quad (4.14)$$

### **Hard Constraint 6**

Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas-Malam (M-M-LM) pada *shift* sebelumnya. Setelah 2 kali *shift* Malam maka, *shift* selanjutnya harus Lepas Malam, kemudian baru dilanjutkan dengan *shift* Libur.

#### **4.3.7.3 Langkah-langkah Penjadwalan Unit SIM Rekam Medis dan Registrasi**

Sebelum menggunakan algoritma *Simulated Annealing Hyper-Heuristic* untuk membuat optimasi penjadwalan maka perlu dilakukan pembuatan pola jadwal untuk bulan Desember 2017. Berikut ini adalah langkah-langkah dalam membuat jadwal manual yang memenuhi syarat sebagai *feasible solution* untuk unit SIM rekam medis dan registrasi yang nantinya akan diterapkan pada JAVA.

1. Isikan pola M-M-LM-L mengikuti pola sebelumnya pada setiap anggota.
2. Pengisian pola M-M-LM-L mengikuti aturan setiap shift malam hanya berisi 1 staff. Bila hari tersebut sudah ada orang yang mendapat shift malam, maka staff lain tidak boleh mendapat shift malam.
3. Pengisian pola M-M-LM-L diikuti dengan pola P-P-P-L-S-S-S-(-1) di bagian belakangnya.
4. Mengisi pola yang kosong dengan ketentuan apabila dalam satu hari jumlah P, S, dan M-nya kurang dari sama dengan 3 maka diisi dengan "M". Apabila lebih dari 3 maka diisi P.

#### **4.3.8 Pemodelan Unit Instalasi Gizi**

Unit Instalasi Gizi di RSIA Kendangsari Merr Surabaya ini terdiri atas 19 orang personil yang terbagi menjadi 1 orang ahli gizi, 4 orang *chef*, 4 orang *helper*, 6 orang penyaji, 2 orang petugas *café* dan 2 orang *driver*. Dalam penjadwalan instalasi gizi terdapat pembagian jenis shift yang agak berbeda dengan

unit lainnya. Terdapat shift P1 untuk jam kerja 04.00-11.30 WIB, shift P2 untuk jam kerja 05.00-12.00, shift pagi (06.00-13.00), shift PS (06.00-20.00), shift Malam (21.00-07.00), shift Siang (13.00-20.00), shift Lepas Malam dan Libur. Shift P1 dan P2 ini berlaku untuk *chef* dan *helper*. Shift PS berlaku untuk bagian *café* dan *driver*.

#### 4.3.8.1 Batasan

Penyelesaian permasalahan penjadwalan staf pada unit Instalasi Gizi ini didasarkan atas batasan sesuai dengan aturan dan regulasi yang berlaku Berikut merupakan batasan-batasan yang berlaku pada unit Instalasi Gizi RSIA Kendangsari.

##### *Hard Constraints*

1. Jumlah staf pada shift Pagi-Sore berjumlah 2 pada hari Minggu dan Senin.
2. Jumlah staf pada shift malam setiap harinya selalu sama dengan 1.
3. Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas-Malam (M-M-LM) pada *shift* sebelumnya khusus untuk penyaji.

#### 4.3.8.2 Perumusan Batasan

##### **Hard Constraint 1**

Jumlah staf pada shift Pagi-Sore berjumlah 2 pada hari Minggu dan Senin. Dirumuskan untuk nilai  $d =$  hari Minggu dan hari Senin

$$\sum_{i=1}^n X_{id4} = 2 \quad (4.15)$$

##### **Hard Constraint 2**

Jumlah staf pada shift malam setiap harinya selalu sama dengan

1. Dirumuskan untuk setiap nilai  $d$ , dan  $t=6$

$$\sum_{i=1}^n X_{id6} = 1 \quad (4.16)$$

### Hard Constraint 3

Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM) pada *shift* sebelumnya khusus untuk penyaji. Setelah 2 kali *shift* Malam maka, *shift* selanjutnya harus Lepas Malam, kemudian baru dilanjutkan dengan *shift* Libur.

#### 4.3.8.3 Langkah-langkah Penjadwalan Unit Instalasi Gizi

Sebelum menggunakan algoritma *Simulated Annealing Hyper-Heuristic* untuk membuat optimasi penjadwalan maka perlu dilakukan pembuatan pola jadwal untuk bulan Desember 2017. Berikut ini adalah langkah-langkah dalam membuat jadwal manual yang memenuhi syarat sebagai *feasible solution* untuk unit instalasi gizi yang nantinya akan diterapkan pada JAVA.

1. Ahli Gizi selalu memiliki shift pagi dari hari Senin-Sabtu dan libur setiap hari Minggu.
2. Untuk Staf 2-5 (Chef), isikan pola P1,P,S,P1,P,L,S,S,MS,MS. Cara mengisi jarak 10 kotak dengan startingpoint +2
3. Untuk Staf 6,8,9 (Helper), isikan pola P2,P,S,S,L,P,S,P2,P,MS. Cara mengisi jarak 10 kotak dengan startingpoint +2
4. Untuk staf nomer 7 isikan pola, S,S,S,P1,P1,P1,L dengan jarak antar pola 7
5. Staff 10 nomor 5 S,S,P,P,MD,MD,L dengan jarak 7
6. Lihat pola M-M-LM-L yang belum lengkap pada bulan sebelumnya untuk Staff 11-15
7. Isikan pola M-M-LM-L melanjutkan pola bulan sebelumnya untuk Staff 11-15

8. Setiap hari harus memiliki paling tidak 1 orang shift malam sehingga jika salah satu staf sudah shift LM maka harus ada staf yang lain shift M
9. Untuk staf no 11-15 dengan tipe skill Penyaji memiliki pola sehabis shift libur adalah P-P-S-S
10. Untuk staf 10.-15 jika array= null, jika kolom terdapat MD maka isi P, dan jika tidak ada MD maka isi MD
11. Alokasikan request Libur dan Cuti,
12. Jika jumlah Libur, Cuti, dan Lepas malam kurang dari sama dengan satu maka shift sisa merupakan shift middle.
13. Untuk staff 16 dengan tipe skill *Café* dan 19 dengan tipe skill *driver* memiliki pola "PS", "S", "S", "P", "P", "P", "PS", "L", "P", "P", "S", "S", "S", "L"
14. Untuk staff 17 dengan tipe skill *Cafe* dan 18 dengan tipe skill *driver* memiliki pola "L", "P", "P", "S", "S", "S", "L", "PS", "S", "S", "P", "P", "P", "PS"

### 4.3.9 Pemodelan Unit Instalasi Rawat Jalan dan IGD

Unit Instalasi Rawat Jalan dan IGD memiliki staff sejumlah 8 orang personil yang semuanya merupakan staff anggota. Dalam penjadwalan unit Instalasi Rawat Jalan dan IGD terdapat 7 jenis shift baik shift kerja maupun shift libur, yaitu shift pagi, shift sore, shift malam, *middle shift*, cuti, libur, dan lepas malam.

#### 4.3.9.1 Batasan

Penyelesaian permasalahan penjadwalan staf pada unit Instalasi Rawat Jalan dan IGD ini didasarkan atas batasan sesuai dengan aturan dan regulasi yang berlaku. Berikut merupakan batasan-batasan yang berlaku pada unit Instalasi Rawat Jalan dan IGD RSIA Kendangsari.

##### *Hard Constraints*

1. Jumlah staf pada shift pagi setiap harinya lebih dari sama dengan 1
2. Jumlah staf pada shift pagi setiap harinya kurang dari sama dengan 2

3. Jumlah staf pada shift siang setiap harinya lebih besar sama dengan 2
4. Jumlah staf pada shift siang setiap harinya kurang dari sama dengan 4
5. Jumlah staf pada shift malam setiap harinya harus berjumlah lebih dari sama dengan 1
6. Jumlah staf pada shift malam setiap harinya harus berjumlah kurang dari sama dengan 2
7. Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-LepasMalam (M-M-LM) pada *shift* sebelumnya.

#### 4.3.9.2 Perumusan Batasan

##### Hard Constraint 1

Jumlah staf pada shift pagi setiap harinya lebih dari sama dengan 1. Dirumuskan untuk setiap nilai  $d$ , dan  $t=1$

$$\sum_{i=1}^n X_{id1} \geq 1 \quad (4.17)$$

##### Hard Constraint 2

Jumlah staf pada shift pagi setiap harinya kurang dari sama dengan 2. Dirumuskan untuk setiap nilai  $d$ , dan  $t=1$

$$\sum_{i=1}^n X_{id1} \leq 2 \quad (4.18)$$

##### Hard Constraint 3

Jumlah staf pada shift siang setiap harinya lebih besar sama dengan 2. Dirumuskan untuk setiap nilai  $d$ , dan  $t=5$

$$\sum_{i=1}^n X_{id5} \geq 2 \quad (4.19)$$

##### Hard Constraint 4

Jumlah staf pada shift siang setiap harinya kurang dari sama dengan 4. Dirumuskan untuk setiap nilai  $d$ , dan  $t=5$



$$\sum_{i=1}^n X_{id5} \leq 4 \quad (4.20)$$

### Hard Constraint 5

Jumlah staf pada shift malam setiap harinya harus berjumlah lebih dari sama dengan 1. Dirumuskan untuk setiap nilai d, dan t=6

$$\sum_{i=1}^n X_{id6} \geq 1 \quad (4.21)$$

### Hard Constraint 6

Jumlah staf pada shift malam setiap harinya harus berjumlah kurang dari sama dengan 2. Dirumuskan untuk setiap nilai d, dan t=6

$$\sum_{i=1}^n X_{id6} \leq 2 \quad (4.22)$$

### Hard Constraint 7

Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM) pada *shift* sebelumnya. Setelah 2 kali *shift* Malam maka, *shift* selanjutnya harus Lepas Malam, kemudian baru dilanjutkan dengan *shift* Libur.

## 4.3.9.3 Langkah-langkah Penjadwalan Unit Kamar

### Operasi

Sebelum menggunakan algoritma *Simulated Annealing Hyper-Heuristic* untuk membuat optimasi penjadwalan maka perlu dilakukan pembuatan pola jadwal untuk bulan Desember 2017. Berikut ini adalah langkah-langkah dalam membuat jadwal manual yang memenuhi syarat sebagai *feasible solution* untuk unit instalasi rawat jalan dan IGD yang nantinya akan diterapkan pada JAVA.

1. Masukkan pola M M LM L berdasarkan pola bulan sebelumnya

2. Setiap minggu tiap perawat dapat 1 kali shift pagi, 3 kali shift sore, 1 libur, sisanya middle shift.
3. Total setiap bulan untuk tiap shift adalah 4 kali shift pagi, 9 kali shift sore, 4 libur, malam 4 kali (dari MMLM 1) sisanya MD atau MS (acak).
4. Masukkan pola “M,M,LM,L,P,P,S,S,S,S,L,MS,MD,S,S,S”

#### **4.3.10 Pemodelan Unit Kamar Operasi**

Unit Kamar Operasi memiliki staff sejumlah 6 orang personil yang semuanya merupakan staff anggota. Dalam penjadwalan unit Kamar Operasi hanya terdapat shift pagi, siang dan libur. Tidak terdapat *shift* malam pada unit kamar operasi ini.

##### **4.3.10.1 Batasan**

Penyelesaian permasalahan penjadwalan staf pada unit Kamar Operasi ini didasarkan atas batasan sesuai dengan aturan dan regulasi yang berlaku. Berikut merupakan batasan-batasan yang berlaku pada unit Kamar Operasi RSIA Kendangsari.

##### *Hard Constraints*

1. Jumlah staf pada shift pagi setiap hari harus lebih dari sama dengan dua
2. Jumlah staf pada shift pagi setiap hari harus kurang dari sama dengan empat
3. Jumlah staf pada shift siang setiap hari harus kurang dari sama dengan dua
4. Jumlah staf pada shift siang setiap hari harus kurang dari sama dengan tiga
5. Jumlah staf pada shift pagi pada hari Minggu harus terdiri dari empat staff

##### **4.3.10.2 Perumusan Batasan**

Batasan yang dirumuskan berdasarkan batasan sebagai berikut.

##### **Hard Constraint 1**

Jumlah staf pada shift pagi setiap hari harus lebih dari sama dengan 2. Dirumuskan untuk setiap nilai  $d$ , dan  $t=1$

(4.23)

$$\sum_{i=1}^n X_{id1} \geq 2$$

### Hard Constraint 2

Jumlah staf pada shift pagi setiap hari harus kurang dari sama dengan 4. Dirumuskan untuk setiap nilai d, dan t=1

$$\sum_{i=1}^n X_{id1} \leq 4 \quad (4.24)$$

### Hard Constraint 3

Jumlah staf pada shift siang setiap hari harus lebih dari sama dengan 2. Dirumuskan untuk setiap nilai d, dan t=5

$$\sum_{i=1}^n X_{id5} \geq 2 \quad (4.25)$$

### Hard Constraint 4

Jumlah staf pada shift siang setiap hari harus kurang dari sama dengan 3. Dirumuskan untuk setiap nilai d, dan t=5

$$\sum_{i=1}^n X_{id5} \leq 3 \quad (4.26)$$

### Hard Constraint 5

Jumlah staf pada shift pagi pada hari Minggu harus terdiri dari empat staff. Dirumuskan untuk setiap nilai d jika nilai d adalah hari Minggu.

$$\sum_{i=1}^n X_{id1} = 4 \quad (4.27)$$

### 4.3.10.3 Langkah-langkah Penjadwalan Unit Kamar Operasi

Sebelum menggunakan algoritma *Simulated Annealing Hyper-Heuristic* untuk membuat optimasi penjadwalan maka perlu dilakukan pembuatan pola jadwal untuk bulan Desember 2017. Berikut ini adalah langkah-langkah dalam membuat jadwal manual yang memenuhi syarat sebagai *feasible solution* untuk unit Kamar Operasi yang nantinya akan diterapkan pada JAVA.

1. Isikan Pola {"P", "P", "S", "P", "S", "L", "P", "P", "S", "-1"} untuk setiap stafi dengan roling turun satu kotak (gunakan perulangan kolom didalam baris)
2. Mengisikan kolom kosong dengan spesifikasi  $s=2$ , dan  $p \leq 3$  dengan langkah menjumlahkan isi kolom, kemudian membuat kondisi yang sesuai.
3. Setiap hari minggu shift P harus berjumlah 4 dan L berjumlah 2 (isi kolom L-L-P-P-P-P secara berulang ke bawah turun 2 kotak)

### 4.3.11 *Soft Constraint*

Yang menjadi *soft constraint* dalam pemodelan ini adalah :

1. Nilai optimalitas dari Jain's Fairness Index dari libur yang didapatkan oleh setiap staf. Apabila hari libur yang didapatkan jatuh pada hari Minggu, maka bobot yang diberikan adalah 100. Bila jatuh pada hari Sabtu, bobot yang diberikan adalah 50 dan bila jatuh pada hari kerja maka bobot yang diberikan adalah 25.
2. Jumlah libur tiap staf dalam 1 bulan sama dengan jumlah hari minggu pada 1 bulan.

### 4.3.12 Perumusan *Soft Constraint*

*Soft Constraint* yang dirumuskan adalah sebagai berikut.

#### Soft Constraint 1

Nilai Jain's Fairness Index antar staf

$$JFI = \frac{(\sum_{i=1}^n f_i)^2}{(n * \sum(f_i)^2)} \quad (4.29)$$

$$f_i = \sum_{d=1}^k \sum_{t=1}^z |X_{idt} - 1|(100M_d + 50Sb_d + 25W_d) \quad (4.28)$$

Dimana

$$M_d \begin{cases} 1, \text{jika } d \text{ pada hari Minggu} \\ 0, \text{jika tidak} \end{cases}$$

$$Sb_d \begin{cases} 1, \text{jika } d \text{ pada hari Sabtu} \\ 0, \text{jika tidak} \end{cases}$$

$$W_d \begin{cases} 1, \text{jika } d \text{ pada hari selain Sabtu dan Minggu} \\ 0, \text{jika tidak} \end{cases}$$

### Soft Constraint 2

Jumlah libur tiap staf dalam 1 bulan sama dengan jumlah hari minggu pada 1 bulan. Dirumuskan untuk setiap nilai d, dan t=10.

$$\sum_{i=1}^n X_{id10} = C \quad (4.30)$$

$$C = \sum \text{hari Minggu}$$

Dimana

$$M_d \begin{cases} 1, \text{jika } d \text{ pada hari Minggu} \\ 0, \text{jika tidak} \end{cases}$$

$$Sb_d \begin{cases} 1, \text{jika } d \text{ pada hari Sabtu} \\ 0, \text{jika tidak} \end{cases}$$

$$W_d \begin{cases} 1, \text{jika } d \text{ pada hari selain Sabtu dan Minggu} \\ 0, \text{jika tidak} \end{cases}$$

### 4.3.13 Fungsi Tujuan

Tujuan dari pemodelan permasalahan ini adalah untuk memaksimalkan nilai *Jain's Fairness Index* antar staf dalam setiap unit.

### 4.3.14 Perumusan Fungsi Tujuan

Fungsi tujuan yang dirumuskan adalah sebagai berikut

$$\mathbf{MAX} JFI = \frac{(\sum_{i=1}^n f_i)^2}{(n * \sum (f_i)^2)} \quad (4.31)$$

dimana

$$f_i = \sum_{d=1}^k \sum_{t=1}^z |X_{idt} - 1| (4M_d + 2Sb_d + W_d) \quad (4.32)$$

Dimana

$$M_d \begin{cases} 1, \text{ jika } d \text{ pada hari Minggu} \\ 0, \text{ jika tidak} \end{cases}$$

$$Sb_d \begin{cases} 1, \text{ jika } d \text{ pada hari Sabtu} \\ 0, \text{ jika tidak} \end{cases}$$

$$W_d \begin{cases} 1, \text{ jika } d \text{ pada hari selain Sabtu dan Minggu} \\ 0, \text{ jika tidak} \end{cases}$$

## 4.4 Pemodelan Simulated Annealing Hyper-Heuristic

Pemodelan algoritma *Simulated Annealing Hyper-Heuristic* yang dilakukan mengacu pada pseudo code dari Ruibin Bai dan Graham Kendall pada penelitian "*An Investigation of*

*Automated Planograms Using A Simulated Annealing Based Hyper-Heuristic*” [22].

Define an objective function  $f$  and a set of heuristics  $H$ ;  
 Define a cooling schedule: starting temperature  $t_s > 0$ , a temperature reduction function  $\alpha$  and a number of iterations for each temperature  $nrep$  ;  
 Select an initial solution  $s_0$  ;  
**Repeat**  
     Randomly select a heuristic  $h \in H$  ;  
      $iteration\_count = 0$  ;  
     **Repeat**  
          $Iteration\_count++$ ;  
         Applying  $h$  to  $s_0$  , get a new solution  $s_1$ ;  
          $\delta = f(s_1) - f(s_0)$   
         **if** ( $\delta \geq 0$ ) then  $s_0 = s_1$ ;  
         **else**  
             Generate a random  $x$  uniformly in the range (0,1)  
             **if**  $x < \exp(\delta/t)$  then  $s_0 = s_1$  ;  
     **Until**  $iteration\_count = nrep$  ;  
     Set  $t = \alpha(t)$ ;  
**Until** the stopping criteria = *true*.

**Gambar 4.1** Algoritma *Simulated Annnaling Hyper-Heuristic* [22]

Berdasarkan Gambar 4.1, langkah-langkah awal yang dilakukan pada algoritma *Simulated Annealing Hyper-Heuristic* adalah mendefinisikan fungsi tujuan yaitu memaksimalkan nilai Jain’s Fairness Index jadwal awal yang sudah dibuat. Lalu mendefinisikan beberapa heuristik yang akan dipakai pada program. Di dalam penelitian ini akan digunakan 2 jenis *low-level* heuristic yaitu :

1. H1 : Mengganti shift “Middle Shift” dengan “Libur” pada salah satu karyawan dalam kurun waktu 1 bulan
2. H2 : Menukar shift karyawan yang memiliki shift Libur dengan shift karyawan lain dalam 1 hari

Atur juga jadwal pendinginan (*cooling schedule*) mulai dari temperatur awal, fungsi penurunan temperatur yang digunakan serta jumlah iterasi yang dilakukan untuk setiap temperatur. Dalam pencarian algoritma *Simulated Annealing* untuk mencapai nilai yang global maksimum ditentukan oleh beberapa parameter, di antaranya adalah temperatur awal, *cooling factor*, jumlah iterasi setiap temperatur, temperatur akhir dan beberapa kondisi lainnya [23]. Parameter yang menjadi hal paling signifikan adalah *cooling factor* yang menjelaskan bagaimana *Simulated Annealing* mengurangi temperature ke nilai berikutnya [24]. Hingga saat ini sudah banyak metode dan algoritma yang digunakan untuk menentukan parameter dari *cooling schedule* milik *Simulated Annealing*. Dalam penelitian ini akan digunakan *Geometric Scheduling* milik van Laarhoven dan Aarts [25] yang dinyatakan dalam rumus berikut.

$$T_{k+1} = \alpha T_k$$

$T_{k+1}$  dan  $T_k$  menyatakan nilai temperatur lama dan baru, sedangkan nilai  $\alpha$  merupakan *cooling factor* yang merupakan nilai tetap yang berada antara interval [0.8, 0.99] [24]. Selanjutnya adalah memilih solusi awal. Solusi awal yang dipakai adalah nilai JFI dari jadwal awal bulan Desember 2017 yang sudah dibuat sebelumnya.

Algoritma *Simulated Annealing Hyper-Heuristic* dimulai dengan melakukan pemilihan secara random terhadap heuristik yang sudah didefinisikan atau ditentukan sebelumnya.

Iterasi dimulai ketika heuristik awal diaplikasikan terhadap ke solusi awal untuk mendapatkan solusi baru. Setelah mendapatkan solusi baru kemudian hitung perbedaan antara solusi baru dan solusi awal. Apabila nilai solusi baru lebih kecil daripada solusi awal maka solusi baru diterima dan menggantikan solusi lama. Namun bila solusi baru nilainya lebih besar, solusi tersebut masih dapat diterima dengan melakukan perhitungan  $\exp(\delta/t)$  lalu dibandingkan dengan



nilai random  $x$  antara 0 hingga 1. Apabila nilai random  $x$  lebih besar dibandingkan hasil perhitungan  $\exp(\delta/t)$  maka solusi dapat diterima. Bila sudah mendapatkan solusi baru maka tambahkan nilai iterasi.

Selanjutnya untuk menentukan nilai temperature berikutnya, dilakukan pengurangan temperature dengan fungsi  $t = \alpha(t)$ . Nilai  $\alpha$  ini bisa dijadikan parameter sehingga dapat diubah sesuai dengan keinginan. Dalam penelitian ini akan digunakan nilai *alpha* sebesar 0.98. Setelah mendapat nilai temperatur baru maka ulangi langkah-langkah dari awal pemilihan heuristik. Algoritma akan berhenti ketika sudah mencapai kriteria pemberhentian (*stopping criteria*). Kriteria pemberhentian yang dipakai dalam penelitian ini adalah apabila temperatur telah mencapai suhu tertentu yang sudah ditentukan.

*(Halaman ini sengaja dikosongkan)*

## BAB V IMPLEMENTASI

Bab ini akan membahas mengenai implementasi dari model yang telah dibuat sebelumnya. fitness.

### 5.1 Lingkungan Uji Coba

Uji coba dengan menggunakan pemodelan *Simulated Annealing Hyper-Heuristic* untuk kasus penjadwalan perawat di 6 unit pada Rumah Sakit Ibu dan Anak Kendangsari Merr Surabaya ini dilakukan dengan menggunakan bantuan NetBeans dan dengan bahasa pemrograman JAVA. Komputer yang digunakan adalah sebuah laptop Intel Core i5 dengan kapasitas RAM sebesar 4 GB dan kapasitas *hard disk* sebesar 240 GB. Perangkat lunak yang digunakan dalam pengerjaan penelitian ini adalah sistem operasi Windows 7 Home Premium dengan aplikasi komputasi NetBeans. Rincian perangkat keras dan lunak yang digunakan dapat dilihat pada Tabel 5.1.

**Tabel 5.1 Perangkat Keras dan Lunak Yang Digunakan**

<b>Perangkat Keras</b>	
Komputer	Laptop
Processor	Intel®Core™ i5-2467M CPU @ 1.60GHz
RAM	4.00 GB
<i>Hard Disk</i>	240 GB
<b>Perangkat Lunak</b>	
Microsoft Windows 7 Home Premium	Sistem Operasi
NetBeans IDE 8.0	Aplikasi untuk membuat program dengan Bahasa pemrograman JAVA
Microsoft Excel 2016	Aplikasi untuk mengolah data hasil penjadwalan

## 5.2 Membuat Otomasi Penjadwalan

Langkah awal yang dilakukan adalah mencoba membuat *feasible solution* berdasarkan batasan-batasan yang ada pada setiap unit. Pembuatan *feasible solution* ini pada awalnya dibuat secara manual seperti yang sudah dijelaskan pada bagian sub bab 4.3. Setelah *feasible solution* telah berhasil dibuat maka akan dipindahkan ke dalam program.

### 5.2.1 Memasukkan Data Set ke dalam Program JAVA

Jadwal bulan November 2017 dari masing-masing unit sebelumnya diubah ke dalam bentuk .csv terlebih dahulu. Setelah merubah file dalam bentuk .csv, kemudian csv dimasukkan ke dalam program dengan menggunakan kode program berikut. Di sini akan digunakan data yang berasal dari unit farmasi. Kode program untuk memasukkan data set dapat dilihat pada kode program 6.1.

```

1. String csvSplitBy = ",";
2. try (BufferedReader br = new BufferedReader(new
   FileReader(csvFile))) {
3.     int baris = 0;
4.     String line;
5.     while ((line = br.readLine()) != null) {
6.         String[] jadwal = line.split(csvSplitBy);
7.         roster[baris] = jadwal;
8.         baris++;
9.     }
10. }
11. catch (IOException e) {
12. }

```

#### Kode Program 5.1 Memasukkan Data Set

Sesudah jadwal lama terbaca ke dalam program, maka perlu dilakukan pemanjangan array sebanyak jumlah hari bulan Desember 2017 untuk pengisian jadwal baru.

```

1. int newSize = roster[0].length + LengthOfMonth[newMonths];
2. String temporary[][] = new String [NumberOfEmployee][newSize];
3. for (int i = 0; i < temporary.length; i++) {
4.     System.arraycopy(roster[i],0, temporary[i],0, roster[i].length);
5. }
6. roster = temporary;
7. for (String[] roster1 : roster) {
8.     for (int j = 0; j < roster[0].length; j++) {
9.         if (roster1[j] == null) {
10.            roster1[j] = "-1";
11.        }
12.    }
13. }

```

### Kode Program 5.2 Menambah Array Untuk Jadwal Baru

## 5.2.2 Memasukkan Jadwal Baru

Apabila array baru sudah dibuat maka langkah selanjutnya adalah mengisi array tersebut dengan jadwal baru berdasarkan langkah-langkah yang sudah dijelaskan sesuai dengan unit yang diinginkan sesuai pada sub bab 4.3.

### 5.2.2.1 Otomasi Penjadwalan Unit Instalasi Farmasi

Berikut adalah langkah-langkah dalam pembuatan otomasi jadwal unit Farmasi :

1. Membuat pola untuk Kepala Unit yang terdiri dari shift pagi pada hari Senin hingga Sabtu dan libur untuk hari Minggu. Hal ini berlaku juga untuk Petugas Gudang Obat yang memiliki pola jadwal yang serupa. Langkah ini dilakukan dengan membuat Kode Program 5.3.

```

1. for (int i = NumberOfDays; i < NumberOfDays+LengthOfMonth[newMonths] ;
    i++) {
2. roster[0][i] = "P";
3. roster[NumberOfEmployee-1][i] = "P";
4. if((i+FirstDay)%7 == 0){
5.     roster[0][i]= "L";
6.     roster[NumberOfEmployee-1][i] = "L";
7. }

```

### Kode Program 5.3 Pola Jadwal Kepala Unit Farmasi dan Petugas Gudang Obat

2. Membuat pola untuk staff senior yang terdiri dari shift siang untuk Senin-Sabtu dan libur untuk hari Minggu. Langkah ini dilakukan dengan membuat Kode Program 5.4.

```

1. for (int i = NumberOfDays; i <NumberOfDays+LengthOfMonth[newMonths] ;
      i++) {
2.   roster[1][i] = "S";
3.   if((i+FirstDay)%7 == 0){
4.     roster[1][i]= "L";
5.   }

```

#### **Kode Program 5.4 Pola Jadwal Staff Senior Farmasi**

3. Membuat pola M M LM M P P S S (-1) (-1) untuk staff no.3 hingga 7. Serta melakukan pengisian untuk staf Middle Shift. Kriteria pengisian *shift* untuk *Middle Shift* adalah apabila jumlah *shift* Lepas Malam, Libur dan Cuti kurang dari sama dengan 1 maka diisi dengan *Middle Shift*. Langkah ini dilakukan dengan membuat Kode Program 5.5.

```

1. int StartingPoint=0;
2. int NumbE= 5;
3. String Pattern []={"M","M", "LM","L","P","P","S","S","-1", "-1"};
4. for (int i = 0; i < NumbE; i++) {
5.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
           j++) {
6.         roster[i+2][j]= Pattern[(j-StartingPoint)%10];
7.     }
8.     StartingPoint +=2;
9. }
10. String NamaShift[]={ "P", "S", "M", "LM", "L", "CT", "MS"};
11.
12. for (int j = NumberOfDays; j <NumberOfDays+LengthOfMonth[newMonths];
       j++) {
13.     int JmlShift [] ={0,0,0,0,0,0};
14.     for (int i = 0 ; i <NumberOfEmployee; i++) {
15.         for (int k = 0; k < JmlShift.length; k++) {
16.             if(roster[i][j].equalsIgnoreCase(NamaShift[k])) {
17.                 JmlShift[k] ++;
18.             }
19.         }
20.         if(roster[i][j]== "-1"){
21.             if((JmlShift[3]+JmlShift[4]+JmlShift[5])<= 1){
22.                 roster[i][j]="MS";
23.             }else{
24.                 roster[i][j]="S";
25.             }
26.         }
27.     }

```

**Kode Program 5.5 Pola Jadwal Staff Unit Farmasi**

### 5.2.2.2 Otomasi Penjadwalan Unit Ruang Bayi dan NICU

Berikut adalah langkah-langkah dalam pembuatan otomasi jadwal unit Ruang Bayi dan NICU:

1. Membuat pola untuk Kepala Unit yang terdiri dari shift pagi pada hari Senin hingga Sabtu dan libur untuk hari Minggu. Langkah ini dilakukan dengan membuat Kode Program 5.6.

```

1. for (int i = NumberOfDays; i <NumberOfDays+LengthOfMonth[newMonths];
    i++) {
2.     roster[0][i] = "P";
3.     if((i+FirstDay)%7 == 0){
4.         roster[0][i]= "L";
5.     }
6. }

```

### Kode Program 5.6 Pola Jadwal Kepala Unit Ruang Bayi dan NICU

- Membuat pola M M LM L P P S S MS MS untuk staff no.2 hingga berikutnya. Langkah ini dilakukan dengan membuat Kode Program 5.7.

```

1. int StartingPoint=0;
2. int NumbE= 12;
3. String Pattern []={"M", "M", "LM", "L", "P", "P", "S", "S", "MS", "MS"};
4. for (int i = 0; i < NumbE; i++) {
5.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
        j++) {
6.         roster[i+1][j]= Pattern[(j-StartingPoint)%10];
7.     }
8.     StartingPoint +=2;
9. }

```

### Kode Program 5.7 Pola Jadwal Staf Unit Ruang Bayi dan NICU

#### 5.2.2.3 Otomasi Penjadwalan Unit SIM Rekam Medis dan Registrasi

Berikut adalah langkah-langkah dalam pembuatan otomasi jadwal unit SIM Rekam Medis dan Registrasi:

- Memasukkan pola M M LM L P P P S S S (-1) (-1) untuk seluruh staf yang ada. Langkah ini dilakukan dengan membuat Kode Program 5.8.



```

1. int StartingPoint=0;
2. String Pattern []={"M","M","LM","L","P","P","S","S","S","-1","-1"};
3. for (int i = 0; i < NumberOfEmployee; i++) {
4.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
           j++) {
5.         roster[i][j]= Pattern[(j-StartingPoint)%12];
6.     }
7.     StartingPoint +=2;
8. }

```

#### Kode Program 5.8 Pola Jadwal Staf Rekam Medis dan Registrasi (1)

- Menentukan pola *Middle Shift* berdasarkan aturan apabila jumlah shift pagi, siang dan malam kurang dari sama dengan 3, maka diisi dengan *Middle Shift*. Langkah ini dilakukan dengan membuat Kode Program 5.9.

```

1. String NamaShift[]={ "P", "S", "M", "LM", "L", "CT", "MS" };
2. for (int j = NumberOfDays; j <NumberOfDays+LengthOfMonth[newMonths];
       j++) {
3.     int JmlShift [] = {0,0,0,0,0,0,0};
4.     for (int i = 0 ; i < NumberOfEmployee; i++) {
5.         for (int k = 0; k < JmlShift.length; k++) {
6.             if(roster[i][j].equalsIgnoreCase(NamaShift[k])) {
7.                 JmlShift[k] ++;
8.             }
9.         }
10.        if(roster[i][j]== "-1"){
11.            if((JmlShift[0]+JmlShift[1]+JmlShift[2]) <= 3){
12.                roster[i][j]="MS";
13.            }else {
14.                roster[i][j]="P";
15.            }
16.        }
17.    }
18. }

```

#### Kode Program 5.9 Pola Jadwal Staf Rekam Medis dan Registrasi (2)

### 5.2.2.4 Otomasi Penjadwalan Unit Instalasi Gizi

Berikut adalah langkah-langkah dalam pembuatan otomasi jadwal unit Instalasi Gizi:

- Memasukkan pola untuk Ahli Gizi yang terdiri dari shift pagi pada hari Senin hingga Sabtu dan libur untuk hari

Minggu. Langkah ini dilakukan dengan membuat Kode Program 5.10.

```

1. for (int i = NumberOfDays; i < NumberOfDays+LengthOfMonth[newMonths];
      i++) {
2.     roster[0][i] = "P";
3.     roster[NumberOfEmployee-1][i] = "P";
4.     if((i+FirstDay)%7 == 0){
5.         roster[0][i]= "L";
6.         roster[NumberOfEmployee-1][i] = "L";
7.     }

```

#### Kode Program 5.10 Pola Jadwal Ahli Gizi

- Memasukkan pola P1, P, S, P1, P, L, S, S, MD, MD untuk jenis staf *Chef*. Langkah ini dilakukan dengan membuat Kode Program 5.11.

```

1. int StartingPointChef=0;
2. int NumbChef= 4;
3. String PatternChef [] = {"P1", "P", "S", "P1", "P", "L", "S", "S", "MD", "MD"};
4. for (int i = 0; i < NumbChef; i++) {
5.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
          j++) {
6.         roster[i+1][j]= PatternChef[(j-StartingPointChef)%10];
7.     }
8.     StartingPointChef += 2;
9. }

```

#### Kode Program 5.11 Pola Jadwal Chef

- Memasukkan pola L P P S S S P P untuk staf nomor 6. Langkah ini dilakukan dengan membuat Kode Program 5.12.

```

1. String Pattern7 [] = {"L", "P1", "P1", "S", "S", "S", "P1", "P1"};
2. for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
      j++) {
3.     roster[5][j]= Pattern7[(j+5)%8];
4. }

```

#### Kode Program 5.12 Pola Jadwal Staf nomor 6

- Memasukkan pola P2, P, S, S, L, P, S, P2, P, MS untuk staf nomor 7 hingga nomor 9. Langkah ini dilakukan dengan membuat Kode Program 5.13.

```

1. int StartingPointHelper=0;
2. int NumbHelper = 3;
3. String PatternHelper [] = {"P2", "P", "S", "S", "L", "P", "S", "P2", "P", "MS"};
4. for (int i = 0; i < NumbHelper; i++) {
5.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
           j++) {
6.         roster[i+6][j]= PatternHelper[(j-StartingPointHelper)%10];
7.     }
8.     StartingPointHelper += 2;
9. }

```

### Kode Program 5.13 Pola Jadwal Helper

5. Memasukkan pola L P P S S S P P untuk staff nomor 10 sebagai pekerja di bagian Gudang. Langkah ini dilakukan dengan membuat Kode Program 5.14.

```

1. String Pattern10 [] = {"S", "S", "P", "P", "MD", "MD", "L"};
2. for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
       j++) {
3.     roster[9][j]= Pattern10[(j+5)%7];
4. }

```

### Kode Program 5.14 Pola Staf Gudang Makanan

6. Memasukkan pola M M LM L P P S S untuk staff nomor 11 hingga 15 yang bekerja sebagai Penyaji. Langkah ini dilakukan dengan membuat Kode Program 5.15.

```

1. int StartingPointPenyaji=0;
2. int NumbE= 5;
3. String Pattern [] = {"M", "M", "LM", "L", "P", "P", "S", "S", "-1", "-1"};
4. for (int i = 0; i < NumbE; i++) {
5.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
           j++) {
6.         roster[i+10][j]= Pattern[(j-StartingPointPenyaji)%10];
7.     }
8.     StartingPointPenyaji += 2;
9. }

```

### Kode Program 5.15 Pola Staf Penyaji

7. Memasukkan pola PS S S P P P PS L P P S S S L untuk staf nomor 16 dan 19. Langkah ini dilakukan dengan membuat Kode Program 5.16.

```

1. for (int i = NumberOfDays; i < NumberOfDays+LengthOfMonth[newMonths] ;
    i++) {
2.     String Pattern1619 [] = {"PS", "S", "S", "P", "P", "P", "PS", "L", "P",
3.                             "P", "S", "S", "S", "L"};
4.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
        j++) {
5.         roster[15][j]= Pattern1619[(j+9)%14];
6.         roster[18][j]= Pattern1619[(j+9)%14];
7.     }
8. }

```

#### **Kode Program 5.16 Pola Staf Nomor 16 - 19**

8. Memasukkan pola L P P S S S L PS S S P P P PS untuk staf nomor 17 dan 18. Langkah ini dilakukan dengan membuat Kode Program 5.17.

```

1. for (int i = NumberOfDays; i < NumberOfDays+LengthOfMonth[newMonths] ;
    i++) {
2.     String Pattern1718 [] = {"L", "P", "P", "S", "S", "S", "L", "PS", "S",
3.                             "S", "P", "P", "P", "PS"};
4.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
        j++) {
5.         roster[16][j]= Pattern1718[(j+9)%14];
6.         roster[17][j]= Pattern1718[(j+9)%14];
7.     }
8. }

```

#### **Kode Program 5.17 Pola Staf nomor 17 dan 18**

9. Memasukkan pola Middle Shift ke dalam shift yang masih kosong. Langkah ini dilakukan dengan membuat Kode Program 5.18.

```

1. String NamaShift[]={ "P", "S", "M", "LM", "L", "CT", "MD"};
2.   for (int j = NumberOfDays; j <NumberOfDays+LengthOfMonth[newMonths];
        j++) {
3.       int JmlShift [] ={0,0,0,0,0,0,0};
4.       for (int i = 0 ; i <NumberOfEmployee; i++) {
5.           for (int k = 0; k < JmlShift.length; k++) {
6.               if(roster[i][j].equalsIgnoreCase(NamaShift[k])) {
7.                   JmlShift[k] ++;
8.               }
9.           }
10.          if(roster[i][j] == "-1"){
11.              if(JmlShift[6] == 0){
12.                  roster[i][j]="MD";
13.              }else{
14.                  roster[i][j]="P";
15.              }
16.          }
17.      }
18.  }

```

**Kode Program 5.18 Pola Middle Shift**

### 5.2.2.5 Otomasi Penjadwalan Unit Instalasi Rawat Jalan dan IGD

Berikut adalah langkah-langkah dalam pembuatan otomasi jadwal unit Rawat Jalan dan IGD :

1. Memasukkan pola M M LM M P P S S S S MD MD S S S untuk keseluruhan staf. Langkah ini dilakukan dengan membuat Kode Program 5.19.

```

1. int StartingPoint=0;
2. int NumbE= 8;
3. String Pattern []={"M", "M", "LM", "L", "P", "P", "S", "S", "S", "S", "S", "L",
                    "MD", "MS", "S", "S", "S"};
4. for (int i = 0; i < NumbE; i++) {
5.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
          j++) {
6.         roster[i][j]= Pattern[(j-StartingPoint)%15];
7.     }
8.     StartingPoint +=2;
9. }

```

**Kode Program 5.19 Pola Staf Unit Rawat Jalan dan IGD**

### 5.2.2.6 Otomasi Penjadwalan Kamar Operasi

Berikut adalah langkah-langkah dalam pembuatan otomasi jadwal unit Kamar Operasi :

1. Memasukkan pola P P S P S L P P S (-1) ke dalam pola untuk semua staf. Langkah ini dilakukan dengan membuat Kode Program 5.20.

```

1. int StartingPoint=0;
2. String Pattern []={"P", "P", "S", "P", "S", "L", "P", "P", "S", "-1"};
3. for (int i = 0; i < NumberOfEmployee; i++) {
4.     for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
           j++) {
5.         roster[i][j]= Pattern[(j-StartingPoint)%10];
6.     }
7.     StartingPoint ++;
8. }

```

#### Kode Program 5.20 Pola Staf Kamar Operasi

2. Memasukkan kolom yang masih kosong dengan shift pagi atau siang. Langkah ini dilakukan dengan membuat Kode Program 5.21.

```

1. String NamaShift[]={ "S", "P", "L", "CT" };
2. for (int j = NumberOfDays; j <NumberOfDays+LengthOfMonth[newMonths];
           j++) {
3.     int JmlShift [] ={0,0,0,0};
4.     for (int i = 0 ; i <NumberOfEmployee; i++) {
5.         for (int k = 0; k < JmlShift.length; k++) {
6.             if(roster[i][j].equalsIgnoreCase(NamaShift[k])) {
7.                 JmlShift[k] ++;
8.             }
9.         }
10.        if(roster[i][j]== "-1"){
11.            if(JmlShift[0]<1){
12.                roster[i][j]= "S";
13.            }else{
14.                roster[i][j]="P";
15.            }
16.        }
17.    }
18. }

```

#### Kode Program 5.21 Pengisian Middle Shift

3. Mengisi pola hari Minggu dengan 4 shift pagi dan 2 shift libur. Langkah ini dilakukan dengan membuat Kode Program 5.22.

```

1. int StartingPointLibur=0;
2. String pola2 []={"L","L","P","P","P","P"};
3. for (int j = NumberOfDays; j < NumberOfDays+LengthOfMonth[newMonths];
      j++) {
4.     for (int k =0; k < pola2.length; k++) {
5.         if((j+FirstDay)%7 == 0 ){
6.             roster[k][j]=pola2[(k+StartingPointLibur+2)%6];
7.         }
8.     }StartingPointLibur +=2;
9. }

```

**Kode Program 5.22 Pengisian Shift Hari Minggu Unit Kamar Operasi**

### 5.2.3 Melakukan Perhitungan Jain's Fairness Index

Setelah dilakukan otomasi dari penjadwalan di tiap unit, maka langkah selanjutnya adalah menghitung *Jain's Fairness Index* antar staf di tiap unit. Hal ini dilakukan untuk mengetahui tingkat keadilan dari dari tiap karyawan dalam hal mendapatkan hari libur. Langkah perhitungan *Jain's Fairness Index* dilakukan dengan membuat Kode Program 5.23.

```

1. public double JFi(String[][] roster){
2.     String libur[]={ "Hari Minggu", "Hari Sabtu", "Hari Kerja"};
3.     double jfi= 0;
4.     double sum=0;
5.     double JmlhJfiStafkuadrat=0;
6.     double JmlSelStaf=0;
7.     double JmlSelStaf_kuadrat=0;
8.     for (int i = 0 ; i <NumberOfEmployee; i++) {
9.         int a=0;
10.        int JmlLibur[]= {0,0,0};
11.        int Bobot[]={4,2,1};
12.        int liburXbobot= 0;
13.        int Jml_liburXbobot=0;
14.        double JFIperStaf_Kuadrat= 0;
15.        for (int j = NumberOfDays; j <NumberOfDays+LengthOfMonth
[newMonths]; j++) {
16.            if(roster[i][j].equalsIgnoreCase("L")){
17.                if((j+FirstDay)%7 == 0){
18.                    JmlLibur[0]++;
19.                }else if ((j+FirstDay)%7 == 6){
20.                    JmlLibur[1]++;
21.                }else{
22.                    JmlLibur[2]++;
23.                }
24.            }
25.        }
26.        for (int k = 0; k < JmlLibur.length; k++) {
27.            int bb= JmlLibur[k];
28.            liburXbobot =JmlLibur[k]*Bobot[k];
29.            Jml_liburXbobot +=liburXbobot;
30.        }
31.        JFIperStaf_Kuadrat = Math.pow(Jml_liburXbobot, 2);
32.        JmlhJfiStafkuadrat +=JFIperStaf_Kuadrat;
33.        JmlSelStaf +=Jml_liburXbobot;
34.    }
35.    JmlSelStaf_kuadrat = Math.pow(JmlSelStaf, 2);
36.    jfi=(JmlSelStaf_kuadrat /(NumberOfEmployee*JmlhJfiStafkuadrat));
37.    return jfi;
38. }

```

### Kode Program 5.23 Menghitung Nilai Jain's Fairness Index

Pertama-tama diinisiasikan terlebih dahulu array serta variabel yang nantinya akan digunakan ke dalam perhitungan. Selanjutnya lakukan iterasi untuk keseluruhan jadwal untuk melakukan pencarian indeks yang memiliki shift libur. Shift libur yang ditemukan akan dimasukkan ke dalam array yang berbeda-beda. Array pertama untuk libur yang jatuh pada hari Minggu. Array kedua untuk libur yang jatuh pada hari Sabtu



dan Array terakhir untuk libur yang jatuh pada hari kerja (Senin-Jum'at).

Selanjutnya adalah pemberian bobot untuk masing-masing array. Apabila shift libur jatuh pada hari Minggu, maka akan diberi bobot sebesar 100. Apabila jatuh pada hari Sabtu maka akan diberi bobot sebesar 50 dan apabila jatuh pada hari kerja maka akan diberi bobot sebesar 25.

Setelah dikalikan dengan bobot, selanjutnya dijumlahkan untuk nilai yang dimiliki dari masing-masing karyawan. Nilai yang dimiliki karyawan nantinya akan dijumlahkan keseluruhannya. Selain itu nilai hasil penjumlahan masing-masing karyawan nantinya akan dipangkatkan juga serta ditotalkan nilainya dari seluruh karyawan. Terakhir masukkan rumus JFInya.

#### **5.2.4 Melakukan Pengecekan Constraint**

Sebelum jadwal diterapkan algoritma, dilakukan pengecekan constraint pada jadwal yang sudah dibuat sesuai dengan batasan yang dimiliki masing-masing unit. Berikut adalah contoh kode program untuk melakukan pengecekan constraint pada Unit Farmasi. Pada Unit Farmasi, jumlah pekerja pada hari kerja harus terdiri dari 3 orang pada *shift* Pagi, 2 orang pada *shift* Siang, dan 1 orang pada *shift* Malam. Namun pada hari Minggu, hanya ada 1 orang yang bekerja pada *shift* Pagi. Langkah untuk membuat pengecekan *constraint* dilakukan dengan membuat Kode Program 5.24.

```

1. public boolean cekConstraint(String [][] roster){
2.     boolean value1= false;
3.     boolean value2= false;
4.     for (int j = NumberOfDays; j <NumberOfDays+LengthOfMonth[newMonths]; j++) {
5.         String [] Shift ={"P","S","M"};
6.         int [] Jml_Shift = {0,0,0};
7.         for (int i = 0; i < NumberOfEmployee; i++) {
8.             if((j+FirstDay)%7 == 0){
9.                 if(roster[i][j].equalsIgnoreCase("P")){
10.                    Jml_Shift[0]++;
11.                } else if(roster[i][j].equalsIgnoreCase("S")){
12.                    Jml_Shift[1]++;
13.                } else if(roster[i][j].equalsIgnoreCase("M")){
14.                    Jml_Shift[2]++;
15.                }
16.            }
17.            if ((Jml_Shift[0] == 1 && Jml_Shift[1]== 2 )&& Jml_Shift[2]== 1){
18.                value1 = true;
19.            }else{
20.                value1 = false;
21.            }
22.        } else{
23.            if(roster[i][j].equalsIgnoreCase("P")){
24.                Jml_Shift[0]++;
25.            } else if(roster[i][j].equalsIgnoreCase("S")){
26.                Jml_Shift[1]++;
27.            } else if(roster[i][j].equalsIgnoreCase("M")){
28.                Jml_Shift[2]++;
29.            }
30.            if (Jml_Shift[0]== 3 && Jml_Shift[1]== 2 && Jml_Shift[2]== 1){
31.                value2 = true;
32.            } else{
33.                value2 = false;
34.            }
35.        }
36.    }
37. }
38. System.out.println("Cek Constraint");
39. System.out.println("Nilai value1 : " + value1);
40. System.out.println("Nilai value2 : " + value2);
41. if (value1=value2) {
42.     value1=true;
43. } else {
44.     value1=false;
45. }
46. System.out.println(value1);
47. return value1;
48. }

```

### Kode Program 5.24 Mengecek Constraint Unit Farmasi

### 5.3. Membuat Optimasi Penjadwalan

Pada tahap ini akan didefinisikan *low-level heuristic* yang digunakan, algoritma serta parameter pada *Simulated Annealing Hyper-Heuristic* yang dipakai, serta algoritma *Hill Climbing* sebagai bentuk perbandingan.

#### 5.3.1 Mendefinisikan Low Level Heuristic

Dalam penggunaan algoritma *hyper-heuristic* ini akan digunakan 2 jenis *low-level heuristic* untuk menjalankannya. Berikut adalah kode program yang akan digunakan dalam *low-level heuristic*.

1. H1 (*Move*): Mengganti shift “Middle Shift” dengan “Libur” pada salah satu karyawan dalam kurun waktu 1 bulan. Pembuatan *low-level heuristic (Move)* ini dibuat sesuai dengan Kode Program 5.25.

```

1. public void move(){
2.     int JmlLibur;
3.     int selisih;
4.     int JumlahLibur_seharusnya = (LengthOfMonth[newMonths]/7);
5.     if(((NumberOfDays + (FirstDay - 1)) % 7) >=4){
6.         JumlahLibur_seharusnya ++;
7.     }
8.     for (int i = 0 ; i <NumberOfEmployee; i++) {
9.         JmlLibur=0;
10.        selisih=0;
11.        List<Integer> posisiMS = new ArrayList<>();
12.        for (int j = NumberOfDays; j <NumberOfDays+LengthOfMonth[newMonths];
13.            j++) {
14.            if(roster[i][j].equalsIgnoreCase("L")){
15.                JmlLibur++;
16.            }
17.            if (roster [i][j].equalsIgnoreCase("MS")){
18.                posisiMS.add(new Integer(j));
19.            }
20.            selisih = JumlahLibur_seharusnya-JmlLibur;
21.            if (selisih >0 && posisiMS.size()>0){
22.                for (int k = 0; k < selisih; k++) {
23.                    int IndexMSygDiganti = rand.nextInt(posisiMS.size());
24.                    roster[i][posisiMS.get(IndexMSygDiganti)]= "L";
25.                }
26.            }
27.        }
28.    }

```

#### Kode Program 5.25 Low-Level Heuristic Move

Dilakukan perhitungan jumlah libur seharusnya pada bulan Desember. Didapatkan bahwa jumlah hari libur seharusnya pada Desember 2017 adalah 5 hari. Kemudian akan dicari karyawan yang memiliki *Middle Shift* dan Libur kemudian disimpan pada array masing-masing. Kemudian dilakukan perhitungan selisih antara jumlah libur yang sudah ada dengan jumlah libur seharusnya. Apabila ditemukan adanya perbedaan maka akan dilakukan penggantian *Middle Shift* secara random menjadi Libur.

- H2 (*Swap*): Menukar shift karyawan yang memiliki shift Libur dengan shift karyawan lain dalam 1 hari yang sama. Pembuatan *low-level heuristic (Swap)* ini dibuat sesuai dengan Kode Program 5.26.

```

1. public void swap(){
2.     List<Integer> posisiLHari = new ArrayList<>();
3.     List<Integer> posisiLEmployee = new ArrayList<>();
4.     for (int j = NumberOfDays; j <NumberOfDays+LengthOfMonth[newMonths];
5.         j++) {
6.         for (int i = 0 ; i <NumberOfEmployee; i++) {
7.             if(roster[i][j].equalsIgnoreCase("L")){
8.                 posisiLHari.add(new Integer(j));
9.                 posisiLEmployee.add(new Integer(i));
10.            }
11.        }
12.        int memilihHari = rand.nextInt(posisiLHari.size());
13.        String temp = roster[posisiLEmployee.get(memilihHari)]
14.            [posisiLHari.get(memilihHari)];
15.        int memilihEmployee;
16.        do {
17.            memilihEmployee = rand.nextInt(NumberOfEmployee);
18.        }while(memilihEmployee == posisiLEmployee.get(memilihHari));
19.        roster[posisiLEmployee.get(memilihHari)][posisiLHari.get(memilihHari)]
20.            = roster[posisiLEmployee.get(memilihEmployee)]
21.            [posisiLHari.get(memilihHari)]= temp;

```

#### Kode Program 5.26 Low Level Heuristic Swap

Dibuat 2 jenis Array List yang nantinya akan diisi dengan posisi *shift* Libur jatuh pada hari apa dan *shift* Libur milik karyawan siapa. Kemudian dilakukan pengisian pada list tersebut untuk menyimpan semua *shift* Libur pada bulan tersebut. Selanjutnya dilakukan pemilihan secara acak terhadap *shift* Libur yang ingin diubah. Setelah terpilih maka pada kolom atau hari yang sama dilakukan pemilihan karyawan secara acak. Selanjutnya dilakukan penukaran *shift* terhadap *shift* Libur yang telah ditentukan di awal.

### 5.3.2 Menerapkan Algoritma Simulated Annealing Hyper-Heuristic

Langkah selanjutnya adalah menuliskan kode program untuk optimasi sesuai dengan Kode Program 5.27.

```

1. public void SAHH (double cooldown, double Tawal, double Takhir){
2.     String [][] bestSolution = new String [NumberOfEmployee]
           [LengthOfMonth[newMonths]]
3.     bestJFICost = curJFICost = JFi(roster);
4.     double bestJFICostFinal = 0;
5.     double TAwal = Tawal;
6.     double TAKhir = Takhir;
7.     double alpha = cooldown;
8.     int hitungIterasi;
9.     int iterasisuhu = 0;
10.
11.    while (TAwal >= TAKhir ) {
12.        hitungIterasi = 0;
13.        String [][] Solpertama = roster.clone();
14.        for (int i = 0; i < 1; i++) {
15.            hitungIterasi++;
16.            System.out.printf("%5d, %5f, %5f, %5f \n", iterasisuhu, TAwal
                ,curJFICost, bestJFICost );
17.            if (Math.random() < 0.5) {
18.                move();
19.            } else {
20.                swap();
21.            }
22.            curJFICost = JFi(roster);
23.            double selisihJFI = curJFICost - bestJFICost;
24.            double ap = Math.exp(Math.abs(selisihJFI) / TAwal);
25.            Math.random();
26.            if (selisihJFI > 0 ) {
27.                bestJFICost = curJFICost;
28.            } else {
29.                double x = Math.random();
30.                if(x > ap){
31.                    bestJFICost = curJFICost;
32.                    System.out.println("Akse X > ap");
33.                } else{
34.                    roster = Solpertama.clone();
35.                }
36.            }
37.        }
38.        //TAwal = TAwal / (1 + (alpha*TAwal));
39.        TAwal = TAwal * alpha ;
40.        iterasisuhu++;
41.    }
42.    System.out.println("Iterasi Suhu : " + iterasisuhu);
43.    System.out.println("Best Solution Model : " + bestJFICost);
44. }

```

**Kode Program 5.27 Algoritma Simulated Annealing Hyper-Heuristic**

Pada awalnya akan diinisiasi beberapa variabel seperti temperatur awal, temperatur akhir, *cooling schedule* dan

beberapa hal lainnya. Kemudian dilakukan iterasi mulai dari temperatur awal hingga temperatur akhir. Selama iterasi berlangsung, pada awalnya akan dilakukan pemilihan heuristik secara random antara *move()* atau *swap()*. Kemudian hasil perhitungan dari JFI yang telah terpilih akan disimpan sebagai *current solution*.

Selanjutnya akan dilakukan perhitungan selisih antar *current solution* dan *best solution*. Apabila nilai *current solution* lebih baik daripada *best solution*, maka nilai *current solution* akan dijadikan *best solution*. Namun apabila nilainya lebih kecil, bisa jadi ditolak atau diterima dengan adanya probabilitas. Nilai probabilitas ini dihitung dengan nilai eksponensial dipangkatkan dengan selisih antara *best solution* dan *current solution* kemudian dibagi dengan temperatur saat iterasi. Kemudian nilai probabilitas ini dibandingkan dengan nilai random. Apabila nilai random ini hasilnya lebih besar dari probabilitas maka akan solusi akan diterima. Bila tidak, maka kembali ke solusi sebelumnya dan akan dilakukan penurunan suhu.

Untuk melakukan penentuan pada nilai *cooling schedule* dilakukan beberapa pengujian. Pengujian ini dilakukan pada unit Farmasi.

**Tabel 5.2 Pengujian Nilai *Cooling Schedule* pada Unit Farmasi**

No	Nilai <i>Cooling Schedule</i>	Jumlah Iterasi	Nilai JFI
1	0.1	4	0.8252
2	0.2	5	0.8193
3	0.3	6	0.8618
4	0.4	8	0.7898
5	0.5	10	0.8002
6	0.6	14	0.8147
7	0.7	20	0.8315
8	0.8	31	0.8836

No	Nilai <i>Cooling Schedule</i>	Jumlah Iterasi	Nilai JFI
9	0.9	66	0.8620
10	0.91	74	0.9067
11	0.92	83	0.9316
12	0.93	96	0.9166
13	0.94	112	0.8120
14	0.95	135	0.9039
15	0.96	170	0.9061
16	0.97	227	0.8563
17	0.98	342	0.9288
18	0.99	688	0.8922

Berdasarkan Tabel 5.2, dapat diketahui bahwa semakin rendah koefisien pada *cooling schedule*, maka akan semakin sedikit iterasi yang dihasilkan. Penurunan suhu ini dilakukan dengan menggunakan *Geometric Scheduling* dengan pemilihan nilai untuk *cooling schedule* adalah 0.98 yang menghasilkan sejumlah 342 iterasi. Hal ini dikarenakan apabila menggunakan nilai *cooling schedule* yang lebih kecil, maka masih ada kemungkinan algoritma mencapai *best solution*.



## BAB VI HASIL DAN PEMBAHASAN

Pada bagian ini akan dijelaskan hasil implementasi dari rancangan penerapan algoritma *Simulated Annealing Hyper-Heuristic*. Bagian ini akan menjelaskan perbandingan hasil penjadwalan serta JFI masing-masing dari jadwal manual, otomatis, maupun optimasi dengan menggunakan algoritma *Simulated Annealing Hyper-Heuristic*.

### 6.1 Hasil Perhitungan Nilai Jain's Fairness Index Jadwal Manual

Nilai *Jain's Fairness Index* dihitung untuk keenam unit pada jadwal bulan November 2017. Hasil perhitungan ini akan dibandingkan dengan hasil lainnya.

#### 6.1.1 Jadwal Manual Unit Instalasi Farmasi

Jadwal manual untuk Unit Farmasi yang sudah dimasukkan ke dalam program, akan dimunculkan seperti pada Gambar 6.1.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R
P	P	P	P	L	P	P	P	P	P	P	L	P	P	P
S	MS	S	S	MS	L	MS	MS	MS	S	S	L	S	S	S
M	M	LM	L	P	S	P	P	M	M	LM	L	CT	P	S
P	P	L	MS	M	M	LM	L	P	P	CT	S	S	S	M
LM	L	S	S	S	P	M	M	LM	L	P	MS	P	CT	P
L	S	M	M	LM	L	S	S	S	CT	S	P	M	M	LM
S	S	P	P	L	S	S	S	S	S	M	M	LM	L	MS
P	P	P	P	L	P	P	P	P	P	P	L	P	P	P
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K
P	P	P	L	P	P	P	P	P	P	L	P	P	P	P
S	S	L	MS	S	MS	S	S	MS	L	P	S	MS	S	MS
MS	MS	P	CT	MS	M	M	LM	L	P	S	S	S	L	P
M	LM	L	P	P	S	S	M	M	LM	L	P	P	S	S
S	M	M	LM	L	P	P	S	S	M	M	LM	L	P	S
L	S	S	S	S	S	CT	P	S	S	P	M	M	LM	L
P	P	S	M	M	LM	L	MS	P	S	S	L	S	M	M
P	P	P	L	P	P	P	P	P	P	L	P	P	P	P

**Gambar 6.1 Jadwal Manual November 2017 Unit Farmasi**

Berdasarkan jadwal yang sudah ada, dilakukan perhitungan nilai *Jain's Fairness Index* dan didapatkan hasil sebesar 0.9236.

### 6.1.2 Jadwal Manual Unit Ruang Bayi dan NICU

Jadwal manual untuk Unit Ruang Bayi dan NICU yang sudah dimasukkan ke dalam program, akan dimunculkan seperti pada Gambar 6.2.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R
P	P	P	P	L	P	P	P	P	P	P	L	P	P	P
L	P	P	P	M	M	LM	L	P	S	P	P	P	L	P
LM	L	L	S	P	P	M	M	LM	L	S	S	S	M	M
P	CT	M	M	LM	L	P	P	Mid	L	S	M	M	LM	L
S	S	S	M	M	LM	L	S	S	P	M	M	LM	L	L
M	M	LM	L	S	S	S	P	M	M	LM	L	Mid	P	S
LM	L	S	S	P	L	L	CT	M	M	LM	L	CT	S	P
M	M	LM	L	L	P	M	M	LM	L	Mid	S	S	S	M
S	Mid	M	M	LM	L	S	Mid	CT	M	M	LM	L	P	S
Mid	P	P	L	Mid	S	Mid	L	P	S	L	P	M	M	LM
L	S	Mid	P	M	M	LM	S	S	Mid	M	M	LM	L	Mid
M	LM	L	Mid	S	M	M	LM	L	P	P	Mid	M	M	LM
P	M	M	LM	L	Mid	P	M	M	LM	L	L	P	L	M
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K
P	P	P	L	P	P	P	P	P	P	L	P	P	P	P
P	P	L	P	P	P	L	P	P	M	M	LM	L	CT	P
LM	L	P	P	CT	S	P	M	M	LM	L	S	S	P	S
P	P	S	S	P	L	M	M	LM	L	P	M	M	LM	L
M	M	Lm	L	S	S	P	L	S	P	L	P	CT	M	M
L	S	P	S	M	M	LM	L	LM	S	S	L	L	CT	CT
S	Mid	M	M	LM	L	S	S	Mid	M	M	LM	L	S	S
M	LM	L	CT	Mid	M	M	LM	L	S	P	LM	P	M	M
Mid	L	Mid	M	M	LM	L	S	P	Mid	S	M	M	LM	L
L	M	M	LM	L	P	Mid	Mid	M	M	LM	L	S	P	P
S	S	S	M	M	LM	L	M	M	LM	L	S	M	M	LM
L	M	M	LM	L	Mid	S	P	S	L	Mid	P	P	S	M
M	LM	L	Mid	S	M	M	LM	L	P	M	M	LM	L	Mid

Gambar 6.2 Jadwal Manual November 2017 Unit Ruang Bayi dan NICU

Berdasarkan jadwal yang sudah ada, dilakukan perhitungan nilai *Jain's Fairness Index* dan didapatkan hasil sebesar 0.8922.

### 6.1.3 Jadwal Manual Unit SIM Rekam Medis dan Registrasi

Jadwal manual untuk SIM Rekam Medis dan Registrasi yang sudah dimasukkan ke dalam program, akan dimunculkan seperti pada Gambar 6.3.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R
P	CT	P	P	L	MS	S	S	P	P	P	L	S	P	S
L	S	S	S	P	M	M	LM	L	S	S	MS	MS	M	M
S	P	P	MS	S	L	P	P	M	M	LM	L	CT	S	MS
S	M	M	LM	L	S	S	M	LM	L	S	P	P	MS	L
P	MS	S	M	M	LM	L	S	S	P	P	S	M	LM	L
M	LM	L	L	MS	P	P	P	MS	S	M	M	LM	L	P

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K
M	M	LM	L	MS	S	M	M	M	LM	L	S	S	S	P
LM	L	P	P	S	S	P	P	L	S	P	P	P	M	LM
S	S	S	L	M	M	LM	L	P	S	L	S	P	P	M
S	P	P	S	P	P	CT	P	S	M	M	LM	L	P	MS
P	P	S	MS	L	P	MS	S	P	P	S	M	M	LM	L
P	S	M	M	LM	L	S	S	S	P	MS	P	S	S	S

**Gambar 6.3 Jadwal Manual November 2017 Unit SIM Rekam Medis dan Registrasi**

Berdasarkan jadwal yang sudah ada, dilakukan perhitungan nilai *Jain's Fairness Index* dan didapatkan hasil sebesar 0.9009.

### 6.1.4 Jadwal Manual Unit Instalasi Gizi

Jadwal manual untuk SIM Rekam Medis dan Registrasi yang sudah dimasukkan ke dalam program, akan dimunculkan seperti pada Gambar 6.4.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R
P	P	P	P	L	P	P	P	P	P	L	L	P	P	P
P	L	S	P1	P	P1	S	S	P	CT	L	L	P1	P1	S
L	MD	MD	MD	P1	MD	CT	MD	S	P1	P1	L	L	S	MD
MD	P2	P1	L	S	S	P	CT	CT	L	MD	S	S	MD	P1
S	S	CT	S	MD	L	MD	P	S	S	S	P1	MD	L	CT
P1	P1	S	CT	L	S	P1	P1	P1	MD	S	MD	L	S	S
P2	S	S	S	S	L	S	S	S	S	S	L	S	P	P
P2	L	P2	CT	L	P2	P2	P2	L	P	P	S	P2	P2	P2
S	S	P	P	P2	L	S	S	S	P2	P2	P2	L	L	CT
P	P	MD	MD	P	L	S	P	P	P	MD	MD	L	S	P
L	CT	S	S	MD	M	M	M	LM	L	P	P	M	M	LM
S	S	M	M	M	LM	L	MD	S	S	S	L	CT	CT	M
MD	MD	P	P	L	P	P	S	M	M	LM	L	S	S	P
M	M	LM	CT	S	MD	S	P	P	P	M	M	LM	CT	S
P	P	MD	MD	P	S	L	M	M	LM	L	S	P	P	MD
P	P	P	S	L	PS	P	S	S	S	P	L	PS	S	P
S	S	S	P	PS	L	S	P	P	P	S	PS	L	P	S
S	S	S	P	PS	L	P	S	S	P	P	L	PS	S	S
CT	CT	CT	S	L	PS	S	CT	CT	S	S	PS	L	CT	CT

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K
P	P	P	MOD	P	P	P	P	L	L	L	P	P	P	P
S	S	P	P1	S	L	P	P2	S	S	P1	P2	P	P1	S
S	P	MD	MD	P1	MD	MD	S	CT	L	S	MD	MD	MD	P
MD	L	S	S	MD	S	S	CT	P	P1	MD	P1	P1	L	CT
P	MD	L	CT	CT	P1	CT	MD	MD	MD	L	S	S	S	MD
P1	P1	P1	S	S	L	P1	P1	P1	S	CT	S	L	CT	P1
P2	P2	P2	P	S	S	S	S	P2	P	P	L	S	P	S
L	S	S	S	P	P	CT	CT	S	P2	P2	S	S	P2	P2
S	S	S	P2	P2	P2	P2	S	L	S	S	CT	P2	S	S
MD	S	S	P	L	S	P	P	P	CT	L	CT	CT	CT	CT
L	CT	MD	MD	P	M	M	LM	L	S	S	MD	S	P	P
M	LM	L	S	S	P	P	CT	CT	M	M	LM	L	S	S
P	M	M	LM	L	S	S	M	M	LM	L	S	P	MD	MD
S	P	P	L	MD	MD	CT	CT	L	L	P	M	M	LM	L
L	MD	MD	M	M	LM	L	S	S	P	P	MD	M	M	M
P	P	S	PS	L	CT	CT	CT	PS	PS	L	S	P	P	P
S	S	P	L	PS	S	P	P	P	CT	L	PS	P	S	S
P	P	L	S	PS	S	P	P	P	S	PS	L	P	S	S
S	S	PS	P	L	P	S	S	S	P	L	PS	S	P	P

**Gambar 6.4 Jadwal Manual November 2017 Unit Instalasi Gizi**

Berdasarkan jadwal yang sudah ada, dilakukan perhitungan nilai *Jain's Fairness Index* dan didapatkan hasil sebesar 0.8516.

### 6.1.5 Jadwal Manual Unit Instalasi Rawat Jalan dan IGD

Jadwal manual untuk SIM Rekam Medis dan Registrasi yang sudah dimasukkan ke dalam program, akan dimunculkan seperti pada Gambar 6.5.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R
P	MS	S	MS	L	S	P	MS	L	CT	M	M	LM	S	S
MS	S	P	L	M	M	LM	MS	S	MS	L	L	S	MS	MS
S	P	S	P	L	P	MS	P	MS	CT	S	S	MS	P	L
L	S	MS	MS	S	CT	S	S	M	M	LM	L	P	S	S
S	S	M	M	LM	L	MS	S	S	S	MS	L	S	S	P
M	M	LM	L	P	S	S	CT	MS	S	P	L	M	M	LM
LM	S	L	P	L	S	S	S	P	P	MS	L	S	S	M
S	MS	S	S	L	MS	M	M	LM	S	P	P	L	MS	
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K
P	MS	L	S	S	M	L	P	P	CT	S	P	MS	M	M
S	S	P	P	CT	S	MS	S	P	L	P	MS	CT	MS	MS
CT	M	M	LM	S	S	MS	L	S	P	M	M	LM	L	L
MS	CT	S	S	M	L	P	MS	MS	S	S	S	M	L	L
MS	L	MS	M	M	LM	P	MS	S	L	CT	MS	S	S	P
L	P	MS	L	S	S	S	M	M	LM	L	S	MS	S	S
M	LM	L	L	MS	P	MS	S	S	MS	L	CT	S	P	MS
S	S	P	L	P	MS	S	S	S	M	M	LM	L	MS	S

**Gambar 6.5 Jadwal Manual November 2017 Unit Instalasi Rawat Jalan dan IGD**

Berdasarkan jadwal yang sudah ada, dilakukan perhitungan nilai *Jain's Fairness Index* dan didapatkan hasil sebesar 0.9184.

### 6.1.6 Jadwal Manual Unit Kamar Operasi

Jadwal manual untuk SIM Rekam Medis dan Registrasi yang sudah dimasukkan ke dalam program, akan dimunculkan seperti pada Gambar 6.6.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R
P	P	P	P	P	CT	CT	S	S	P	P	L	P	S	P
P	S	S	S	L	P	P	P	P	L	S	P	S	P	S
P	S	P	L	P	S	S	P	P	S	P	P	L	P	P
S	P	P	P	L	S	S	P	P	P	CT	L	S	P	P
L	P	S	S	P	P	P	L	P	S	S	P	P	S	L
P	L	P	P	P	P	L	S	S	P	P	P	P	CT	S
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
K	J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K
S	S	L	P	S	S	S	P	L	P	P	P	P	S	P
P	L	P	P	P	P	P	L	P	S	P	S	S	L	P
S	P	S	L	S	P	P	S	P	CT	L	S	S	P	S
P	P	P	P	L	S	S	S	P	P	L	P	S	S	S
P	S	S	P	P	P	L	P	S	P	P	P	P	P	CT
P	P	P	L	P	P	P	P	S	S	P	L	P	P	P

**Gambar 6.6 Jadwal Manual November 2017 Unit Kamar Operasi**

Berdasarkan jadwal yang sudah ada, dilakukan perhitungan nilai *Jain's Fairness Index* dan didapatkan hasil sebesar 0.8069.

## 6.2 Hasil Otomasi Penjadwalan dan Nilai Jain's Fairness Index dari Otomasi

Setelah menghitung nilai *Jain's Fairness Index* dari jadwal manual, akan dihitung nilai *Jain's Fairness Index* dari jadwal otomatis berdasarkan pola yang sudah didefinisikan pada sub bab 5.2. Selain itu juga akan dilakukan perhitungan terhadap hari libur berdasarkan jadwal otomasi di tiap unitnya.

### 6.2.1 Hasil Otomasi Unit Instalasi Farmasi

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K	J
P	P	L	P	P	P	P	P	P	L	P	P	P	P	P
S	S	L	S	S	S	S	S	L	S	S	S	S	S	S
M	M	LM	L	P	P	S	S	MS	S	M	M	LM	L	P
MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM
S	S	S	MS	M	M	LM	L	P	P	S	S	MS	MS	M
P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS
LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S
P	P	L	P	P	P	P	P	P	L	P	P	P	P	P

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
P	L	P	P	P	P	P	P	L	P	P	P	P	P	P	L
S	L	S	S	S	S	S	S	L	S	S	S	S	S	S	L
P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M
L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	S
M	LM	L	P	P	S	S	MS	S	M	M	LM	L	P	P	S
MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P
S	S	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM
P	L	P	P	P	P	P	P	L	P	P	P	P	P	P	L
Hari Minggu:	500			Hari Sabtu:	0		Hari Kerja:	0	Sum:	500		Sum Kuadrat:	250000.0		
Hari Minggu:	500			Hari Sabtu:	0		Hari Kerja:	0	Sum:	500		Sum Kuadrat:	250000.0		
Hari Minggu:	100			Hari Sabtu:	0		Hari Kerja:	50	Sum:	150		Sum Kuadrat:	22500.0		
Hari Minggu:	0			Hari Sabtu:	50		Hari Kerja:	50	Sum:	100		Sum Kuadrat:	10000.0		
Hari Minggu:	0			Hari Sabtu:	0		Hari Kerja:	75	Sum:	75		Sum Kuadrat:	5625.0		
Hari Minggu:	100			Hari Sabtu:	50		Hari Kerja:	25	Sum:	175		Sum Kuadrat:	30625.0		
Hari Minggu:	0			Hari Sabtu:	50		Hari Kerja:	50	Sum:	100		Sum Kuadrat:	10000.0		
Hari Minggu:	500			Hari Sabtu:	0		Hari Kerja:	0	Sum:	500		Sum Kuadrat:	250000.0		

Jfi: 0.665158371040724

**Gambar 6.7 Hasil Otomasi Unit Farmasi**

Dari hasil otomasi jadwal bulan Desember 2017 unit Farmasi pada Gambar 6.7, dihasilkan nilai JFI sebesar 0.6651. Nilai JFI ini lebih kecil dibandingkan dengan nilai JFI pada jadwal manual. Maka dari itu perlu dilakukan optimasi agar mendapatkan nilai yang lebih tinggi.

**Tabel 6.1 Perhitungan Libur Hasil Otomasi Unit Farmasi**

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
1	101	0	0	5	5
2	102	0	0	5	5
3	103	2	0	1	3
4	104	2	0	0	2
5	105	3	0	0	3
6	106	1	0	1	2
7	107	2	0	0	2
8	108	0	0	5	5

Berdasarkan perhitungan hari libur jadwal otomasi pada Tabel 6.1, diketahui bahwa jumlah libur yang dimiliki oleh unit Farmasi masih belum seimbang. Masih terdapat staf yang memiliki jumlah libur senilai 2. Nilai ini tidak sesuai dari total minimal libur yang harus dimiliki yaitu sejumlah 5 hari libur.

## 6.2.2 Hasil Otomasi Unit Ruang Bayi dan NICU

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K	J	
P	P	L	P	P	P	P	P	P	L	P	P	P	P	P	
M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	
MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	
S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	
P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	
LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	
M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	
MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	
S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	
P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	
LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	
M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	
MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K	J	Sb	M
P	L	P	P	P	P	P	P	L	P	P	P	P	P	P	L
P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M
L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS
M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S
MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P
S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM
P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M
L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS
M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S
MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P
S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM
P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M
L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS
M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S
MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P
S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM
P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M
L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS
M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S
MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P
S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM
P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M
L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS
Hari Minggu:	500			Hari Sabtu:	0		Hari Kerja:	0	Sum:	500	Sum Kuadrat:	250000.0			
Hari Minggu:	100			Hari Sabtu:	0		Hari Kerja:	50	Sum:	150	Sum Kuadrat:	22500.0			
Hari Minggu:	0			Hari Sabtu:	50		Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0			
Hari Minggu:	0			Hari Sabtu:	0		Hari Kerja:	75	Sum:	75	Sum Kuadrat:	5625.0			
Hari Minggu:	100			Hari Sabtu:	50		Hari Kerja:	25	Sum:	175	Sum Kuadrat:	30625.0			
Hari Minggu:	0			Hari Sabtu:	50		Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0			
Hari Minggu:	100			Hari Sabtu:	0		Hari Kerja:	50	Sum:	150	Sum Kuadrat:	22500.0			
Hari Minggu:	0			Hari Sabtu:	50		Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0			
Hari Minggu:	0			Hari Sabtu:	0		Hari Kerja:	75	Sum:	75	Sum Kuadrat:	5625.0			
Hari Minggu:	100			Hari Sabtu:	50		Hari Kerja:	25	Sum:	175	Sum Kuadrat:	30625.0			
Hari Minggu:	0			Hari Sabtu:	50		Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0			
Hari Minggu:	100			Hari Sabtu:	0		Hari Kerja:	50	Sum:	150	Sum Kuadrat:	22500.0			
Hari Minggu:	0			Hari Sabtu:	50		Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0			

Jfi: 0.6647727272727273

### Gambar 6.8 Hasil Otomasi Unit Ruang Bayi dan NICU

Dari hasil otomasi jadwal bulan Desember 2017 unit Ruang Bayi dan NICU pada Gambar 6.8, dihasilkan nilai JFI sebesar 0.6647. Nilai ini lebih kecil dibandingkan dengan nilai JFI hasil manual. Maka dari itu perlu dilakukan optimasi untuk meningkatkan nilai JFI.

Tabel 6.2 Perhitungan Libur Hasil Otomasi Unit Ruang Bayi dan NICU

No	Kode Staf	Jumlah Libur		Total
		Hari Kerja	Sabtu Minggu	
1	201	0	0	5

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
2	201	2	0	1	3
3	201	2	1	0	3
4	201	3	0	0	3
5	201	1	1	1	3
6	201	2	1	0	3
7	201	2	0	1	3
8	201	2	1	0	3
9	201	3	0	0	3
10	201	1	1	1	3
11	201	2	1	0	3
12	201	2	0	1	3
13	201	2	1	0	3

Berdasarkan perhitungan hari libur jadwal otomasi pada Tabel 6.2, diketahui bahwa jumlah libur yang dimiliki oleh unit Ruang Bayi dan NICU masih belum seimbang. Masih terdapat staf yang memiliki jumlah libur senilai 3. Nilai ini tidak sesuai dari total minimal libur yang harus dimiliki yaitu sejumlah 5 hari libur.

### 6.2.3 Hasil Otomasi Unit SIM Rekam Medis dan Registrasi

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K	J
P	S	S	S	MS	MS	M	M	LM	L	P	P	P	S	S
P	P	P	S	S	S	MS	MS	M	M	LM	L	P	P	P
LM	L	P	P	P	S	S	S	MS	MS	M	M	LM	L	P
M	M	LM	L	P	P	P	S	S	S	MS	MS	M	M	LM
MS	MS	M	M	LM	L	P	P	P	S	S	S	MS	MS	M
S	S	P	P	M	M	LM	L	P	P	P	S	S	S	P
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K	J	Sb
S	MS	MS	M	M	LM	L	P	P	P	S	S	S	MS	MS
S	S	S	MS	MS	M	M	LM	L	P	P	P	S	S	S
P	P	S	S	S	MS	MS	M	M	LM	L	P	P	P	S
L	P	P	P	S	S	S	MS	MS	M	M	LM	L	P	P
M	LM	L	P	P	P	S	S	S	MS	MS	M	M	LM	L
P	M	M	LM	L	P	P	P	S	S	S	P	P	M	M



Hari Minggu:	100	Hari Sabtu:	0	Hari Kerja:	25	Sum:	125	Sum Kuadrat:	15625.0
Hari Minggu:	100	Hari Sabtu:	0	Hari Kerja:	25	Sum:	125	Sum Kuadrat:	15625.0
Hari Minggu:	0	Hari Sabtu:	50	Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0
Hari Minggu:	0	Hari Sabtu:	50	Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0
Hari Minggu:	0	Hari Sabtu:	50	Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0
Hari Minggu:	0	Hari Sabtu:	0	Hari Kerja:	50	Sum:	50	Sum Kuadrat:	2500.0

Jfi: 0.9411764705882353

### Gambar 6.9 Hasil Otomasi Unit SIM RM dan Registrasi

Dari hasil otomasi jadwal bulan Desember 2017 unit SIM Rekam Medis dan Registrasi pada Gambar 6.9, dihasilkan nilai JFI sebesar 0.9411. Nilai ini sudah lebih besar dibandingkan dengan nilai JFI jadwal. Namun tetap perlu dilakukan optimasi untuk mencapai nilai JFI yang lebih maksimal.

**Tabel 6.3 Perhitungan Libur Hasil Otomasi Unit SIM RM dan Registrasi**

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
1	301	2	1	0	3
2	302	2	1	0	3
3	303	2	0	0	2
4	304	1	0	1	2
5	305	1	0	1	2
6	306	2	1	0	3

Berdasarkan perhitungan hari libur jadwal otomasi pada Tabel 6.3, diketahui bahwa jumlah libur yang dimiliki oleh unit SIM Rekam Medis dan Registrasi masih belum seimbang. Masih terdapat staf yang memiliki jumlah libur senilai 2. Nilai ini masih jauh dari total minimal libur yang harus dimiliki yaitu sejumlah 5 hari libur.

### 6.2.4 Hasil Otomasi Unit Instalasi Gizi

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K	J	
P	P	L	P	P	P	P	P	P	L	P	P	P	P	P	
P1	P	S	P1	P	L	S	S	MD	MD	P1	P	S	P1	P	
MD	MD	P1	P	S	P1	P	L	S	S	MD	MD	P1	P	S	
S	S	MD	MD	P1	P	S	P1	P	L	S	S	MD	MD	P1	
P	L	S	S	MD	MD	P1	P	S	P1	P	L	S	S	MD	
S	S	S	P1	P1	L	P1	P1	S	S	S	P1	P1	L	P1	
P2	P	S	S	L	P	S	P2	P	MS	P2	P	S	S	L	
P	MS	P2	P	S	S	L	P	S	P2	P	MS	P2	P	S	
S	P2	P	MS	P2	P	S	S	L	P	S	P2	P	MS	P2	
S	S	P	P	MD	MD	L	S	S	P	P	MD	MD	L	S	
M	M	LM	L	P	P	S	S	P	P	M	M	LM	L	P	
P	P	M	M	LM	L	P	P	S	S	P	P	M	M	LM	
S	S	P	P	M	M	LM	L	P	P	S	S	P	P	M	
P	P	S	S	P	P	M	M	LM	L	P	P	S	S	P	
LM	L	P	P	S	S	MD	MD	M	M	LM	L	P	P	S	
S	S	L	PS	S	S	P	P	PS	L	P	P	S	S	S	
P	P	PS	L	P	P	S	S	S	L	PS	S	S	P	P	
P	P	PS	L	P	P	S	S	S	L	PS	S	S	P	P	
S	S	L	PS	P	S	P	P	PS	L	P	P	S	S	S	
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K	J	Sb	M
P	L	P	P	P	P	P	P	L	P	P	P	P	P	P	L
L	S	S	MD	MD	P1	P	S	P1	P	L	S	S	MD	MD	P1
P1	P	L	S	S	MD	MD	P1	P	S	P1	P	L	S	S	MD
P	S	P1	P	L	S	S	MD	MD	P1	P	S	P1	P	L	S
MD	P1	P	S	P1	P	L	S	S	MD	MD	P1	P	S	P1	P
P1	S	S	S	P1	P1	L	P1	P1	S	S	S	P1	P1	L	P1
P	S	P2	P	MS	P2	P	S	S	L	P	S	P2	P	MS	P2
S	L	P	S	P2	P	MS	P2	P	S	S	L	P	S	P2	P
P	S	S	L	P	S	P2	P	MS	P2	P	S	S	L	P	S
S	P	P	MD	MD	L	S	S	P	P	MD	MD	L	S	S	P
P	S	S	P	P	M	M	LM	L	P	P	S	S	P	P	M
L	P	P	S	S	P	P	M	M	LM	L	P	P	S	S	P
M	LM	L	P	P	S	S	P	P	M	M	LM	L	P	P	S
P	M	M	LM	L	P	P	S	S	P	P	M	M	LM	L	P
S	MD	MD	M	M	LM	L	P	P	S	S	P	MD	MD	M	LM
S	L	PS	S	S	P	P	P	PS	L	P	P	S	S	P	L
P	PS	L	P	P	S	S	S	L	PS	S	S	P	P	P	PS
P	PS	L	P	P	S	S	S	L	PS	S	S	P	P	P	PS
S	L	PS	S	S	P	P	P	PS	L	P	P	S	S	P	L
Hari Minggu:	500		Hari Sabtu:	0	Hari Kerja:	0	Sum:	500	Sum Kuadrat:	250000.0					
Hari Minggu:	0	Hari Sabtu:	50	Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0						
Hari Minggu:	0	Hari Sabtu:	0	Hari Kerja:	75	Sum:	75	Sum Kuadrat:	5625.0						
Hari Minggu:	100	Hari Sabtu:	50	Hari Kerja:	25	Sum:	175	Sum Kuadrat:	30625.0						
Hari Minggu:	0	Hari Sabtu:	50	Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0						
Hari Minggu:	0	Hari Sabtu:	50	Hari Kerja:	75	Sum:	125	Sum Kuadrat:	15625.0						
Hari Minggu:	0	Hari Sabtu:	0	Hari Kerja:	75	Sum:	75	Sum Kuadrat:	5625.0						
Hari Minggu:	100	Hari Sabtu:	0	Hari Kerja:	50	Sum:	150	Sum Kuadrat:	22500.0						
Hari Minggu:	0	Hari Sabtu:	50	Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0						
Hari Minggu:	100	Hari Sabtu:	0	Hari Kerja:	100	Sum:	100	Sum Kuadrat:	10000.0						
Hari Minggu:	100	Hari Sabtu:	0	Hari Kerja:	50	Sum:	150	Sum Kuadrat:	22500.0						
Hari Minggu:	0	Hari Sabtu:	50	Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0						
Hari Minggu:	0	Hari Sabtu:	0	Hari Kerja:	75	Sum:	75	Sum Kuadrat:	5625.0						
Hari Minggu:	100	Hari Sabtu:	50	Hari Kerja:	25	Sum:	175	Sum Kuadrat:	30625.0						
Hari Minggu:	0	Hari Sabtu:	50	Hari Kerja:	50	Sum:	100	Sum Kuadrat:	10000.0						
Hari Minggu:	300	Hari Sabtu:	0	Hari Kerja:	50	Sum:	350	Sum Kuadrat:	122500.0						
Hari Minggu:	200	Hari Sabtu:	0	Hari Kerja:	50	Sum:	250	Sum Kuadrat:	62500.0						
Hari Minggu:	200	Hari Sabtu:	0	Hari Kerja:	50	Sum:	250	Sum Kuadrat:	62500.0						
Hari Minggu:	300	Hari Sabtu:	0	Hari Kerja:	50	Sum:	350	Sum Kuadrat:	122500.0						

Jfi: 0.700040176778224

**Gambar 6.10 Hasil Otomasi Unit Gizi**

Dari hasil otomasi jadwal bulan Desember 2017 unit Instalasi Gizi pada Gambar Gambar 6.10, dihasilkan nilai JFI sebesar

sebesar 0.7004. Nilai ini lebih kecil dibandingkan dengan nilai JFI hasil penjadwalan manual. Maka dari itu perlu dilakukan adanya optimasi untuk meningkatkan nilai JFI.

**Tabel 6.4 Perhitungan Libur Hasil Otomasi Unit Gizi**

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
1	401	0	0	5	5
2	402	2	1	0	3
3	403	3	0	0	3
4	404	1	1	1	3
5	405	2	1	0	3
6	406	3	1	0	4
7	407	3	0	0	3
8	408	2	0	1	3
9	409	2	1	0	3
10	410	4	0	0	4
11	411	2	0	1	3
12	412	2	1	0	3
13	413	3	0	0	3
14	414	1	1	1	3
15	415	2	1	0	3
16	416	2	0	3	5
17	417	2	0	2	4
18	418	2	0	2	4
19	419	2	0	3	5

Berdasarkan perhitungan hari libur jadwal otomasi pada Tabel 6.4, diketahui bahwa jumlah libur yang dimiliki oleh unit Gizi masih belum seimbang. Masih terdapat staf yang memiliki jumlah libur senilai 3. Nilai ini tidak sesuai dari total minimal libur yang harus dimiliki yaitu sejumlah 5 hari libur.

### 6.2.5 Hasil Otomasi Unit Instalasi Rawat Jalan dan IGD

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
J	Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K	J	
M	M	LM	L	P	P	S	S	S	S	S	L	MD	MS	S	
MS	S	M	M	LM	L	P	P	S	S	S	S	S	L	MD	
L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S	S	
S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	
S	S	S	S	L	MD	MS	S	M	M	LM	L	P	P	S	
P	S	S	S	S	S	L	MD	MS	S	M	M	LM	L	P	
L	P	P	S	S	S	S	S	L	MD	MS	S	M	M	LM	
M	LM	L	P	P	S	S	S	S	S	L	MD	MS	S	M	
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Sb	M	Sn	S1	R	K	J	Sb	M	Sn	S1	R	K	J	Sb	M
M	M	LM	L	P	P	S	S	S	S	S	L	MD	MS	S	M
MS	S	M	M	LM	L	P	P	S	S	S	S	S	L	MD	MS
L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S	S	L
S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S
S	S	S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S
P	S	S	S	S	S	L	MD	MS	S	M	M	LM	L	P	P
L	P	P	S	S	S	S	S	L	MD	MS	S	M	M	LM	L
M	LM	L	P	P	S	S	S	S	S	L	MD	MS	S	M	M
Hari Minggu:	0			Hari Sabtu:	0			Hari Kerja:	100	Sum:	100	Sum Kuadrat:	10000.0		
Hari Minggu:	0			Hari Sabtu:	0			Hari Kerja:	100	Sum:	100	Sum Kuadrat:	10000.0		
Hari Minggu:	100			Hari Sabtu:	100			Hari Kerja:	50	Sum:	250	Sum Kuadrat:	62500.0		
Hari Minggu:	200			Hari Sabtu:	0			Hari Kerja:	50	Sum:	250	Sum Kuadrat:	62500.0		
Hari Minggu:	0			Hari Sabtu:	0			Hari Kerja:	100	Sum:	100	Sum Kuadrat:	10000.0		
Hari Minggu:	0			Hari Sabtu:	0			Hari Kerja:	100	Sum:	100	Sum Kuadrat:	10000.0		
Hari Minggu:	200			Hari Sabtu:	100			Hari Kerja:	25	Sum:	325	Sum Kuadrat:	105625.0		
Hari Minggu:	100			Hari Sabtu:	0			Hari Kerja:	75	Sum:	175	Sum Kuadrat:	30625.0		

Jfi: 0.8132780082987562

**Gambar 6.11 Hasil Otomasi Unit Rawat Jalan dan IGD**

Dari hasil otomasi jadwal bulan Desember 2017 unit Instalasi Rawat Jalan dan IGD pada Gambar Gambar 6.11, dihasilkan nilai JFI sebesar 0.8132. Nilai ini lebih kecil dibandingkan dengan hasil penjadwalan manual. Maka dari itu perlu dilakukan adanya optimasi untuk meningkatkan nilai JFI.

**Tabel 6.5 Perhitungan Libur Hasil Otomasi Unit Rawat Jalan dan IGD**

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
1	501	3	1	0	4
2	502	4	0	0	4
3	503	3	0	1	4
4	504	2	0	2	4
5	505	2	0	2	4

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
6	506	4	0	0	4
7	507	2	2	0	4
8	508	1	2	0	3

Berdasarkan perhitungan hari libur jadwal otomasi pada Tabel 6.5, diketahui bahwa jumlah libur yang dimiliki oleh unit Rawat Jalan dan IGD masih belum seimbang. Masih terdapat staf yang memiliki jumlah libur senilai 3. Nilai ini tidak sesuai dari total minimal libur yang harus dimiliki yaitu sejumlah 5 hari libur.

### 6.2.6 Hasil Otomasi Unit Kamar Operasi

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	
P	P	L	P	S	L	P	P	S	P	P	P	S	P	S	
S	P	L	S	P	S	L	P	P	P	S	P	P	S	P	
S	S	P	P	S	P	S	L	P	P	S	S	P	P	S	
P	S	P	P	P	S	P	S	L	P	P	S	P	P	P	
P	P	P	P	P	P	S	P	S	L	P	P	S	P	P	
L	P	P	S	P	P	P	S	P	L	L	P	P	S	P	
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
L	P	P	S	S	P	P	S	L	S	L	P	P	S	S	P
S	P	P	P	S	S	P	P	L	P	S	L	P	P	S	P
P	L	L	P	P	S	S	P	P	S	P	S	L	P	P	P
S	L	S	L	P	P	S	P	P	P	S	P	S	L	P	P
P	P	P	S	L	P	P	S	P	P	P	S	P	S	L	L
P	P	S	P	S	L	P	P	P	P	P	P	S	P	S	L
Hari Minggu: 200		Hari Sabtu: 50		Hari Kerja: 50		Sum: 300		Sum Kuadrat: 90000.0							
Hari Minggu: 200		Hari Sabtu: 0		Hari Kerja: 50		Sum: 250		Sum Kuadrat: 62500.0							
Hari Minggu: 100		Hari Sabtu: 0		Hari Kerja: 75		Sum: 175		Sum Kuadrat: 30625.0							
Hari Minggu: 100		Hari Sabtu: 50		Hari Kerja: 50		Sum: 200		Sum Kuadrat: 40000.0							
Hari Minggu: 200		Hari Sabtu: 50		Hari Kerja: 25		Sum: 275		Sum Kuadrat: 75625.0							
Hari Minggu: 200		Hari Sabtu: 0		Hari Kerja: 75		Sum: 275		Sum Kuadrat: 75625.0							

Jfi: 0.9685587089593768

#### Gambar 6.12 Hasil Otomasi Unit Kamar Operasi

Dari hasil otomasi jadwal bulan Desember 2017 unit Kamar Operasi pada Gambar 6.12, dihasilkan nilai JFI sebesar 09685. Nilai ini sudah lebih baik dibandingkan dengan nilai JFI hasil manual. Namun begitu, optimasi tetap perlu dilakukan untuk mendapatkan nilai JFI yang maksimal.

Tabel 6.6 Perhitungan Libur Hasil Otomasi Kamar Operasi

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
1	601	2	1	2	5
2	602	2	0	2	4
3	603	3	0	1	4
4	604	2	1	1	4
5	605	1	1	2	4
6	606	3	0	2	5

Berdasarkan perhitungan hari libur jadwal otomasi pada Tabel 6.6, diketahui bahwa jumlah libur yang dimiliki oleh unit Kamar Operasi sudah cukup seimbang meski masih terdapat staf yang memiliki jumlah libur senilai 4. Nilai ini tidak sesuai dari total minimal libur yang harus dimiliki yaitu sejumlah 5 hari libur.

### 6.3 Hasil Optimasi dengan Menggunakan Algoritma Simulated Annealing Hyper-Heuristic

Pada bagian ini akan dijabarkan hasil eksperimen dari implementasi algoritma *Simulated Annealing Hyper-Heuristic* dalam membuat penjadwalan. Penjabaran eksperimen meliputi hasil penjadwalan optimasi, nilai JFI dari hasil optimasi, diagram *trajectory*, perbandingan dengan menggunakan algoritma *Hill Climbing*, pemetaan *Box Plot* serta perbandingan antara JFI manual, otomasi dan optimasi.

#### 6.3.1 Hasil Penjadwalan dengan Optimasi

Pada bagian ini akan dijelaskan hasil pembagian shift tiap karyawan serta perhitungan hari libur berdasarkan penjadwalan optimasi dengan menggunakan algoritma *Simulated Annealing Hyper-Heuristic*.

### 6.3.1.1 Hasil Optimasi Unit Instalasi Farmasi

Optimasi akan menghasilkan jadwal baru berdasarkan nilai terbaik dari JFI yang didapatkan. Hasil penjadwalan dengan optimasi pada unit Farmasi dapat dilihat pada Gambar 6.13.

No	Kode Staf	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	101	P	P	S	P	P	P	P	P	P	S	P	P	P	P	P
2	102	S	S	L	S	S	S	S	S	S	M	P	L	S	S	S
3	103	M	M	LM	M	L	P	S	S	P	L	M	P	LM	L	P
4	104	MS	L	M	L	LM	P	P	P	S	L	S	M	M	M	LM
5	105	S	S	S	L	M	M	LM	L	P	P	S	S	P	L	M
6	106	P	P	L	S	S	L	M	M	LM	S	L	P	S	S	L
7	107	LM	L	P	P	P	S	L	L	M	L	LM	L	L	P	S
8	108	P	P	L	P	P	L	P	P	L	L	P	S	P	P	P

No	Kode Staf	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
		16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	101	P	L	P	P	P	P	P	P	L	P	P	P	P	P	L	P
2	102	S	S	S	S	P	S	S	S	L	S	L	S	S	S	S	L
3	103	P	L	S	MS	MS	M	L	LM	L	LM	L	S	S	MS	M	L
4	104	L	P	P	S	S	MS	L	M	L	L	S	P	P	S	I	S
5	105	M	LM	M	P	L	S	S	L	S	M	M	LM	L	P	P	S
6	106	S	M	L	LM	S	P	P	S	S	P	P	M	M	LM	P	L
7	107	L	S	MS	M	M	LM	M	P	P	S	S	MS	MS	M	L	LM
8	108	P	L	P	P	P	P	P	P	M	P	P	P	P	P	P	M

Gambar 6.13 Hasil Jadwal Optimasi Unit Farmasi

Berdasarkan jadwal hasil optimasi, akan didapatkan informasi mengenai jumlah hari libur serta pembagian jumlah shift pada tiap staf.

Tabel 6.7 Hasil Perhitungan Libur Optimasi Unit Farmasi

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
1	101	0	1	2	3
2	102	2	0	3	5
3	103	4	0	4	8
4	104	3	2	2	7

No	Kode Staf	Jumlah Libur			
		Hari Kerja	Sabtu	Minggu	Total
5	105	5	1	0	6
6	106	4	0	2	6
7	107	4	3	1	8
8	108	1	1	3	5

**Tabel 6.8 Jumlah Shift Tiap Staf Hasil Optimasi Unit Farmasi**

No	Kode Staf	Tipe Shift					
		P	S	M	LM	MS	L
1	101	26	2	0	0	0	3
2	102	2	23	1	0	0	5
3	103	5	5	6	4	3	8
4	104	7	7	5	2	2	7
5	105	6	9	7	3	0	6
6	106	8	9	5	3	0	6
7	107	6	5	5	4	3	8
8	108	23	1	2	0	0	5

Dari tabel Tabel 6.7 diketahui bahwa jumlah libur berkisar antara 3 hingga 8 hari dalam satu bulan. Hasil libur yang dihasilkan cenderung tidak seimbang. Namun, berdasarkan pembagian shift per staf pada Tabel 6.8, nilai JFI yang dihasilkan dengan menggunakan algoritma *Simulated Annealing* adalah 0.9323. Hasil ini lebih baik dibandingkan dengan nilai JFI jadwal manual maupun otomatisasi.

### 6.3.1.2 Hasil Optimasi Unit Ruang Bayi dan NICU

Optimasi akan menghasilkan jadwal baru berdasarkan nilai terbaik dari JFI yang didapatkan. Hasil penjadwalan dengan optimasi pada unit Ruang Bayi dan NICU dapat dilihat pada Gambar 6.14.



No	Kode Staf	J 1	Sb 2	M 3	Sn 4	Sl 5	R 6	K 7	J 8	Sb 9	M 10	Sn 11	Sl 12	R 13	K 14	J 15
1	201	P	P	L	P	P	P	P	P	P	MS	P	P	P	P	P
2	202	M	M	LM	L	P	P	S	S	MS	L	M	M	LM	L	P
3	203	MS	MS	M	M	LM	L	P	P	S	S	MS	L	M	M	LM
4	204	S	S	MS	L	M	M	LM	L	P	P	S	S	MS	L	M
5	205	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	L
6	206	LM	L	P	P	S	S	L	MS	M	M	LM	L	P	P	S
7	207	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P
8	208	MS	MS	M	M	LM	L	P	P	S	S	MS	L	M	M	LM
9	209	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M
10	210	P	P	S	S	MS	L	M	M	LM	L	P	P	S	S	MS
11	211	LM	L	P	P	S	S	MS	L	M	M	LM	L	P	P	S
12	212	M	M	LM	L	P	P	S	S	MS	L	M	M	LM	L	P
13	213	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM

No	Kode Staf	Sb 16	M 17	Sn 18	Sl 19	R 20	K 21	J 22	Sb 23	M 24	Sn 25	Sl 26	R 27	K 28	J 29	Sb 30	M 31
1	201	P	L	P	P	P	P	P	P	L	P	P	P	P	P	P	L
2	202	P	S	S	L	L	M	M	LM	L	P	P	S	S	MS	MS	M
3	203	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	L
4	204	M	LM	L	P	P	S	L	LM	L	M	M	LM	P	P	P	S
5	205	L	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P
6	206	S	L	MS	M	M	LM	L	P	L	S	S	MS	L	M	M	LM
7	207	P	S	S	MS	L	M	M	L	L	P	P	S	S	L	MS	M
8	208	MS	P	P	S	S	L	P	M	M	LM	L	P	MS	S	S	MS
9	209	M	LM	L	P	P	S	S	L	L	M	M	LM	L	P	P	S
10	210	P	M	M	LM	L	P	L	S	S	L	MS	M	M	LM	L	P
11	211	S	L	MS	M	M	LM	S	P	P	S	S	MS	MS	M	M	LM
12	212	L	S	S	L	MS	M	M	LM	P	P	P	S	S	L	MS	M
13	213	L	P	P	S	S	L	MS	M	M	LM	L	P	P	S	S	L

**Gambar 6.14 Hasil Jadwal Optimasi Unit Ruang Bayi dan NICU**

Berdasarkan jadwal hasil optimasi, akan didapatkan informasi mengenai jumlah hari libur serta pembagian jumlah shift pada tiap staf.

**Tabel 6.9 Hasil Perhitungan Libur Optimasi Unit Ruang Bayi dan NICU**

No	Kode Staf	Jumlah Libur			
		Hari Kerja	Sabtu	Minggu	Total
1	201	0	0	4	4
2	202	4	0	2	6

No	Kode Staf	Jumlah Libur			
		Hari Kerja	Sabtu	Minggu	Total
3	203	3	1	1	5
4	204	5	0	1	6
5	205	2	2	1	5
6	206	4	1	2	7
7	207	4	1	1	6
8	208	4	0	0	4
9	209	3	1	1	5
10	210	4	1	1	6
11	211	2	1	1	4
12	212	4	1	1	6
13	213	3	1	1	5

**Tabel 6.10 Jumlah Shift Tiap Staf Hasil Optimasi Unit Ruang Bayi dan NICU**

No	Kode Staf	Tipe Shift					
		P	S	M	LM	MS	L
1	201	26	0	0	0	1	4
2	202	6	6	7	3	3	6
3	203	6	6	6	3	5	5
4	204	7	6	6	4	2	6
5	205	7	6	6	3	4	5
6	206	5	6	6	4	3	7
7	207	6	6	7	2	4	6
8	208	6	6	6	3	6	4
9	209	6	7	6	3	4	5
10	210	7	6	6	3	3	6
11	211	6	7	6	4	4	4
12	212	6	6	7	3	3	6
13	213	6	6	6	3	5	5

Dari tabel Tabel 6.9 diketahui bahwa jumlah libur berkisar antara 3 hingga 8 hari dalam satu bulan. Hasil libur yang dihasilkan cenderung tidak seimbang. Namun, berdasarkan

pembagian shift per staf pada Tabel 6.10, nilai JFI yang dihasilkan dengan menggunakan algoritma *Simulated Annealing* adalah 0.9285. Hasil ini lebih baik dibandingkan dengan nilai JFI jadwal manual maupun otomatisasi.

### 6.3.1.3 Hasil Optimasi Unit SIM Rekam Medis dan Registrasi

Optimasi akan menghasilkan jadwal baru berdasarkan nilai terbaik dari JFI yang didapatkan. Hasil penjadwalan dengan optimasi pada unit Ruang Bayi dan NICU dapat dilihat pada Gambar 6.15.

No	Kode Staf	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	301	P	S	S	S	MS	MS	M	M	LM	L	P	L	P	S	S
2	302	P	P	P	S	S	S	L	L	M	M	LM	P	P	P	P
3	303	LM	L	P	P	P	S	S	S	L	P	M	M	LM	L	P
4	304	M	M	LM	P	P	P	P	S	S	S	L	S	M	M	LM
5	305	L	MS	M	M	LM	L	P	P	P	S	S	S	L	MS	M
6	306	S	S	P	L	M	M	LM	L	P	L	P	L	S	S	P

No	Kode Staf	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
		16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	301	S	L	MS	M	M	LM	L	P	P	P	S	S	S	MS	L	M
2	302	S	S	S	MS	L	M	M	LM	L	P	P	P	P	S	S	L
3	303	P	P	S	S	S	P	L	M	M	LM	L	P	P	P	S	S
4	304	L	P	P	P	S	S	S	L	L	M	M	LM	L	P	P	P
5	305	M	LM	L	P	L	L	S	S	S	S	MS	M	M	LM	L	P
6	306	P	M	M	LM	P	P	P	P	S	L	S	P	P	M	M	LM

Gambar 6.15 Hasil Jadwal Optimasi Unit SIM Rekam Medis

Berdasarkan jadwal hasil optimasi, akan didapatkan informasi mengenai jumlah hari libur serta pembagian jumlah shift pada tiap staf.

Tabel 6.11 Hasil Perhitungan Libur Optimasi Unit SIM Rekam Medis dan Registrasi

No	Kode Staf	Jumlah Libur			
		Hari Kerja	Sabtu	Minggu	Total
1	301	2	1	2	5

No	Kode Staf	Jumlah Libur			
		Hari Kerja	Sabtu	Minggu	Total
2	302	3	0	2	5
3	303	3	2	0	5
4	304	2	2	1	5
5	305	6	1	0	7
6	306	4	0	1	5

**Tabel 6.12 Jumlah Shift Tiap Staf Hasil Optimasi Unit SIM Rekam Medis dan Registrasi**

No	Kode Staf	Tipe Shift					
		P	S	M	LM	MS	L
1	301	6	9	5	2	4	5
2	302	10	9	4	2	1	5
3	303	11	8	4	3	0	5
4	304	10	7	6	3	0	5
5	305	5	7	6	3	3	7
6	306	11	6	6	3	0	5

Dari tabel Tabel 6.11 diketahui bahwa jumlah libur berkisar antara 5 hingga 7 hari dalam satu bulan. Hasil libur yang dihasilkan sudah cukup seimbang dengan adanya 1 staf yang memiliki libur lebih dari 5. Berdasarkan pembagian shift per staf pada Tabel 6.12, nilai JFI yang dihasilkan dengan menggunakan algoritma *Simulated Annealing* adalah 0.96433. Hasil ini lebih baik dibandingkan dengan nilai JFI jadwal manual maupun otomatisasi.

### 6.3.1.4 Hasil Optimasi Unit Gizi

Optimasi akan menghasilkan jadwal baru berdasarkan nilai terbaik dari JFI yang didapatkan. Hasil penjadwalan dengan optimasi pada unit Ruang Bayi dan NICU dapat dilihat pada Gambar 6.16.

No	Kode	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	401	P	P	L	P	P	P	P	P	P	P	P	P	P	P	P
2	402	P1	P	S	P1	P	L	S	S	MD	MD	P1	P	S	P1	P
3	403	MD	MD	P1	P	S	P1	P	L	S	S	MD	MD	P1	P	S
4	404	S	S	MD	MD	P1	P	S	P1	P	L	S	S	MD	MD	P1
5	405	P	L	S	S	MD	MD	P1	P	S	P1	P	L	S	S	MD
6	406	S	S	S	P1	P1	L	P1	P1	S	L	S	P1	P1	L	P1
7	407	P2	P	S	S	L	P	S	P2	P	L	P2	P	S	S	L
8	408	P	L	L	L	S	S	L	P	S	P2	P	L	P2	P	S
9	409	S	P2	P	L	P2	P	S	S	L	P	S	L	P	P	P2
10	410	S	S	P	P	MD	MD	L	S	S	P	P	MD	MD	L	S
11	411	M	M	LM	L	P	P	S	S	P	L	M	M	LM	L	P
12	412	P	P	M	M	LM	L	P	P	S	S	P	P	M	M	LM
13	413	S	S	P	P	M	M	LM	L	P	L	S	S	P	P	M
14	414	P	P	S	S	P	P	M	M	LM	L	S	P	S	S	P
15	415	LM	L	P	P	S	S	MD	MD	M	M	LM	P2	P	P	S
16	416	S	S	P2	PS	S	S	P	P	P	PS	L	P	P	S	S
17	417	P	P	PS	L	P	P	S	S	S	P	PS	S	S	L	P
18	418	P	P	PS	P	P	P	S	S	S	S	PS	S	S	P	P
19	419	S	S	L	PS	S	S	P	P	P	PS	L	P	P	S	S

No	Kode	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
		16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	401	P	L	P	P	P	P	P	P	P	L	P	P	P	P	L	L
2	402	L	S	S	MD	MD	P1	P	S	P1	P	L	S	S	MD	MD	L
3	403	P1	L	L	S	S	MD	MD	P1	P	S	P1	P	L	S	S	MD
4	404	P	S	L	P	L	S	S	MD	MD	P1	P	S	P1	P	P	S
5	405	MD	P1	P	S	P1	P	P2	S	S	MD	MD	P1	P	S	P1	P
6	406	P1	P	S	S	P1	P1	L	P1	P1	S	S	S	P1	P1	L	P1
7	407	P	S	P2	P	MS	P2	P	S	S	L	P	S	P2	P	L	P2
8	408	S	S	P	S	P2	P	L	P2	P	S	S	L	P	S	P2	P
9	409	P	S	L	P	S	L	P	L	P2	P2	P	S	L	L	P	S
10	410	S	P	P	MD	MD	S	S	S	P	P	MD	MD	MD	S	S	P
11	411	P	S	S	P	P	M	M	LM	L	P	P	S	S	P	P	M
12	412	L	P	P	S	S	P	P	M	M	LM	L	P	P	S	P	P
13	413	M	LM	L	P	P	S	S	P	L	M	M	LM	L	P	P	S
14	414	P	M	M	LM	L	P	P	S	S	P	P	M	M	LM	L	P
15	415	S	MD	MD	M	M	LM	L	P	P	S	S	P	L	M	M	LM
16	416	S	L	PS	S	S	P	P	P	PS	L	P	P	S	S	S	L

Gambar 6.16 Hasil Jadwal Optimasi Unit Gizi

Berdasarkan jadwal hasil optimasi, akan didapatkan informasi mengenai jumlah hari libur serta pembagian jumlah shift pada tiap staf.

**Tabel 6.13 Hasil Perhitungan Libur Optimasi Unit Gizi**

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
1	401	1	1	3	5
2	402	2	1	1	4
3	403	3	0	1	4
4	404	2	0	1	3
5	405	1	1	0	2
6	406	3	1	1	5
7	407	3	1	1	5
8	408	5	1	1	7
9	409	5	1	1	7
10	410	2	0	0	2
11	411	2	0	2	4
12	412	2	1	0	3
13	413	3	0	2	5
14	414	1	1	1	3
15	415	2	1	0	3
16	416	2	0	2	4
17	417	3	0	1	4
18	418	1	0	1	2
19	419	1	0	2	3

**Tabel 6.14 Jumlah Shift Tiap Staf Hasil Optimasi Gizi**

No	Kode Staf	Tipe Shift							
		P	P1	P2	S	MD	LM	L	PS
1	401	26	0	0	0	0	0	0	5
2	402	6	6	0	9	0	6	0	4
3	403	5	6	0	9	0	7	0	4
4	404	7	5	0	10	0	6	0	3
5	405	7	6	1	9	0	6	0	2
6	406	1	15	0	10	0	0	0	5

No	Kode Staf	Tipe Shift							
		P	P1	P2	S	MD	LM	L	PS
7	407	9	0	7	9	0	0	0	5
8	408	9	0	5	10	0	0	0	7
9	409	10	0	4	10	0	0	0	7
10	410	9	0	0	11	0	9	3	2
11	411	11	0	0	6	7	0	3	4
12	412	13	0	0	6	6	0	3	3
13	413	10	0	0	7	6	0	3	5
14	414	13	0	0	6	6	0	4	3
15	415	7	0	1	6	6	4	0	3
16	416	10	0	1	12	0	0	0	4
17	417	12	1	0	9	0	0	0	4
18	418	13	0	0	11	0	0	0	2
19	419	11	1	0	12	0	0	0	3

Dari tabel Tabel 6.13 diketahui bahwa jumlah libur berkisar antara 2 hingga 7 hari dalam satu bulan. Hasil libur yang dihasilkan cenderung tidak seimbang. Namun berdasarkan pembagian shift per staf pada Tabel 6.14, nilai JFI yang dihasilkan dengan menggunakan algoritma *Simulated Annealing* adalah 0.855224. Hasil ini lebih baik dibandingkan dengan nilai JFI jadwal manual maupun otomasi.

### 6.3.1.5 Hasil Optimasi Unit Rawat Jalan dan IGD

Optimasi akan menghasilkan jadwal baru berdasarkan nilai terbaik dari JFI yang didapatkan. Hasil penjadwalan dengan optimasi pada unit Rawat Jalan dan IGD dapat dilihat pada Gambar 6.17.

No	Kode Staf	J		Sb		M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		
1	501	M	M	LM	L	P	L	S	S	S	S	S	S	L	MD	MS	S	
2	502	MS	S	M	M	LM	P	P	P	S	S	S	S	S	L	MD		
3	503	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S	S	S	
4	504	S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S	S		
5	505	S	S	S	S	L	MD	MS	S	M	M	LM	L	P	P	S		
6	506	P	S	S	S	S	L	MD	L	S	M	M	LM	L	P			
7	507	L	P	P	S	S	S	S	S	L	MD	MS	S	M	M	LM		
8	508	M	LM	L	P	P	S	S	S	S	S	L	MD	MS	S	M		

No	Kode Staf	Sb		M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
		16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
1	501	M	M	LM	L	P	P	S	S	S	S	S	L	MD	L	S	M	
2	502	L	S	M	M	LM	L	P	P	S	S	S	S	S	L	MD	L	
3	503	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S	S	L	
4	504	S	S	L	MD	MS	S	M	M	L	S	P	P	S	S	S	S	
5	505	S	S	S	S	L	MD	L	S	M	M	LM	L	L	P	S	S	
6	506	P	S	S	S	S	S	S	MD	L	L	M	M	LM	L	P	P	
7	507	L	P	P	S	S	S	S	S	LM	MD	L	S	M	M	LM	L	
8	508	M	LM	L	P	P	S	L	S	S	S	L	MD	P	S	M	M	

Gambar 6.17 Hasil Jadwal Optimasi Unit Rawat Jalan dan IGD

Berdasarkan jadwal hasil optimasi, akan didapatkan informasi mengenai jumlah hari libur serta pembagian jumlah shift pada tiap staf.

Tabel 6.15 Hasil Perhitungan Libur Optimasi Unit Rawat Jalan dan IGD

No	Kode Staf	Jumlah Libur			Total
		Hari Kerja	Sabtu	Minggu	
1	501	6	0	0	6
2	502	3	1	1	5
3	503	2	2	1	5
4	504	1	0	3	4
5	505	6	0	0	6
6	506	4	1	1	6
7	507	2	2	1	5
8	508	4	0	1	5



**Tabel 6.16 Jumlah Shift Tiap Staf Hasil Optimasi Unit Rawat Jalan dan IGD**

No	Kode Staf	Tipe Shift						
		P	S	M	LM	MS	MD	L
1	501	3	12	5	2	1	2	6
2	502	5	12	4	2	1	2	5
3	503	4	12	4	2	2	2	5
4	504	4	14	4	1	2	2	4
5	505	3	13	4	2	1	2	6
6	506	5	12	4	2	0	2	6
7	507	4	12	4	3	1	2	5
8	508	5	11	5	2	1	2	5

Dari tabel Tabel 6.15 diketahui bahwa jumlah libur berkisar antara 4 hingga 6 hari dalam satu bulan. Hasil libur yang dihasilkan cenderung tidak seimbang. Namun berdasarkan pembagian shift per staf pada Tabel 6.16, nilai JFI yang dihasilkan dengan menggunakan algoritma *Simulated Annealing* adalah 0.944606. Hasil ini lebih baik dibandingkan dengan nilai JFI jadwal manual maupun otomatisasi.

### 6.3.1.6 Hasil Optimasi Unit Kamar Operasi

Optimasi akan menghasilkan jadwal baru berdasarkan nilai terbaik dari JFI yang didapatkan. Hasil penjadwalan dengan optimasi pada unit Kamar Operasi dapat dilihat pada Gambar 6.18.

No	Kode Staf	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	601	P	P	L	P	S	L	P	P	S	P	P	P	S	P	S
2	602	S	P	L	S	P	S	L	P	P	P	S	P	P	S	P
3	603	S	S	P	P	S	P	S	L	P	P	S	S	P	P	S
4	604	P	S	P	P	P	S	P	S	L	P	P	S	P	P	P
5	605	P	P	P	P	P	P	S	P	S	L	P	P	S	P	P
6	606	L	P	P	S	P	P	P	S	P	L	L	P	P	S	P

No	Kode Staf	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
		16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	601	L	P	P	S	S	P	P	S	L	S	L	P	P	S	S	P
2	602	S	P	P	P	S	S	P	P	L	P	S	L	P	P	S	P
3	603	P	L	L	P	L	S	S	P	P	S	P	S	L	P	P	P
4	604	S	L	S	L	P	P	S	P	P	P	S	P	S	L	P	P
5	605	P	P	P	S	P	P	P	S	P	P	P	S	P	S	L	L
6	606	P	P	S	P	S	L	P	P	P	P	P	P	S	P	S	L

**Gambar 6.18 Hasil Jadwal Optimasi Unit Kamar Operasi**

Berdasarkan jadwal hasil optimasi, akan didapatkan informasi mengenai jumlah hari libur serta pembagian jumlah shift pada tiap staf.

**Tabel 6.17 Hasil Perhitungan Libur Optimasi Unit Kamar Operasi**

No	Kode Staf	Jumlah Libur			
		Hari Kerja	Sabtu	Minggu	Total
1	601	2	1	2	5
2	602	2	0	2	4
3	603	4	0	1	5
4	604	2	1	1	4
5	605	0	1	2	3
6	606	3	0	2	5

**Tabel 6.18 Jumlah Shift Tiap Staf Hasil Optimasi Uit Kamar Operasi**

No	Kode Staf	Tipe Shift		
		P	S	L
1	601	16	10	5
2	602	17	10	4
3	603	15	11	5
4	604	18	9	4
5	605	21	7	3
6	606	19	7	5

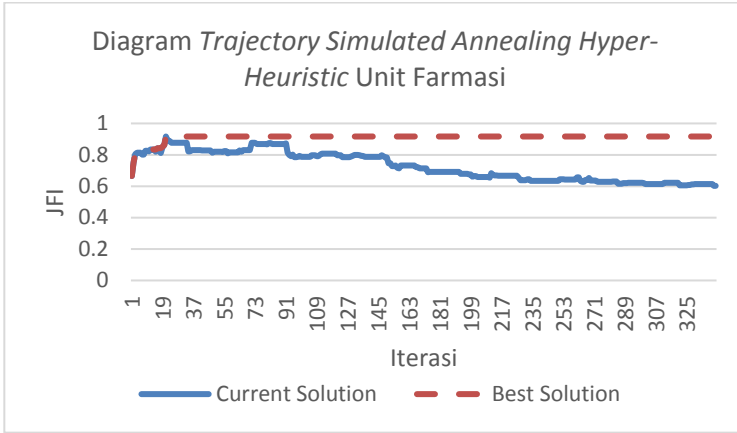
Dari tabel Tabel 6.17 diketahui bahwa jumlah libur berkisar antara 3 hingga 5 hari dalam satu bulan. Hasil libur yang dihasilkan cenderung tidak seimbang. Namun berdasarkan pembagian shift per staf pada Tabel 6.18, nilai JFI yang dihasilkan dengan menggunakan algoritma *Simulated Annealing* adalah 0.978359. Hasil ini lebih baik dibandingkan dengan nilai JFI jadwal manual maupun otomatisasi.

### 6.3.2 Diagram Trajectory

Diagram *trajectory* ini dibuat antara jumlah iterasi dengan nilai JFI yang berfungsi untuk melihat perbandingan antara nilai *current solution* dan *best solution* yang digunakan selama iterasi berlangsung.

#### 6.3.2.1 Diagram Trajectory Unit Farmasi

Diagram *trajectory* memetakan kinerja algoritma *Simulated Annealing Hyper-Heuristic* selama 342 iterasi pada Unit Farmasi.

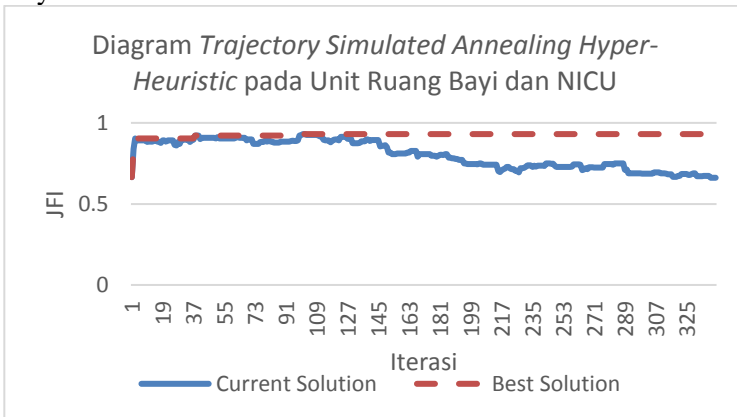


**Gambar 6.19** Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Farmasi

Berdasarkan hasil pengujian unit Farmasi pada Gambar 6.19 diketahui bahwa algoritma *Simulated Annealing Hyper-Heuristic* menemukan nilai terbaik ketika berada pada iterasi ke-21 dengan nilai *best solution* sebesar 0.9167.

### 6.3.2.2 Diagram Trajectory Unit Ruang Bayi dan NICU

Diagram trajectory memetakan kinerja algoritma *Simulated Annealing Hyper-Heuristic* selama 342 iterasi pada Unit Ruang Bayi dan NICU.

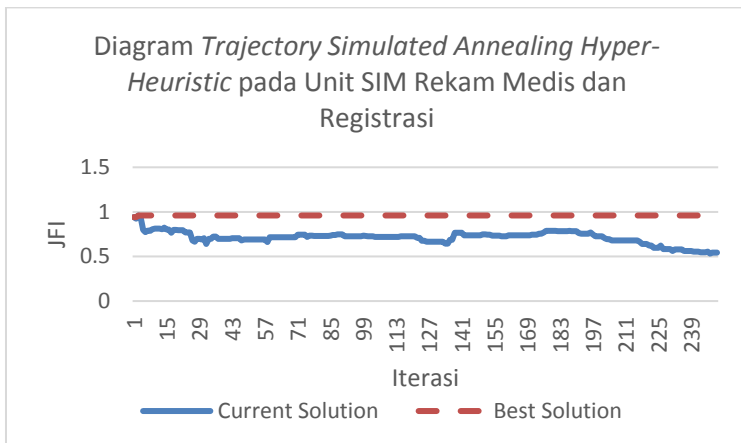


**Gambar 6.20 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Ruang Bayi dan NICU**

Berdasarkan hasil pengujian unit Ruang Bayi dan NICU pada Gambar 6.20, diketahui bahwa algoritma *Simulated Annealing Hyper-Heuristic* menemukan nilai terbaik ketika berada pada iterasi ke-101 dengan nilai *best solution* sebesar 0.930664.

**6.3.2.3 Diagram Trajectory Unit SIM Rekam Medis dan Registrasi**

Diagram trajectory memetakan kinerja algoritma *Simulated Annealing Hyper-Heuristic* selama 342 iterasi pada Unit SIM Rekam Medis dan Registrasi.

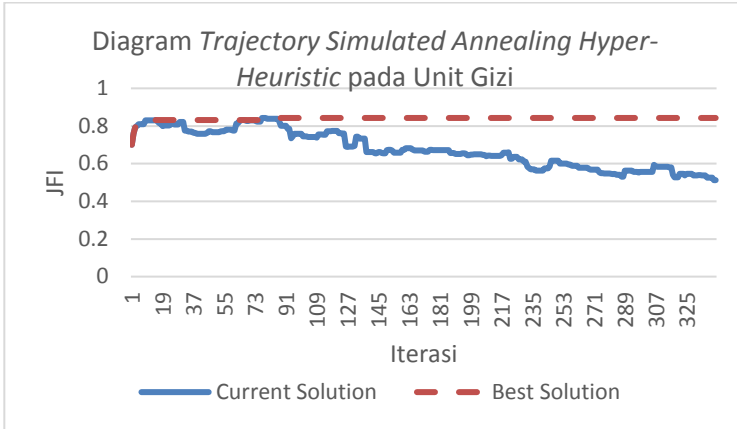


**Gambar 6.21 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit SIM Rekam Medis dan Registrasi**

Berdasarkan hasil pengujian unit SIM Rekam Medis dan Registrasi pada Gambar 6.21, diketahui bahwa algoritma *Simulated Annealing Hyper-Heuristic* menemukan nilai terbaik ketika berada pada iterasi ke-3 dengan nilai *best solution* sebesar 0.96.

**6.3.2.4 Diagram Trajectory Unit Gizi**

Diagram trajectory memetakan kinerja algoritma *Simulated Annealing Hyper-Heuristic* selama 342 iterasi pada Unit Gizi.

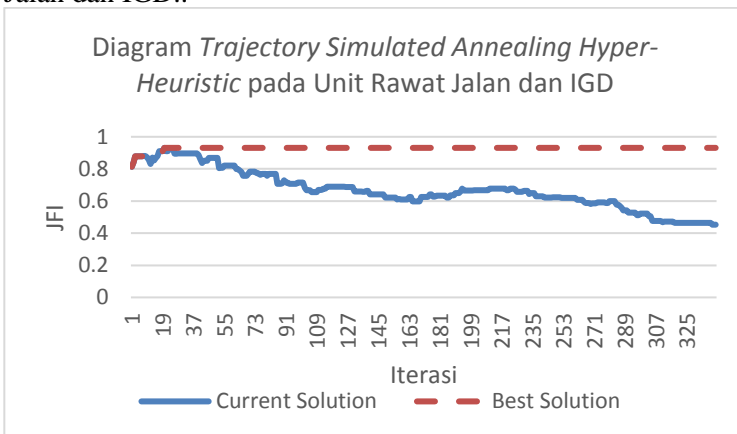


**Gambar 6.22** Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Gizi

Berdasarkan hasil pengujian unit Gizi pada Gambar 6.22, diketahui bahwa algoritma *Simulated Annealing Hyper-Heuristic* menemukan nilai terbaik ketika berada pada iterasi ke-77 dengan nilai *best solution* sebesar 0.842721.

### 6.3.2.5 Diagram Trajectory Unit Rawat Jalan dan IGD

Diagram trajectory memetakan kinerja algoritma *Simulated Annealing Hyper-Heuristic* selama 342 iterasi pada Unit Rawat Jalan dan IGD..

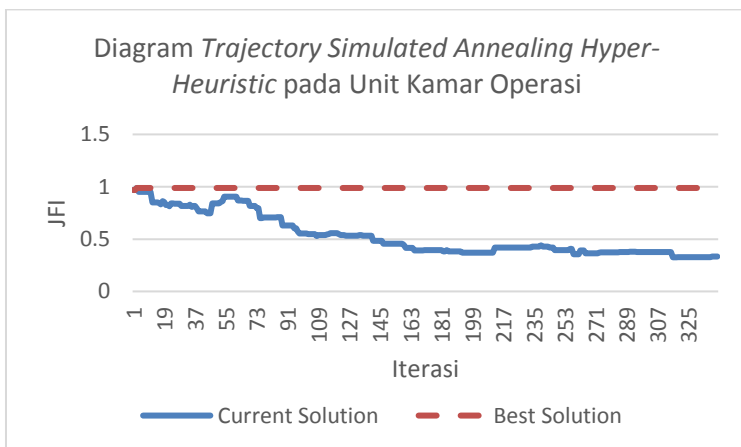


**Gambar 6.23 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Rawat Jalan dan IGD**

Berdasarkan hasil pengujian unit Rawat Jalan dan IGD pada Gambar 6.23, diketahui bahwa algoritma *Simulated Annealing Hyper-Heuristic* menemukan nilai terbaik ketika berada pada iterasi ke-20 dengan nilai *best solution* sebesar 0.931034.

**6.3.2.6 Diagram Trajectory Unit Kamar Operasi**

Diagram trajectory memetakan kinerja algoritma *Simulated Annealing Hyper-Heuristic* selama 342 iterasi pada Unit Kamar Operasi.



**Gambar 6.24 Diagram Trajectory Simulated Annealing Hyper-Heuristic Unit Kamar Operasi**

Berdasarkan hasil pengujian unit Kamar Operasi pada Gambar 6.24, diketahui bahwa algoritma *Simulated Annealing Hyper-Heuristic* menemukan nilai terbaik ketika berada pada iterasi ke-2 dengan nilai *best solution* sebesar 0.968559.

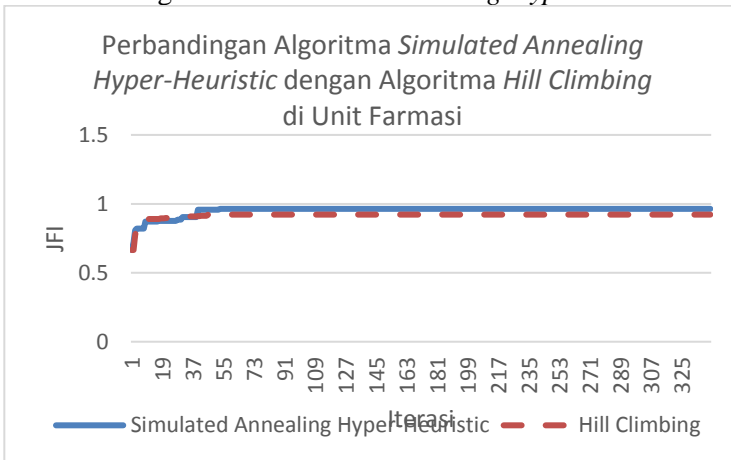
**6.2.2 Perbandingan Algoritma Simulated Annealing Hyper-Heuristic dengan Hill Climbing**

Berikut ini akan dibuat perbandingan antara penggunaan algoritma *Simulated Annealing Hyper-Heuristic* dengan

algoritma *Hill Climbing*. Kedua algoritma ini akan dijalankan dengan jumlah iterasi yang sama yaitu sebanyak 342 iterasi.

### 6.2.2.1 Perbandingan Algoritma Unit Farmasi

Dilakukan percobaan dengan menggunakan algoritma *Hill Climbing* pada Unit Farmasi dan hasilnya dibandingkan dengan hasil milik algoritma *Simulated Annealing Hyper-Heuristic*.



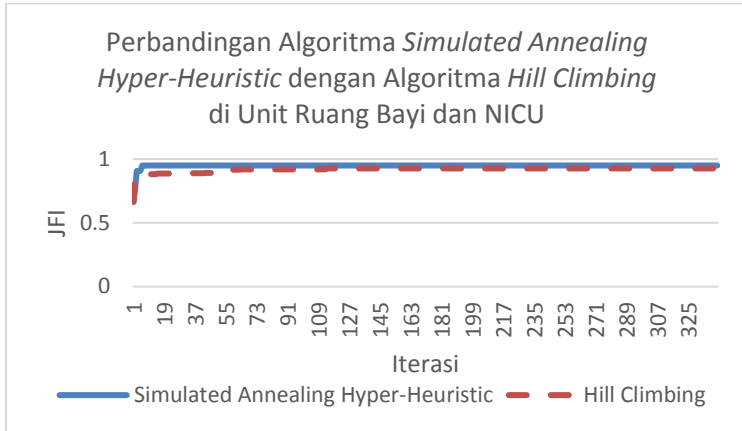
**Gambar 6.25 Perbandingan Algoritma pada Unit Farmasi**

Dari hasil percobaan unit Farmasi pada Gambar 6.25, ditemukan bahwa algoritma *Simulated Annealing Hyper-Heuristic* mencapai nilai terbaik pada iterasi ke-53 dengan nilai 0.963813. Sedangkan algoritma *Hill Climbing* mencapai nilai terbaik pada iterasi ke-46 dengan nilai 0.922.

### 6.2.2.2 Perbandingan Algoritma Unit Ruang Bayi dan NICU

Dilakukan percobaan dengan menggunakan algoritma *Hill Climbing* pada Unit Ruang Bayi dan NICU kemudian hasilnya dibandingkan dengan hasil milik algoritma *Simulated Annealing Hyper-Heuristic*.



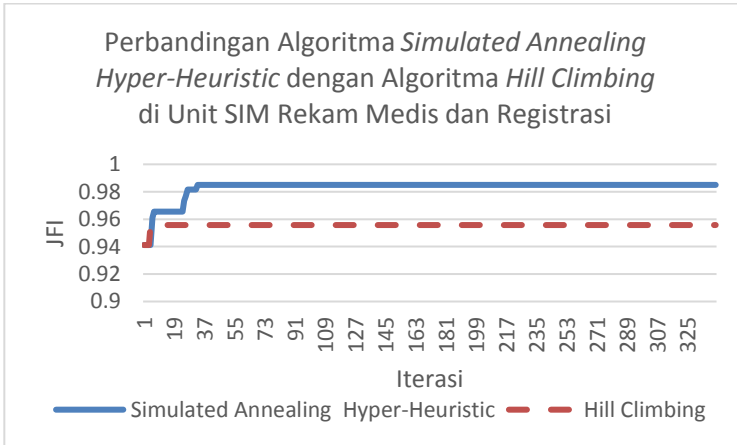


**Gambar 6.26** Perbandingan Algoritma pada Unit Ruang Bayi dan NICU

Dari hasil percobaan unit Ruang Bayi dan NICU pada Gambar 6.26, ditemukan bahwa algoritma *Simulated Annealing Hyper-Heuristic* mencapai nilai terbaik pada iterasi ke-11 dengan nilai 0.949635. Sedangkan algoritma *Hill Climbing* mencapai nilai terbaik pada iterasi ke-64 dengan nilai 0.920447.

### 6.2.2.3 Perbandingan Algoritma Unit SIM Rekam Medis dan Registrasi

Dilakukan percobaan dengan menggunakan algoritma *Hill Climbing* pada Unit SIM Rekam Medis dan Registrasi kemudian hasilnya dibandingkan dengan hasil milik algoritma *Simulated Annealing Hyper-Heuristic*.

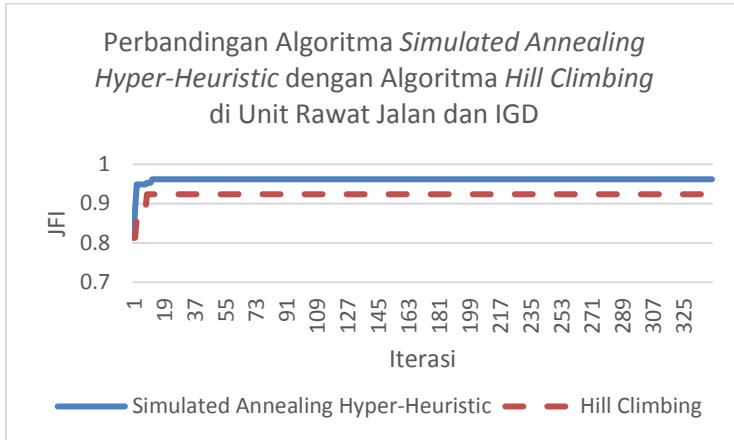


**Gambar 6.27** Perbandingan Algoritma pada Unit SIM Rekam Medis dan Registrasi

Dari hasil percobaan unit SIM Rekam Medis dan Registrasi pada Gambar 6.27, ditemukan bahwa algoritma *Simulated Annealing Hyper-Heuristic* mencapai nilai terbaik pada iterasi ke-33 dengan nilai 0.985003. Sedangkan algoritma *Hill Climbing* mencapai nilai terbaik pada iterasi ke-5 dengan nilai 0.955752.

#### 6.2.2.4 Perbandingan Algoritma Unit Rawat Jalan dan IGD

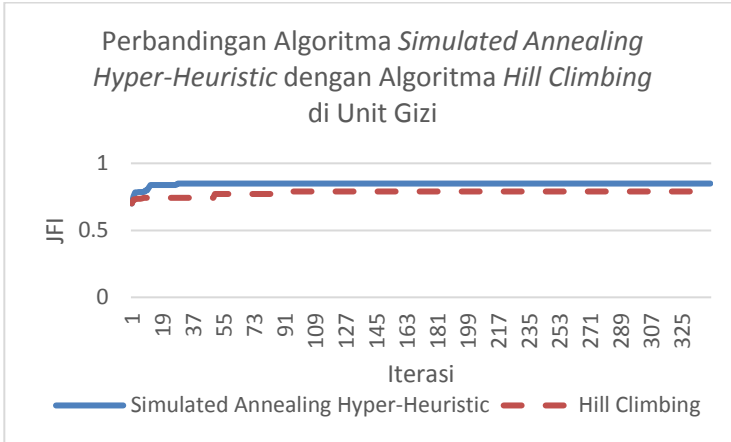
Dilakukan percobaan dengan menggunakan algoritma *Hill Climbing* pada Unit Rawat Jalan dan IGD kemudian hasilnya dibandingkan dengan hasil milik algoritma *Simulated Annealing Hyper-Heuristic*.



**Gambar 6.28 Perbandingan Algoritma pada Unit Rawat Jalan dan IGD**  
 Dari hasil percobaan unit Rawat Jalan dan IGD pada Gambar 6.28, ditemukan bahwa algoritma *Simulated Annealing Hyper-Heuristic* mencapai nilai terbaik pada iterasi ke-14 dengan nilai 0.962023. Sedangkan algoritma *Hill Climbing* mencapai nilai terbaik pada iterasi ke-11 dengan nilai 0.923883.

### 6.2.2.5 Perbandingan Algoritma Unit Gizi

Dilakukan percobaan dengan menggunakan algoritma *Hill Climbing* pada Unit Gizi kemudian hasilnya dibandingkan dengan hasil milik algoritma *Simulated Annealing Hyper-Heuristic*.

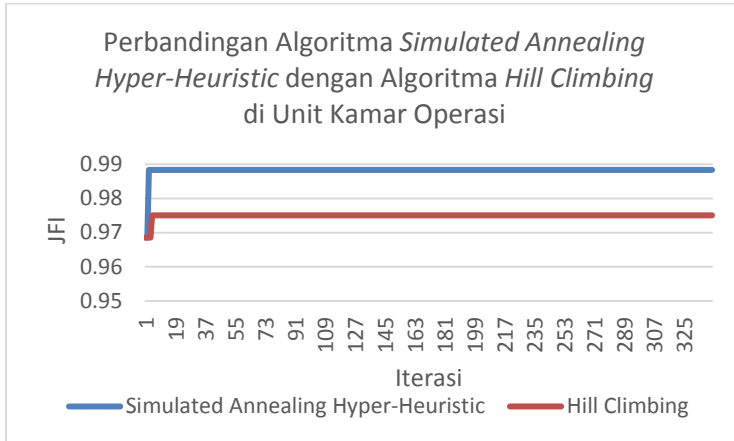


**Gambar 6.29 Perbandingan Algoritma Unit Gizi**

Dari hasil percobaan unit Gizi pada Gambar 6.29, ditemukan bahwa algoritma *Simulated Annealing Hyper-Heuristic* mencapai nilai terbaik pada iterasi ke-14 dengan nilai 0849011. Sedangkan algoritma *Hill Climbing* mencapai nilai terbaik pada iterasi ke-93 dengan nilai 0.790043.

### 6.2.2.6 Perbandingan Algoritma Unit Kamar Operasi

Dilakukan percobaan dengan menggunakan algoritma *Hill Climbing* pada Unit Kamar Operasi kemudian hasilnya dibandingkan dengan hasil milik algoritma *Simulated Annealing Hyper-Heuristic*.



**Gambar 6.30** Perbandingan Algoritma Unit Kamar Operasi

Dari hasil percobaan unit Kamar Operasi pada Gambar 6.30, ditemukan bahwa algoritma *Simulated Annealing Hyper-Heuristic* mencapai nilai terbaik pada iterasi ke-3 dengan nilai 0.988359. Sedangkan algoritma *Hill Climbing* mencapai nilai terbaik pada iterasi ke-6 dengan nilai 0.97507.

### 6.2.3 Perbandingan Statistik Algoritma *Simulated Annealing Hyper-Heuristic* dengan Algoritma *Hill Climbing*

Berikut ini adalah hasil perbandingan kinerja antara algoritma *Simulated Annealing Hyper-Heuristic* dan Algoritma *Hill Climbing* setelah dilakukan selama 20 kali percobaan untuk tiap algoritmanya dan diterapkan 342 kali iterasi untuk setiap percobaannya. Tabel-tabel di bawah ini adalah perhitungan nilai minimal, maksimal, dan rata-rata nilai *best solution* dari hasil 20 kali percobaan tersebut.

#### 6.2.3.1 Perbandingan Statistik Unit Farmasi

Dilakukan perbandingan statistik nilai JFI terbaik yang didapat pada Unit Farmasi.

Tabel 6.19 Statistik Perbandingan Algoritma Unit Farmasi

	<i>Simulated Annealing Hyper-Heuristic</i>	<i>Hill Climbing</i>
<b>MIN</b>	0.665158	0.665158
<b>MAX</b>	0.963813	0.960227
<b>AVERAGE</b>	0.8934	0.8804

Dari Tabel 6.19, diketahui bahwa nilai maksimal untuk JFI pada Unit Farmasi yang didapat oleh algoritma *Simulated Annealing Hyper-Heuristic* yaitu senilai 0.963813 lebih besar dibandingkan JFI maksimal milik *Hill Climbing* yaitu 0.960227. Sedangkan secara keseluruhan nilai rata-rata *best solution* paling baik dihasilkan oleh algoritma *Simulated Annealing Hyper-Heuristic* dengan nilai 0.8934 dibandingkan dengan algoritma *Hill Climbing* yang mendapat nilai 0.8804.

### 6.2.3.2 Perbandingan Statistik Unit Ruang Bayi dan NICU

Dilakukan perbandingan statistik nilai JFI terbaik yang didapat pada Unit Ruang Bayi dan NICU.

Tabel 6.20 Statistik Perbandingan Algoritma Unit Ruang Bayi dan NICU

	<i>Simulated Annealing Hyper-Heuristic</i>	<i>Hill Climbing</i>
<b>MIN</b>	0.664773	0.664773
<b>MAX</b>	0.960592	0.974227
<b>AVERAGE</b>	0.9183	0.9318

Dari Tabel 6.20, diketahui bahwa nilai maksimal untuk JFI pada Unit Ruang Bayi dan NICU yang didapat oleh algoritma *Hill Climbing* yaitu senilai 0.960592 lebih besar dibandingkan JFI maksimal milik *Simulated Annealing Hyper-Heuristic* yaitu 0.960227. Sedangkan secara keseluruhan nilai rata-rata *best solution* paling baik dihasilkan oleh algoritma *Hill Climbing*

dengan nilai 0.9318 dibandingkan dengan algoritma *Simulated Annealing Hyper-Heuristic* yang mendapat nilai 0.9183.

### 6.2.3.3 Perbandingan Statistik Unit SIM Rekam Medis dan Registrasi

Dilakukan perbandingan statistik nilai JFI terbaik yang didapat pada Unit SIM Rekam Medis dan Registrasi.

**Tabel 6.21** Statistik Perbandingan Algoritma Unit SIM Rekam Medis dan Registrasi

	<i>Simulated Annealing Hyper-Heuristic</i>	<i>Hill Climbing</i>
<b>MIN</b>	0.941176	0.941176
<b>MAX</b>	0.985003	0.979592
<b>AVERAGE</b>	0.9539	0.9487

Dari Tabel 6.21, diketahui bahwa nilai maksimal untuk JFI pada Unit SIM Rekam Medis dan Registrasi yang didapat oleh algoritma *Simulated Annealing Hyper-Heuristic* yaitu senilai 0.9850 lebih besar dibandingkan JFI maksimal milik *Hill Climbing* yaitu 0.9795. Sedangkan secara keseluruhan nilai rata-rata *best solution* paling baik dihasilkan oleh algoritma *Simulated Annealing Hyper-Heuristic* dengan nilai 0.9539 dibandingkan dengan algoritma *Hill Climbing* yang mendapat nilai 0.9487.

### 6.2.3.4 Perbandingan Statistik Unit Gizi

Dilakukan perbandingan statistik nilai JFI terbaik yang didapat pada Unit Gizi.

**Tabel 6.22** Statistik Perbandingan Algoritma Unit Gizi

	<i>Simulated Annealing Hyper-Heuristic</i>	<i>Hill Climbing</i>
<b>MIN</b>	0.70004	0.70004
<b>MAX</b>	0.91789	0.842447

<b>AVERAGE</b>	0.8049	0.7793
----------------	--------	--------

Dari Tabel 6.22, diketahui bahwa nilai maksimal untuk JFI pada Unit Gizi yang didapat oleh algoritma *Simulated Annealing Hyper-Heuristic* yaitu senilai 0.9178 lebih besar dibandingkan JFI maksimal milik *Hill Climbing* yaitu 0.8424. Sedangkan secara keseluruhan nilai rata-rata *best solution* paling baik dihasilkan oleh algoritma *Simulated Annealing Hyper-Heuristic* dengan nilai 0.8049 dibandingkan dengan algoritma *Hill Climbing* yang mendapat nilai 0.7793.

### 6.2.3.5 Perbandingan Statistik Unit Rawat Jalan dan IGD

Dilakukan perbandingan statistik nilai JFI terbaik yang didapat pada Unit Rawat Jalan dan IGD.

**Tabel 6.23 Statistik Perbandingan Algoritma Unit Rawat Jalan dan IGD**

	<i>Simulated Annealing Hyper-Heuristic</i>	<i>Hill Climbing</i>
<b>MIN</b>	0.813278	0.813278
<b>MAX</b>	0.972801	0.96716
<b>AVERAGE</b>	0.9334	0.9217

Dari Tabel 6.23, diketahui bahwa nilai maksimal untuk JFI pada Unit Rawat Jalan dan IGD yang didapat oleh algoritma *Simulated Annealing Hyper-Heuristic* yaitu senilai 0.97280 lebih besar dibandingkan JFI maksimal milik *Hill Climbing* yaitu 0.9671. Sedangkan secara keseluruhan nilai rata-rata *best solution* paling baik dihasilkan oleh algoritma *Simulated Annealing Hyper-Heuristic* dengan nilai 0.9334 dibandingkan dengan algoritma *Hill Climbing* yang mendapat nilai 0.9217.

### 6.2.3.6 Perbandingan Statistik Unit Kamar Operasi

Dilakukan perbandingan statistik nilai JFI terbaik yang didapat pada Unit Kamar Operasi.



**Tabel 6.24** Statistik Perbandingan Algoritma Unit Kamar Operasi

	<i>Simulated Annealing Hyper-Heuristic</i>	<i>Hill Climbing</i>
<b>MIN</b>	0.968559	0.968559
<b>MAX</b>	0.988359	0.988359
<b>AVERAGE</b>	0.9724	0.9741

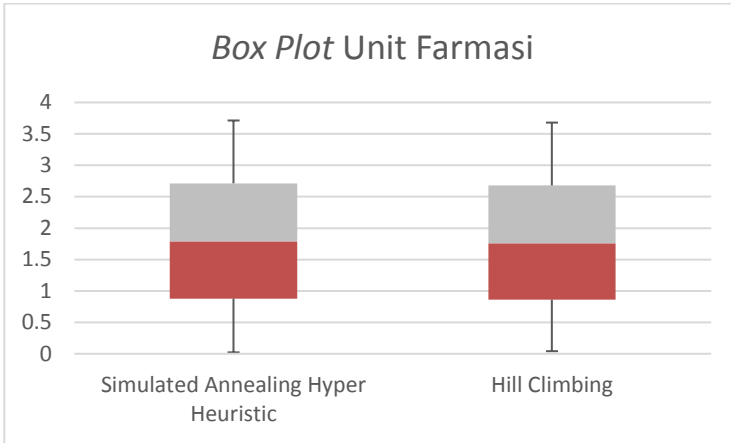
Dari Tabel 6.24, diketahui bahwa nilai maksimal untuk JFI pada Unit Rawat Jalan dan IGD yang didapat oleh algoritma *Simulated Annealing Hyper-Heuristic* yaitu senilai 0.988359 sama dengan JFI maksimal milik *Hill Climbing* yaitu 0.988359. Sedangkan secara keseluruhan nilai rata-rata *best solution* paling baik dihasilkan oleh algoritma *Hill Climbing* dengan nilai 0.9741 dibandingkan dengan algoritma *Simulated Annealing Hyper-Heuristic* yang mendapat nilai 0.9728.

#### **6.2.4 Pemetaan *Box Plot* Algoritma *Simulated Annealing Hyper-Heuristic* dengan Algoritma *Hill Climbing***

Setelah didapatkan hasil nilai minimal, maksimal, dan rata-rata untuk setiap percobaan pada masing-masing algoritma, dibuat diagram *Box Plot* yang bertujuan untuk mengetahui persebaran data.

##### **6.3.5.1 *Box Plot* Algoritma *Simulated Annealing Hyper-Heuristic* dan Algoritma *Hill Climbing* pada Unit Farmasi**

Pemetaan *Box Plot* pada Unit Farmasi dilakukan dengan menghitung nilai Q1, Q2, Q3, serta nilai maksimum dan minimum pada hasil percobaan selama 20 kali.

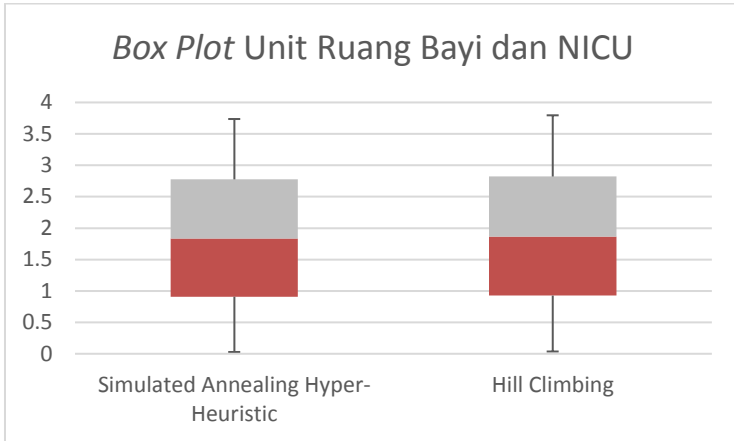


**Gambar 6.31** *Box Plot Unit Farmasi*

Berdasarkan Gambar 6.31 diketahui bahwa nilai Q1 pada Algoritma *Simulated Annealing Hyper-Heuristic* unit Farmasi adalah sebesar 0.87894. Nilai Q2-nya adalah 0.9089 sementara nilai Q3-nya adalah sebesar 0.92517. Pada Algoritma *Hill Climbing* diketahui bahwa nilai Q1 yang dimiliki adalah sebesar 0.86231, nilai Q2-nya adalah 0.89389 sementara nilai Q3-nya adalah sebesar 0.92319.

### **6.3.5.2** *Box Plot Algoritma Simulated Annealing Hyper-Heuristic dan Algoritma Hill Climbing pada Unit Ruang Bayi dan NICU*

Pemetaan Box Plot pada Unit Ruang Bayi dan NICU dilakukan dengan menghitung nilai Q1, Q2, Q3, serta nilai maksimum dan minimum pada hasil percobaan selama 20 kali.

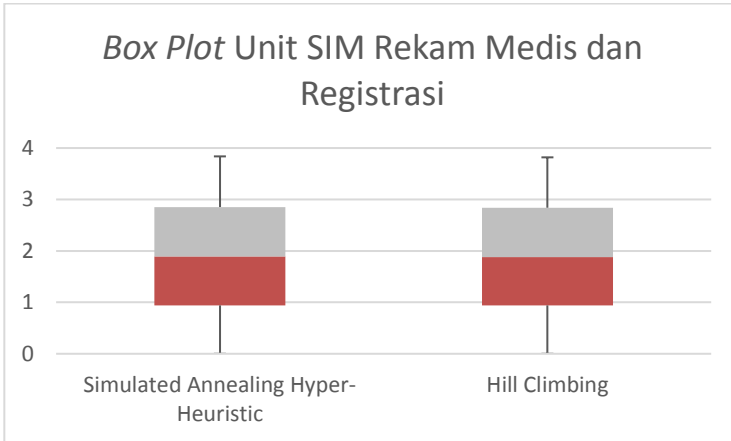


**Gambar 6.32** *Box Plot* Unit Ruang Bayi dan NICU

Berdasarkan Gambar 6.32 diketahui bahwa nilai Q1 pada Algoritma *Simulated Annealing Hyper-Heuristic* unit Ruang Bayi dan NICU adalah sebesar 0.907724. Nilai Q2-nya adalah 0.925551, sementara nilai Q3-nya adalah sebesar 0.942332. Pada Algoritma *Hill Climbing* diketahui bahwa nilai Q1 yang dimiliki adalah sebesar 0.926026, nilai Q2-nya adalah 0.941227 sementara nilai Q3-nya adalah sebesar 0.953769.

### 6.3.5.3 *Box Plot* Algoritma *Simulated Annealing Hyper-Heuristic* dan Algoritma *Hill Climbing* pada Unit SIM Rekam Medis dan Registrasi

Pemetaan *Box Plot* pada Unit SIM Rekam Medis dan Registrasi dilakukan dengan menghitung nilai Q1, Q2, Q3, serta nilai maksimum dan minimum pada hasil percobaan selama 20 kali.

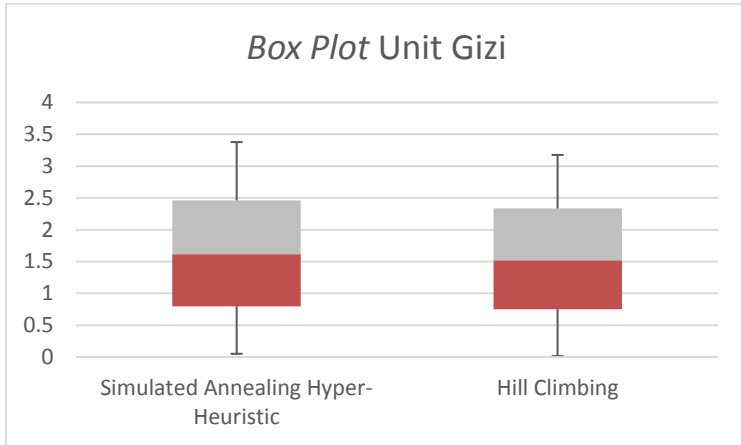


**Gambar 6.33** *Box Plot* Unit SIM Rekam Medis dan Registrasi

Berdasarkan Gambar 6.33 diketahui bahwa nilai Q1 pada Algoritma *Simulated Annealing Hyper-Heuristic* unit SIM Rekam Medis dan Registrasi adalah sebesar 0.87894. Nilai Q2-nya adalah 0.9089 sementara nilai Q3-nya adalah sebesar 0.92517. Pada Algoritma *Hill Climbing* diketahui bahwa nilai Q1 yang dimiliki adalah sebesar 0.86231, nilai Q2-nya adalah 0.89389 sementara nilai Q3-nya adalah sebesar 0.92319.

#### 6.3.5.4 *Box Plot* Algoritma *Simulated Annealing Hyper-Heuristic* dan Algoritma *Hill Climbing* pada Unit Gizi

Pemetaan *Box Plot* pada Unit Gizi dilakukan dengan menghitung nilai Q1, Q2, Q3, serta nilai maksimum dan minimum pada hasil percobaan selama 20 kali.

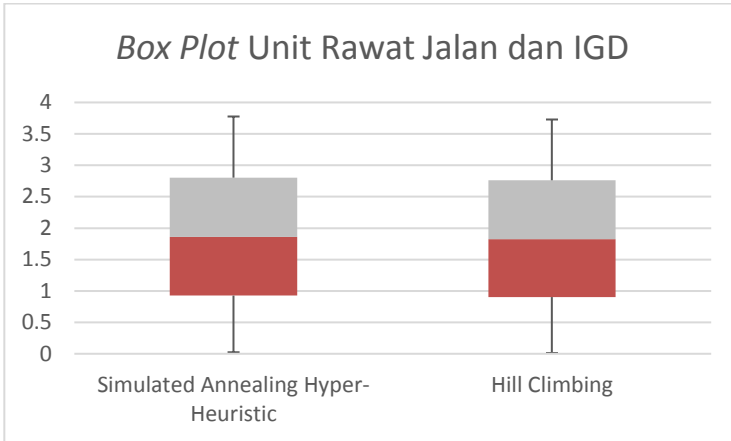


**Gambar 6.34** *Box Plot Unit Gizi*

Berdasarkan Gambar 6.34 diketahui bahwa nilai Q1 pada Algoritma *Simulated Annealing Hyper-Heuristic* unit Gizi adalah sebesar 0.7976. Nilai Q2-nya adalah 0.8184 sementara nilai Q3-nya adalah sebesar 0.843124. Pada Algoritma *Hill Climbing* diketahui bahwa nilai Q1 yang dimiliki adalah sebesar 0.7505, nilai Q2-nya adalah 0.7697 sementara nilai Q3-nya adalah sebesar 0.8123.

### 6.3.5.5 *Box Plot* Algoritma *Simulated Annealing Hyper-Heuristic* dan Algoritma *Hill Climbing* pada Unit Rawat Jalan dan IGD

Pemetaan *Box Plot* pada Unit Rawat Jalan dan IGD dilakukan dengan menghitung nilai Q1, Q2, Q3, serta nilai maksimum dan minimum pada hasil percobaan selama 20 kali.

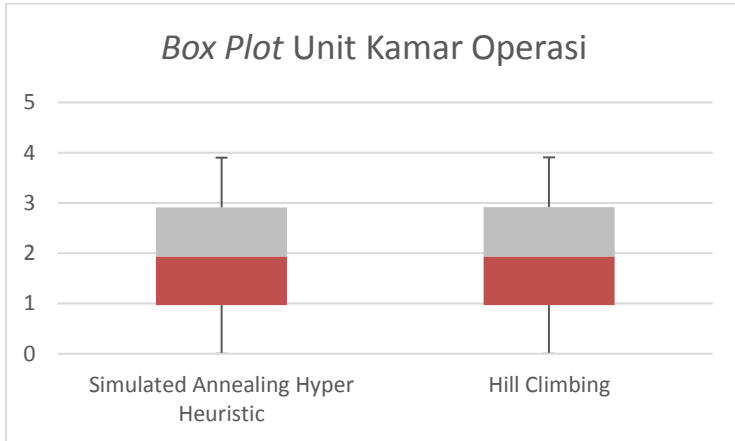


**Gambar 6.35** *Box Plot* Unit Rawat Jalan dan IGD

Berdasarkan Gambar 6.35 diketahui bahwa nilai Q1 pada Algoritma *Simulated Annealing Hyper-Heuristic* unit Rawat Jalan dan IGD adalah sebesar 0.9260. Nilai Q2-nya adalah 0.9332, sementara nilai Q3-nya adalah sebesar 0.9439. Pada Algoritma *Hill Climbing* diketahui bahwa nilai Q1 yang dimiliki adalah sebesar 0.9027, nilai Q2-nya adalah 0.9216 sementara nilai Q3-nya adalah sebesar 0.9369.

### 6.3.5.6 *Box Plot* Algoritma *Simulated Annealing Hyper-Heuristic* dan Algoritma *Hill Climbing* pada Unit Kamar Operasi

Pemetaan *Box Plot* pada Unit Kamar Operasi dilakukan dengan menghitung nilai Q1, Q2, Q3, serta nilai maksimum dan minimum pada hasil percobaan selama 20 kali.



**Gambar 6.36** Box Plot Unit Kamar Operasi

Berdasarkan Gambar 6.36 diketahui bahwa nilai Q1 pada Algoritma *Simulated Annealing Hyper-Heuristic* unit Kamar Operasi adalah sebesar 0.9685. Nilai Q2-nya adalah 0.9685, sementara nilai Q3-nya adalah sebesar 0.9758. Pada Algoritma *Hill Climbing* diketahui bahwa nilai Q1 yang dimiliki adalah sebesar 0.9685, nilai Q2-nya adalah 0.9685 sementara nilai Q3-nya adalah sebesar 0.9816.

### 6.3 Validasi Hasil Penjadwalan Otomatis dan Hasil Optimasi dengan Hard Constraints dari Masing-Masing Unit

Setelah dilakukan penjadwalan otomatis, langkah selanjutnya adalah melakukan validasi terhadap hard constraint yang dimiliki oleh masing-masing unit.

**Tabel 6.25** Validasi Hasil Otomasi dan Hasil Optimasi dengan Hard Constraint

No	Unit	Hard Constraint	Hasil Otomasi	Hasil Optimasi
1	Farmasi	Jumlah staf pada shift pagi tiap	Terpenuhi	Terpenuhi

No	Unit	Hard Constraint	Hasil Otomasi	Hasil Optimasi
		harinya harus sama dengan tiga		
		Jumlah staf pada shift siang tiap harinya harus sama dengan dua	Terpenuhi	Terpenuhi
		Jumlah staf pada shift malam tiap harinya harus sama dengan satu	Terpenuhi	Terpenuhi
		Jumlah staf pada shift pagi pada hari minggu harus sama dengan satu	Terpenuhi	Terpenuhi
		Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM-L) pada <i>shift</i> sebelumnya untuk anggota	Terpenuhi	Tidak Terpenuhi
2	NICU Ruang Bayi	Jumlah staf pada shift pagi setiap hari harus lebih dari sama dengan dua dan kurang	Terpenuhi	Terpenuhi



No	Unit	Hard Constraint	Hasil Otomasi	Hasil Optimasi
		dari sama dengan empat		
		Jumlah staf pada shift siang setiap hari harus lebih dari sama dengan satu dan kurang dari samadengan dua	Terpenuhi	Terpenuhi
		Jumlah staf pada shift malam setiap hari harus lebih dari sama dengan dua dan kurang dari sama dengan tiga	Terpenuhi	Terpenuhi
		Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM-L) pada <i>shift</i> sebelumnya untuk anggota	Terpenuhi	Tidak Terpenuhi
3	SIM & RM	Jumlah staf pada shift pagi tiap harinya lebih dari sama dengan satu	Terpenuhi	Terpenuhi

No	Unit	Hard Constraint	Hasil Otomasi	Hasil Optimasi
		atau kurang dari sama dengan dua		
		Jumlah staf pada shift siang tiap harinya lebih dari sama dengan satu atau kurang dari sama dengan dua	Terpenuhi	Terpenuhi
		Jumlah staf pada shift malam tiap harinya harus sama dengan satu	Terpenuhi	Terpenuhi
		Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM-L) pada <i>shift</i> sebelumnya.	Terpenuhi	Tidak Terpenuhi
4	Rawat Jalan dan IGD	Jumlah staf pada shift pagi tiap harinya lebih dari sama dengan satu atau kurang dari sama dengan dua	Terpenuhi	Terpenuhi

No	Unit	Hard Constraint	Hasil Otomasi	Hasil Optimasi
		Jumlah staf pada shift siang setiap hari harus lebih dari samadengan dua dan kurang dari samadengan empat	Terpenuhi	Terpenuhi
		Jumlah staf pada shift malam setiap hari harus berjumlah satu atau dua	Terpenuhi	Terpenuhi
		Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM-L) pada <i>shift</i> sebelumnya untuk anggota.	Terpenuhi	Tidak Terpenuhi
5	Gizi	Jumlah staf pada shift Ps untuk pada driver harus berjumlah dua pada hari Minggu	Terpenuhi	Terpenuhi
		Setiap hari harus memiliki satu shift malam untuk	Terpenuhi	Terpenuhi

No	Unit	Hard Constraint	Hasil Otomasi	Hasil Optimasi
		staf yang berskil sebagai penyaji		
		Untuk mendapatkan hari Libur, maka dibutuhkan pola Malam-Malam-Lepas Malam (M-M-LM-L) pada <i>shift</i> sebelumnya untuk penyaji	Terpenuhi	Tidak Terpenuhi
6	Kamar Operasi	Jumlah staf pada shift pagi setiap hari harus lebih dari samadengan dua dan kurang dari sama dengan empat	Terpenuhi	Terpenuhi
		Jumlah staf pada shift siang setiap hari harus lebih dari samadengan dua atau kurang dari samadengan tiga	Terpenuhi	Terpenuhi
		Jumlah staf pada shift pagi pada hari Minggu harus terdiri dari empat	Terpenuhi	Terpenuhi

#### 6.4 Perbandingan Ketidaksesuaian Jadwal Manual, Otomasi dengan Optimasi

Hasil pada penjadwalan optimasi memenuhi sebagian besar *hard constraint* namun tidak untuk pemenuhan pola M-M-LM-L. Perhitungan terkait persentase ketidaksesuaian jadwal dengan *hard constraint* dapat dilihat pada Tabel 6.26. Perhitungan pelanggaran dilakukan setiap kali ada pola M-M-LM-M yang terlanggar.

Tabel 6.26 Perbandingan Persentase Pelanggaran Jadwal Manual, Otomasi dan Optimasi Terhadap Pola M-M-LM-L

Unit	Manual			Otomasi			Optimasi		
	Jumlah Pola M-M-LM-L	Jumlah Pelanggaran	%	Jumlah Pola M-M-LM-L	Jumlah Pelanggaran	%	Jumlah Pola M-M-LM-L	Jumlah Pelanggaran	%
Farmasi	13	0	0%	14	0	0%	14	10	71.42%
Ruang Bayi dan NICU	41	3	7.31%	34	0	0%	34	5	14.70%
SIM Rekam Medis	14	3	21.42%	14	0	0%	14	3	21.42%

Unit	Manual			Otomasi			Optimasi		
	Jumlah Pola M-M-LM-L	Jumlah Pelanggaran	%	Jumlah Pola M-M-LM-L	Jumlah Pelanggaran	%	Jumlah Pola M-M-LM-L	Jumlah Pelanggaran	%
dan Registrasi									
Gizi	14	0	0%	14	0	0%	14	1	7.14%
Rawat Jalan dan IGD	13	5	38.46%	15	0	0%	15	2	13.33%

Pola M-M-LM-L yang terlanggar ini dihitung pada pola yang terdapat pada tipe skill anggota untuk Unit Farmasi (lihat **Tabel 4.2**), Unit Ruang Bayi dan NICU (lihat **Tabel 4.3**), Unit SIM Rekam Medis dan Registrasi (lihat **Tabel 4.4**), dan Unit Rawat Jalan dan IGD (lihat **Tabel 4.6**). Untuk Unit Gizi, pola ini berlaku untuk tipe skill penyaji (lihat **Tabel 4.5**).

Berdasarkan hasil perhitungan pelanggaran terhadap pola M-M-LM-L, jumlah pelanggaran terbanyak dari hasil optimasi terdapat pada Unit Farmasi dengan persentase pelanggaran sebesar 71.42%. Sedangkan unit dengan jumlah pelanggaran paling sedikit adalah unit Gizi dengan persentase pelanggaran sebesar 7.14%. Unit Kamar Operasi tidak memiliki pola M-M-LM-L ataupun pola yang harus terpenuhi. Maka dari itu untuk Kamar Operasi tidak ada pelanggaran terhadap *hard constraint*.

## 6.6 Perbandingan Nilai JFI Manual, Otomasi dengan Optimasi Jadwal Existing

Bagian ini akan menjabarkan perbandingan antara nilai *Jain's fairness Index* yang didapatkan dari hasil manual, otomasi serta optimasi dengan menggunakan algoritma *Simulated Annealing Hyper-Heuristic*.

### 6.6.1 Unit Farmasi

Tabel 6.27 Perbandingan JFI Unit Farmasi

JFI Manual November 2017	JFI Otomasi Desember 2017	JFI Optimasi Desember 2017
0.9236	0.6651	0.9349

Berdasarkan Tabel 6.27 diketahui bahwa nilai JFI pada unit Farmasi dengan cara manual senilai 0.9236 lebih baik dibandingkan dengan otomasi yang senilai 0.6651. Namun dengan menggunakan optimasi, nilai JFI mengalami kenaikan hingga mencapai nilai 0.9349.

### 6.6.2 Unit Ruang Bayi dan NICU

Tabel 6.28 Perbandingan JFI Ruang Bayi dan NICU

JFI Manual November 2017	JFI Otomasi Desember 2017	JFI Optimasi Desember 2017
0.8922	0.6647	0.9154

Berdasarkan Tabel 6.28 diketahui bahwa nilai JFI pada unit Ruang Bayi dan NICU dengan cara manual senilai 0.8992 lebih baik dibandingkan dengan otomasi yang senilai 0.6647. Namun dengan menggunakan optimasi, nilai JFI mengalami kenaikan hingga mencapai nilai 0.9154.

### 6.6.3 Unit SIM Rekam Medis dan Registrasi

Tabel 6.29 Perbandingan JFI SIM Rekam Medis dan Registrasi

JFI Manual November 2017	JFI Otomasi Desember 2017	JFI Optimasi Desember 2017
0.9009	0.9411	0.9795

Berdasarkan Tabel 6.29 diketahui bahwa nilai JFI pada unit SIM Rekam Medis dan Registrasi dengan otomasi senilai 0.9411 lebih baik dibandingkan dengan cara manual yang senilai 0.9009. Namun dengan menggunakan optimasi, nilai JFI mengalami kenaikan hingga mencapai nilai 0.9795.

#### 6.6.4 Unit Gizi

**Tabel 6.30 Perbandingan JFI Unit Gizi**

<b>JFI Manual November 2017</b>	<b>JFI Otomasi Desember 2017</b>	<b>JFI Optimasi Desember 2017</b>
0.8516	0.7004	0.8577

Berdasarkan Tabel 6.30 diketahui bahwa nilai JFI pada unit Gizi dengan manual senilai 0.8516 lebih baik dibandingkan dengan cara otomasi yang senilai 0.9009. Namun dengan menggunakan optimasi, nilai JFI mengalami kenaikan hingga mencapai nilai 0.8577.

#### 6.6.5 Unit Rawat Jalan dan IGD

**Tabel 6.31 Perbandingan JFI Unit Rawat Jalan dan IGD**

<b>JFI Manual November 2017</b>	<b>JFI Otomasi Desember 2017</b>	<b>JFI Optimasi Desember 2017</b>
0.9184	0.8123	0.9337

Berdasarkan Tabel 6.31 diketahui bahwa nilai JFI pada unit Rawat Jalan dan IGD dengan cara manual senilai 0.9184 lebih baik dibandingkan dengan hasil otomasi yang senilai 0.8123. Namun dengan menggunakan optimasi, nilai JFI mengalami kenaikan hingga mencapai nilai 0.9337.

#### 6.6.6 Unit Kamar Operasi

**Tabel 6.32 Perbandingan JFI Unit Kamar Operasi**

<b>JFI Manual November 2017</b>	<b>JFI Otomasi Desember 2017</b>	<b>JFI Optimasi Desember 2017</b>
0.8069	0.9685	0.9750

Berdasarkan Tabel 6.32 diketahui bahwa nilai JFI pada unit Kamar Operasi dengan otomasi senilai 0.9685 lebih baik



dibandingkan dengan cara manual yang senilai 0.8069. Namun dengan menggunakan optimasi, nilai JFI mengalami kenaikan hingga mencapai nilai 0.9750.

*(Halaman ini sengaja dikosongkan)*

## **BAB VII**

### **KESIMPULAN DAN SARAN**

Bab ini berisikan kesimpulan dari hasil penelitian dan juga saran perbaikan untuk penelitian kedepannya.

#### **7.1 Kesimpulan**

Berdasarkan uji coba yang telah dilakukan dalam tugas akhir ini, maka kesimpulan yang dapat diambil adalah :

1. Pembuatan jadwal yang optimal dapat dilakukan dengan membandingkan nilai keadilan atau *Jain's Fairness Index* antara staf rumah sakit yang di hasilkan oleh penjadwalan manual dan penjadwalan otomatis yang telah di optimasi menggunakan metode *Hyper-Heuristic Simulated Annealing*.
2. Hasil optimasi sebagian besar telah memenuhi *hard constraint* yang telah dijelaskan kecuali untuk *hard constraint* pemenuhan pola M-M-LM-L masih belum dapat terpenuhi.
3. Hasil optimasi juga masih melanggar salah satu *soft constraint* yaitu dalam hal pemenuhan jumlah libur tiap bulannya. Tidak semua karyawan mendapatkan jumlah libur yang seimbang meskipun nilai JFI hasil optimaasi cukup tinggi.
4. Nilai JFI hasil optimasi untuk seluruh unit yang didapat, bernilai lebih besar dibandingkan dengan nilai JFI hasil penjadwalan manual maupun nilai JFI hasil otomasi.
5. Nilai JFI unit Farmasi bertambah dari 0.92 menjadi 0.93 setelah dilakukan optimasi.
6. Nilai JFI unit Ruang Bayi dan NICU bertambah dari 0.89 menjadi 0.91 setelah dilakukan optimasi.
7. Nilai JFI unit SIM Rekam Medis dan Registrasi meningkat dari 0.90 menjadi 0.97 setelah dilakukan optimasi.
8. Nilai JFI unit Gizi meningkat dari 0.85 menjadi 0.86 setelah dilakukan optimasi.
9. Nilai JFI unit Rawat Jalan dan IGD meningkat dari 0.91 menjadi 0.93 setelah dilakukan optimasi.

10. Nilai JFI Unit Kamar Operasi meningkat dari 0.80 menjadi 0.97 setelah dilakukan optimasi.
11. Hasil statistik perbandingan dari implementasi algoritma *Simulated Annealing Hyper-Heuristic* dan algoritma *Hill Climbing* memiliki beberapa perbedaan. Pada unit Farmasi, Rawat Jalan dan IGD, SIM Rekam Medis dan Registrasi, dan Gizi nilai rata-rata hasil optimasi dengan algoritma *Simulated Annealing Hyper-Heuristic* lebih besar dibandingkan dengan *Hill Climbing*. Namun pada unit Ruang Bayi dan NICU dan Kamar Operasi hasil optimasi dengan algoritma *Simulated Annealing Hyper-Heuristic* lebih rendah dibandingkan dengan *Hill Climbing*.
12. Nilai rata-rata hasil optimasi Unit Farmasi, paling tinggi didapatkan oleh algoritma *Simulated Annealing Hyper-Heuristic* dengan nilai rata-rata nilai JFI yang dihasilkan sebesar 0.8934.
13. Nilai rata-rata hasil optimasi Unit Ruang Bayi dan NICU, paling tinggi didapatkan oleh algoritma *Hill Climbing* dengan nilai rata-rata nilai JFI yang dihasilkan sebesar 0.9318.
14. Nilai rata-rata hasil optimasi Unit SIM Rekam Medis dan Registrasi, paling tinggi didapatkan oleh algoritma *Simulated Annealing Hyper-Heuristic* dengan nilai rata-rata nilai JFI yang dihasilkan sebesar 0.9539.
15. Nilai rata-rata hasil optimasi Unit Gizi, paling tinggi didapatkan oleh algoritma *Simulated Annealing Hyper-Heuristic* dengan nilai rata-rata nilai JFI yang dihasilkan sebesar 0.8049.
16. Nilai rata-rata hasil optimasi Unit Rawat Jalan dan IGD, paling tinggi didapatkan oleh algoritma *Simulated Annealing Hyper-Heuristic* dengan nilai rata-rata nilai JFI yang dihasilkan sebesar 0.9334.
17. Nilai rata-rata hasil optimasi Unit Kamar Operasi, paling tinggi didapatkan oleh algoritma *Hill Climbing* dengan nilai rata-rata nilai JFI yang dihasilkan sebesar 0.974.

## 7.2 Saran

Saran yang dapat diberikan berdasarkan proses uji coba, penarikan kesimpulan, dan batasan masalah dari tugas akhir ini adalah :

1. Perlu adanya perubahan fungsi tujuan dari perhitungan nilai *Jain's Fairness Index* seperti berdasarkan jumlah libur per bulannya.
2. ada penelitian ini *low-level heuristic* yang digunakan hanya ada 2. Maka dari itu penelitian selanjutnya dapat menggunakan *low-level heuristic* dengan jumlah yang lebih banyak.
3. Pada penelitian ini, terdapat beberapa batasan yang seharusnya menjadi *hard constraint* seperti pola pada Kepala Unit yang hanya ada *shift* pagi dan libur. Selain itu jumlah libur wajib setiap bulannya bisa ditambahkan menjadi *hard constraint*.
4. Pada penelitian ini, eksperimen yang dilakukan untuk algoritma *Simulated Annealing Hyper-Heuristic* hanya menggunakan parameter yang tetap dan tidak berubah. Penelitian selanjutnya dapat dilakukan pengujian berbagai parameter sehingga didapatkan hasil yang lebih maksimal.
5. Perlu ditambahkan adanya preferensi staf rumah sakit yang lebih detail demi mendukung penentuan kriteria *Jain's Fairness Index*.
6. Penelitian ini hanya menggunakan algoritma *Simulated Annealing Hyper-Heuristic*. Penelitian selanjutnya dapat mencoba menggunakan metode lain seperti *Genetic Algorithm* lalu dilakukan perbandingan untuk menemukan metode terbaik dalam penjadwalan staff rumah sakit.
7. Penjadwalan staff rumah sakit ini dapat di kembangkan menjadi aplikasi menggunakan Bahasa pemrograman yang lain seperti PHP atau Python sesuai dengan kebutuhan rumah sakit.



## DAFTAR PUSTAKA

- [1] RSIA Kendangsari, “Tentang Kami - RSIA Kendangsari Surabaya,” [Online]. Available: <http://rsia.kendangsari.com/tentang-kami/>. [Diakses 25 September 2017].
- [2] B. Hayes, B. A dan P. J, “Factors Contributing to Nurse Job Satisfaction in the Acute Hospital Setting : A review of recent literature,” *Journal of Nursing Management*, pp. 804-814, 2010.
- [3] E. K. Burke, P. D. Causmaecker, G. V. Berghe dan H. V. Landeghem, “The State of The Art of Nurse Rostering,” *Journal of Scheduling*, vol. 7, pp. 441-499, 2004.
- [4] A. P. Hidayatullah dan B. Santosa, “Optimasi Nurse Scheduling Problem (Studi Kasus RSUD Dr. Soetomo Surabaya),” *Jurnal Teknik*, 2014.
- [5] R. Bai, J. Blazewicz, E. Burke dan B. McCollum, “A Simulated Annealing Hyper-Heuristic Methodology for Flexible Decision Support,” *School of CSiT, University of Nottingham, UK, Tech. Rep.*, 2007.
- [6] RSIA Kendangsari, “Fasilitas | RSIA Kendangsari - Surabaya,” [Online]. Available: <http://rsia.kendangsari.com/fasilitas/>. [Diakses 27 September 2017].
- [7] RSIA Kendangsari, “Layanan Unggulan | RSIA Kendangsari - Surabaya,” [Online]. Available: <http://rsia.kendangsari.com/layanan-unggulan/>. [Diakses 28 September 2017].
- [8] M. L. Pinedo, *Scheduling - Theory, Algorithms, and Systems (Third Edition)*, Prentice Hall, 2008.
- [9] P. Smet, “Nurse rostering: models and algorithms for theory, practice and integration with other problems,” 2015.

- [10] I. Solos, I. Tassopoulos dan G. Beligiannis, "A Generic Two-Phase Stochastic Variable Neighborhood Approach for Effectively Solving the Nurse Rostering Problem," *Algorithms*, 2013.
- [11] D. Simic, S. M. D. Simic dan J. Djordjevic, "Challenges For Nurse Rostering Problem and Opportunities in Hospital Logistics," *Journal of Medical Informatics and Technologies*, vol. 23, 2014.
- [12] S. Asta, E. Ozcan dan T. Curtois, "A Tensor Based Hyper-Heuristic for Nurse Rostering," *Knowledge-Based System*, pp. 185-199, 2015.
- [13] F. Buseti, "Simulated Annealing overview," 2003. [Online]. Available: [www.geocities.com/francorbusetti/saweb.pdf](http://www.geocities.com/francorbusetti/saweb.pdf). [Diakses 28 September 2017].
- [14] R. A. Rutenbar, "Simulated Annealing Algorithms : An Overview," *IEEE Circuits and Devices Magazine*, vol. 5, pp. 19-26, 1989.
- [15] K. E. Geltman, "The Simulated Annealing Algorithm," 20 February 2014. [Online]. Available: <http://katrinaeg.com/simulated-annealing.html>. [Diakses 4 October 2017].
- [16] A. Muklason, "Hyper-heuristics and Fairness in Examination Timetabling Problems," 2017.
- [17] E. K. Burke, M. Gendreau, M. Hyde, G. Kendall, G. Ochoa, E. Ozcan dan R. Qu, "Hyper-heuristics: A Survey Of The State Of The Art," *Journal of the Operational Research Society*, pp. 1695-1724, 2013.
- [18] E. K. Burke, M. Hyde, G. Kendall, G. Ochoa, E. Ozcan dan J. R. Woodward, "A Classification of Hyper-heuristic Approaches," *Handbook of Metaheuristics*, pp. 449-468, 2010.
- [19] R. K. Jain, D.-M. W. Chiu dan W. R. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System,"



Hudson, MA: Eastern Research Laboratory, *Digital Equipment Library*, vol. 38, 1984.

- [20] A. Muklason, A. J. Parkes, B. McCollum dan O. Ender, “Initial Results on Fairness in Examination Timetabling,” *6th Multidisciplinary International Conference on Scheduling : Theory and Applications*, 2013.
- [21] J. H. Dinitz dan D. Stinson, “A Hill-Climbing Algorithm For The Construction Of One-Factorizations and Room Squares,” *Society for Industrial and Applied Mathematics*, vol. 8, no. 3, pp. 430-438, 1987.
- [22] R. Bai dan G. Kendall, “An Investigation of Automated Planograms Using A Simulated Annealing Based Hyper-Heuristics,” *Metaheuristics: Progress as Real Problem Solvers*, pp. 87-108, 2005.
- [23] A. Lombardi, “Estimation of The Parameters of ETAS Models by Simulated Annealing | Scientific Reports,” 12 February 2015. [Online]. Available: <https://www.nature.com/articles/srep08417>. [Diakses 6 1 2018].
- [24] A. K. Pephrah, S. K. Appiah dan S. K. Amponsah, “An Optimal Cooling Schedule Using a Simulated Annealing Based Approach,” *Applied Mathematics*, vol. 8, pp. 1195-1210, 2017.
- [25] P. J. van Laarhoven dan E. H. Aarts, “Simulated Annealing: Theory and Applications,” *Mathematics and Its Applications*, vol. 37, 1987.

## BIODATA PENULIS



Penulis lahir di Surabaya pada tanggal 9 Oktober 1994. Penulis merupakan anak ketiga dari tiga bersaudara. Penulis telah menempuh pendidikan formal di sekolah negeri mulai dari SD Pemuda Bangsa, SMP Negeri 3 Depok, dan SMAN 1 Depok. Setelah lulus, penulis melanjutkan ke jenjang perguruan tinggi negeri di Surabaya, yakni Jurusan Sistem Informasi Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2014. Sebagai mahasiswa, penulis aktif dalam urusan organisasi. Tercatat penulis aktif berkontribusi melalui keanggotaan organisasi mahasiswa di BEM Fakultas Teknologi Informasi pada 2015 sebagai anggota dan tahun 2016 sebagai satf ahli. Penulis juga pernah melakukan kerja praktik di Bank Indonesia Pusat Jakarta pada bulan Juli hingga Agustus tahun 2017. Untuk mendapatkan gelar Sarjana Komputer (S.Kom), penulis mengambil laboratorium Rekayasa Data dan Intelejensi Bisnis dengan topik tugas akhir Optimasi menggunakan Simulated Annealing Hyper-Heuristic pada penjadwalan rumah sakit. Untuk kepentingan penelitian penulis juga dapat dihubungi melalui e-mail: [indriartikusumanita@gmail.com](mailto:indriartikusumanita@gmail.com).

## LAMPIRAN A – HASIL WAWANCARA

### Interview Protocol

#### Informasi interview

Interviewer : Rizka Pordella, Indriarti Kusumanita,  
Fachrur Zaffrinda, Fata Hirzi, & Zuli Maulidati  
Narasumber : Bu silvy  
Hari, Tanggal : Senin, 28 Oktober 2017  
Pukul : 10.30-12.30  
Lokasi : RSIA Merr Kendangsari

#### Informasi narasumber

Nama : Bu Sylvy Medtasya Dzykrzyanka, S. Farm,  
M. Farm. Klin, APT, MARS  
Jabatan : Kepala Bagian  
Divisi : Sumber Daya Manusia (SDM) & Hukum  
Instansi : RSIA Merr Kendangsari  
Lama Bekerja : 3.5 tahun

#### Penjelasan interview

1. Interview ini bertujuan untuk salah satu sumber data untuk tugas akhir dengan judul “Optimasi Penjadwalan Perawat Dengan Menggunakan Algoritma Tabu Search, Late acceptance, Reinforcement Learning, Self adaptive learning, simulated annealing Based Hyper-Heuristics (Studi Kasus: Rumah Sakit Ibu Dan Anak Kendangsari)” yang dimaksudkan agar peneliti bisa mendapatkan gambaran mengenai permasalahan penjadwalan perawat yang ada di RSIA Kendangsari Surabaya.
2. Dengan melakukan interview ini diharapkan peneliti mendapatkan informasi mengenai penjadwalan perawat yang sudah diterapkan Rumah Sakit Ibu dan Anak kendangsari Surabaya

3. Untuk menjaga dan menjamin kerahasiaan, maka data – data yang bersifat pribadi akan dirahasiakan oleh peneliti.

Berikut adalah daftar pertanyaan untuk Senior Manager sebagai penanggungjawab proses bisnis yang dijalankan pada RSIA kendangsari Surabaya:

1. Berapa jumlah Staff yang dimiliki oleh RS?  
Answer: terdapat 124 Staff (Keseluruhan medis dan non medis)
2. Ada berapakan Tipe staff perawat dipekerjakan? (Skill Type)?  
Answer: klasifikasi Perawat dilakukan berdasarkan Pelatihan keterampilan yang dimiliki masing masing perawat. Secara keseluruhan perawat yang dimiliki oleh RSIA hanya memiliki sampai PK2 (Pelatihan Keterampilan tingkat 2) sedangkan untuk perawat seior atau kepalaperawat memiliki PK3 (Pelatihan Keterampilan tingkat 3)
3. Berapa jumlah ward atau bangsal yang dimiliki oleh RS?  
Answer : RSIA memiliki 14 unit atau bangsal, terdapat 6 unit RSIA yang memiliki pattern yang unik dalam menjadwalkan staff baik medis maupun non medis yaitu:
  - Kamar Operasi
  - Ruang bayi dan NICU
  - SIM dan RM
  - Instalasi Farmasi
  - Instalasi rawat jalan dan IGD
  - Instalasi Gizi
4. Bagaimana cara pembuatan jadwal perawat saat ini, apakah secara terpusat atau dibagi setiap ward atau bangsal?

Answer: pembuatan Jadwal dilakukan secara per unit yang dilakukan oleh masing masing PJ unit. PJ unit mengumpulkan jadwal kepada bagian SDM RSIA maksimal tanggal 25 pada

setiap bulan untuk bulan berikutnya untuk diinputkan ke dalam sistem mereka.

5. Bagaimanakah aturan umum mengenai penjadwalan perawat?

Answer:

- Staff yang memperoleh shift malam sebanyak maka harus memperoleh libur dihari berikutnya
- Setiap staff dalam satu bulan harus mendapatkan bagian shift malam
- Staff memiliki maksimal jam kerja 7 jam per hari dalam 6 hari perminggu
- jam lembur dihitung minimal 1 jam

6. Siapa yang melakukan penjadwalan perawat?

Answer : Penjadwalan dilakukan oleh setiap pj unit atau bangsal

7. Berapa kali penjadwalan perawat dilakukan? Apakah setiap minggu, atau bulan?

Answer: Penjadwalan perawat dilakukan sekali dalam sat bulan

8. Apakah ada rotasi perpindahan perawat yang dilakukan?

Answer: ada, akan tetapi perpindahan jadwal perawat terseut biasanya tanpa sepengetahuan pihak SDM. Kebanyakan perawat mengganti shift mereka sendiri tanpa melakukan konfirmasi terlebih dahulu kepada pihak SDM.

9. Berapa shift yang dijalankan setiap harinya, dan berapa lama alokasi waktu setiap shift?

Answer: secara umum terdapat 3 shift dengan pembagian 7 jam yaitu

- Pagi: 07.00-14.00
- Sore : 14.00-21.00
- Malam: 21.00-07.00

untuk mengatasi jam kerja yang sibuk pada jam jam tertentu RSIA menerapkan shift middle yang berlangsung dari jam 10.00-17.00 atau jam 12.00-19.00

10. Berapa maksimum jam kerja yang didapatkan masing masing perawat dalam satu minggu?

Tidak ada maksimum jam kerja setiap minggunya. Yang ada hanyalah batas minimal libur untuk setiap bulannya. Setiap bulannya, jatah libur untuk setiap perawat diusahakan harus sama. Apabila tidak sama, maka perawat yang mendapat libur lebih sedikit akan diganti hari liburnya di bulan depan.

11. Bagaimana penjadwalan perawat yang sudah dilakukan saat ini? Apakah terdapat keluhan dari perawat mengenai penjadwalan yang sudah ada?

Answer: Penjadwalan perawat RSIA saat ini dilakukan secara manual oleh masing masing PJ bangsal atau unit RS. Hal tersebut mengakibatkan memakan banyak waktu.

12. Apa saja kekurangan penjadwalan yang telah diterapkan saat ini? apa yang perlu dioptimalkan misal di tingkat manajemen atau di tingkat perawat?

Answer: pihak SDM kesulitan untuk mengontrol jalannya penjadwalan dikarenakan banyak dari perawat yang berganti shift kerja tanpa ada izin terlebih dahulu. Hal yang ingin dioptimalkan oleh RS adalah bagaimana cara mengontrol jalannya penjadwalan dengan menggunakan sistem yang langsung terintegrasi langsung dengan fingerprint yang ada sehingga kecurangan atau pelanggaran Perawat akan saling tukar menukar jam dapat diminimalisasi.

Masalah lain juga adanya jam jam dimana rumah sakit sepi atau sangat ramai. Ketika rumah sakit keadaannya sangat ramai, masalah tersebut biasanya diselesaikan dengan cara mengoper perawat yang bekerja pada shift tersebut pada unit yang renggang ke unit yang padat pasien.

Namun apabila rumah sakit sepi, maka bagian SDM akan meliburkan perawatnya. Misal dokter yang bertugas di RS tersebut harus keluar negeri semuanya untuk mengikuti pekan ilmiah. Melihat hal ini maka SDM berinisiatif untuk meliburkan perawat yang berada pada shift dan unit tersebut.

13. Bagaimana skill yang dimiliki perawat apakah ada perawat khusus yang menangani kasus tertentu (Type skill masing-masing perawat)

Answer: bagian unit yang harus memiliki perawat skill khusus adalah:

- Ruang Operasi
- Rekam Medis

14. Bagaimana jadwal yang sudah diatur, tiba-tiba mengalami perubahan karena salah perawat meminta libur?

Answer: Jatah ambil cuti harus ditentukan masing masing perawat pada saat PJ unit melakukan penjadwalan. Dalam satu bulan perawat diberikan jatah cuti satu hari. Pengajuan cuti dilakukan perawat maksimal tanggal 25 untuk setiap bulannya. Untuk kasus berbeda, misalkan terdapat keluarga meninggal atau terkena musibah yang lain berarti dapat dikurangi dengan jatah cuti yang sudah dia tentukan pada awal penjadwalan.

15. Apakah ada perawat yang memiliki pengurangan jam kerja? (ex. perawat yang sedang hamil, perawat memiliki keterbatasan)

Answer: Tidak ada pengurangan jam kerja setiap shiftnya untuk perawat yang hamil. Namun terdapat pemberian cuti selama 3 bulan yaitu 1.5 bulan sebelum melahirkan dan 1,5 bulan sesudah melahirkan. Pengambilan cuti ini bersifat fleksibel.

## LAMPIRAN B – JADWAL MANUAL BULAN NOVEMBER 2017

Jadwal Bulan November 2017 Unit Instalasi Farmasi

No.	ID Staff	ID Skill	Minggu I						Minggu II						Minggu III						Minggu IV						Minggu V					
			R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	101	1	P	P	P	P	L	P	P	P	P	P	L	P	P	P	P	P	P	P	L	P	P	P	P	P	P	L	P	P	P	P
2	102	3	S	MS	S	S	MS	L	MS	MS	MS	S	S	L	S	S	S	S	L	MS	S	MS	S	S	MS	L	P	S	MS	S	MS	
3	103	4	M	M	LM	L	P	S	P	P	M	M	LM	L	CT	P	S	MS	MS	P	CT	MS	M	M	LM	L	P	S	S	S	L	P
4	104	4	P	P	L	MS	M	M	LM	L	P	P	CT	S	S	S	M	M	LM	L	P	P	S	S	M	M	LM	L	P	P	S	S
5	105	4	LM	L	S	S	S	P	M	M	LM	L	P	MS	P	CT	P	S	M	M	LM	L	P	P	S	S	M	M	LM	L	P	S
6	106	4	L	S	M	M	LM	L	S	S	S	CT	S	P	M	M	LM	L	S	S	S	S	S	CT	P	S	S	P	M	M	LM	L
7	107	4	S	S	P	P	L	S	S	S	S	M	M	LM	L	MS	P	P	S	S	M	M	LM	L	MS	P	S	S	L	S	M	M



No.	ID Staff	ID Skill	Minggu I					Minggu II					Minggu III					Minggu IV					Minggu V								
			R	K	J	Sb	M	Sn	S	l	R	K	J	Sb	M	Sn	S	l	R	K	J	Sb	M	Sn	S	l	R	K			
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
8	108	2	P	P	P	P	L	P	P	P	P	P	L	P	P	P	P	P	P	P	L	P	P	P	P	P	L	P	P	P	P

Jadwal Bulan November 2017 Unit Ruang Bayi dan NICU

No.	ID Staff	ID Skill	Minggu I					Minggu II					Minggu III					Minggu IV					Minggu V														
			R	K	J	Sb	M	Sn	S	l	R	K	J	Sb	M	Sn	S	l	R	K	J	Sb	M	Sn	S	l	R	K									
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30					
1	201	1	P	P	P	P	L	P	P	P	P	P	L	P	P	P	P	P	P	L	P	P	P	P	P	L	P	P	P	P							
2	202	4	C	P	P	P	S	P	L	P	M	P	L	L	P	C	P	P	L	L	L	M	M	L	M	L	C	P	S	P	P						
3	203	4	L	S	S	S	M	M	M	L	C	T	M	P	S	S	S	M	M	M	L	M	L	P	P	M	M	L	P	M	M	L					
4	204	4	P	M	M	L	L	S	S	P	S	S	M	L	M	L	M	L	M	L	M	L	C	T	C	T	S	S	P	P	M	M	L	M	L	S	S

No.	I D S t a f f	I D S k i l	Minggu I					Minggu II					Minggu III					Minggu IV					Minggu V																								
			R	K	J	S b	M	S n	S l	R	K	J	S b	M	S n	S l	R	K	J	S b	M	S n	S l	R	K	J	S b	M	S n	S l	R	K															
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30															
5	205	4	S	P	M	M	L	P	S	P	L	L	M	M	M	L	S	P	M	M	L	P	S	S	S	C	T	S	P	M	M																
6	207	4	M	M	L	L	P	P	M	M	L	M	M	S	P	M	M	S	L	S	S	S	S	P	C	T	M	M	L	M	L	C	T	S	S	M	S										
7	208	4	S	M	M	L	M	L	S	M	M	M	L	S	S	P	P	C	T	M	M	L	M	C	T	P	L	M	M	L	M	L	C	T	P	P											
8	209	4	M	M	L	M	L	S	M	S	P	M	M	L	M	L	L	M	S	L	M	M	P	P	L	P	M	S	S	S	M	M	L	M	L	M	S	C	T	M							
9	210	4	M	M	L	S	P	C	T	S	S	P	P	M	M	L	M	M	S	M	S	S	S	M	M	L	L	P	M	S	P	M	M	L	L	S	M	M	L	S							
10	211	4	L	S	S	P	M	S	M	M	L	M	L	S	P	M	S	C	T	L	M	L	M	L	M	L	P	P	S	M	M	L	M	L	C	T	S	M	S	M	M	L	M				
11	212	4	M	S	S	P	M	M	L	M	L	C	T	S	S	M	M	S	M	i	d	S	M	S	M	M	L	M	L	M	S	S	S	M	S	M	M	L	C	T	M	M	L	M	L	C	T
12	213	4	P	P	S	M	M	L	M	L	M	S	M	M	L	M	M	M	M	M	M	L	M	L	P	M	S	M	M	L	L	P	S	P	S	P	M	S	M	M	L	M	M	S	M		

No	ID Staff	ID Skill	Minggu I					Minggu II					Minggu III					Minggu IV					Minggu V												
			R	K	J	Sb	M	S	S	R	K	J	Sb	M	S	S	R	K	J	Sb	M	S	S	R	K										
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30			
13	214	44	L	L	P	M	S	M	M	M	L	L	P	M	S	P	P	M	M	L	P	S	M	M	L	L	P	P	M	S	P	M	M	L	M

### Jadwal Bulan November 2017 Unit SIM Rekam Medis dan Registrasi

No	ID Staff	ID Skill	Tanggal																																												
			M	S	S	R	K	J	S	M	S	S	R	K	J	S	M	S	S	R	K	J	S	M	S	S	R	K	J	S	M	S	S														
1	301	44	L	P	P	S	M	M	L	L	M	L	P	L	C	T	C	T	P	L	S	S	S	P	P	P	C	T	S	M	D	L	M	M	L	M	L	P	D	L							
2	302	44	M	D	L	S	S	L	P	P	L	L	C	T	S	P	M	D	L	M	M	M	M	L	M	L	M	D	L	P	L	S	S	P	L	S	P	P	M	D	L	M	M	L	M		
3	303	44	S	M	M	L	L	S	S	L	P	P	L	M	M	L	M	L	L	P	L	P	M	M	M	L	M	L	S	S	P	M	D	L	L	L	P	P	S	S	S	S	S	S			
4	304	44	M	L	M	L	M	S	S	M	M	L	L	M	D	L	S	S	C	T	C	T	S	M	M	D	L	L	M	L	P	M	D	L	L	P	P	M	D	L	L	L	S	M	L	P	P
5	305	44	P	P	M	M	M	L	P	S	S	M	D	L	S	P	P	M	D	L	S	M	M	L	M	L	S	M	D	L	S	P	M	M	M	L	L	L	L	C	T	S	S	L	L		

6	3	6	4	L	S	M	D	L	P	P	P	S	M	D	L	M	M	M	L	M	D	L	S	P	P	P	L	S	S	M	M	L	M	L	L	S	M	P	L	S	M	D	L	L	M
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Jadwal Bulan November 2017 Unit Instalasi Gizi(1)

NO	TANGGAL														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	P	P	P	P	L	P	P	P	P	P	P	L	P	P	P
2	P	L	S	P1	P	P1	S	S	P	CT	L	L	P1	P1	S
3	L	MD	MD	MD	P1	MD	CT	MD	S	P1	P1	L	L	S	MD
4	MD	P2	P1	L	S	S	P	CT	CT	L	MD	S	S	MD	P1
5	S	S	CT	S	MD	L	MD	P	S	S	S	P1	MD	L	CT
6	P2	S	S	S	S	L	S	S	S	S	S	L	S	P	P
7	P1	P1	S	CT	L	S	P1	P1	P1	MD	S	MD	L	S	S
8	P2	L	P2	CT	L	P2	P2	P2	L	P	P	S	P2	P2	P2
9	S	S	P	P	P2	L	S	S	S	P2	P2	P2	L	L	CT

NO	TANGGAL														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
10	L	CT	S	S	MD	M	M	M	LM	L	P	P	M	M	LM
11	P	P	MD	MD	P	L	S	P	P	P	MD	MD	L	S	P
12	S	S	M	M	M	LM	L	MD	S	S	S	L	CT	CT	M
13	MD	MD	P	P	L	P	P	S	M	M	LM	L	S	S	P
14	M	M	LM	CT	S	MD	S	P	P	P	M	M	LM	CT	S
15	P	P	MD	MD	P	S	L	M	M	LM	L	S	P	P	MD
16	P	P	P	S	L	PS	P	S	S	S	P	L	PS	S	P
17	S	S	S	P	PS	L	S	P	P	P	S	PS	L	P	S
18	S	S	S	P	PS	L	P	S	S	P	P	L	PS	S	S
19	CT	CT	CT	S	L	PS	S	CT	CT	S	S	PS	L	CT	CT

Jadwal Bulan November 2017 Unit Instalasi Gizi(2)

NO	TANGGAL														
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	P	P	P	MOD	P	P	P	P	P	L	L	P	P	P	P
2	S	S	P	P1	S	L	P	P2	S	S	P1	P2	P	P1	S
3	S	P	MD	MD	P1	MD	MD	S	CT	L	S	MD	MD	MD	P
4	MD	L	S	S	MD	S	S	CT	P	P1	MD	P1	P1	L	CT
5	P	MD	L	CT	CT	P1	CT	MD	MD	MD	L	S	S	S	MD
6	P2	P2	P2	P	S	S	S	S	P2	P	P	L	S	P	S
7	P1	P1	P1	S	S	L	P1	P1	P1	S	CT	S	L	CT	P1
8	L	S	S	S	P	P	CT	CT	S	P2	P2	S	S	P2	P2
9	S	S	S	P2	P2	P2	P2	S	L	S	S	CT	P2	S	S
10	L	CT	MD	MD	P	M	M	LM	L	S	S	MD	S	P	P
11	MD	S	S	P	L	S	P	P	P	CT	L	CT	CT	CT	CT
12	M	LM	L	S	S	P	P	CT	CT	M	M	LM	L	S	S

NO	TANGGAL														
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
13	P	M	M	LM	L	S	S	M	M	LM	L	S	P	MD	MD
14	S	P	P	L	MD	MD	CT	CT	L	L	P	M	M	LM	L
15	L	MD	MD	M	M	LM	L	S	S	P	P	P	MD	M	M
16	P	P	S	PS	L	CT	CT	CT	CT	PS	PS	L	S	P	P
17	S	S	P	L	PS	S	P	P	P	CT	L	PS	P	S	S
18	P	P	L	S	PS	S	P	P	P	S	PS	L	P	S	S
19	S	S	PS	P	L	P	S	S	S	P	L	PS	S	P	P

Jadwal Bulan November 2017 Unit Rawat Jalan dan IGD (1)

No	ID Staf	Minggu I							Minggu II							Minggu III						
		R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	401	M S	S	P	L	M	M	L M	M S	S	M S	L	L	S	M S	M S	S	S	P	P	CT	
2	402	S	P	S	P	L	P	MS	P	M S	CT	S	S	M S	P	L	CT	M	M	L M	S	
3	403	L	S	M S	M S	L S	CT	S	S	M	M	L M	L	P	S	S	M S	CT	S		S	
4	404	S	S	M	M	L M	L	MS	S	S	S	M S	L	S	S	P	M S	L	M S	M	M	
5	405	P	M S	S	M S	L	S	P	M S	L	CT	M	M	L M	S	S	P	M S	L	S	S	
6	406	M	M	L M	L	P	S	S	CT	M S	S	P		M	M	L M	L	P	M S	L	S	
7	407	L M	S	L	P	L	S	S	S	P	P	M S	L	S	S	M	M	L M	L	L	M S	
8	408	S	M S	S	S	L	M S	M	M	L M	S	P	P	P	M S	M S	S	S	P	L	P	



Jadwal Bulan November 2017 Unit Rawat Jalan dan IGD (2)

No.	ID Staff	Minggu IV							Minggu V		
		Sl	R	K	J	Sb	M	Sn	Sl	R	K
		21	22	23	24	25	26	27	28	29	30
1	401	S	MS	S	S	P	L	P	MS	CT	MS
2	402	S	S	MS	L	S	P	M	M	LM	L
3	403	M	L	P	MS	MS	S	S	S	M	L
4	404	LM	P	MS	S	L	CT	MS	S	S	P
5	405	S	M	L	P	P	CT	S	P	MS	M
6	406	S	S	M	M	LM	L	S	MS	S	S
7	407	P	MS	S	S	MS		CT	S	P	MS
8	408	MS	S	S	S	M	M	LM	L	MS	S

Jadwal Bulan November 2017 Unit Kamar Operasi (1)

No	ID staff	ID Skill	Minggu I					Minggu II					Minggu III								
			R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	601	4	P	P	P	P	P	CT	CT	S	S	P	P	L	P	S	P	S	S	L	P
2	602	4	P	S	S	S	L	P	P	P	P	L	S	P	S	P	S	P	L	P	P
3	603	4	P	S	P	L	P	S	S	P	P	S	P	P	L	P	P	S	P	S	L
6	606	4	S	P	P	P	L	S	S	P	P	P	CT	L	S	P	P	P	P	P	P
4	604	4	L	P	S	S	P	P	P	L	P	S	S	P	P	S	L	P	S	S	P
5	605	4	P	L	P	P	P	P	L	S	S	P	P	P	P	CT	S	P	P	P	L

Jadwal Bulan November 2017 Unit Kamar Operasi (2)

No	ID staff	ID Skill	Minggu IV							Minggu V			
			Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K
			20	21	22	23	24	25	26	27	28	29	30
1	601	4	S	S	S	P	L	P	P	P	P	S	P
2	602	4	P	P	P	L	P	S	P	S	S	L	S
3	603	4	S	P	P	S	P	CT	L	S	P	P	P
6	606	4	L	S	S	S	P	P	L	P	S	S	S
4	604	4	P	P	L	P	S	P	P	P	P	P	CT
5	605	4	P	P	P	P	S	S	P	L	P	P	P

## LAMPIRAN C – JADWAL HASIL OPTIMASI BULAN DESEMBER 2017

Hasil Optimasi Bulan Desember 2017 Unit Farmasi

No	Kode Staf	J	Sb	M	Su	Sl	R	K	J	Sb	M	Su	Sl	R	K	J	Sb	M	Su	Sl	R	K	J	Sb	M	Su	Sl	R	K	J	Sb	M
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	101	P	P	S	P	P	P	P	P	P	S	P	P	P	P	P	P	L	P	P	P	P	P	P	L	P	P	P	P	P	L	P
2	102	S	S	L	S	S	S	S	S	S	M	P	L	S	S	S	S	S	S	P	S	S	S	L	S	L	S	S	S	S	L	
3	103	M	M	LM	M	L	P	S	S	P	L	M	P	LM	L	P	P	L	S	MS	MS	M	L	LM	L	LM	L	S	S	MS	M	L
4	104	MS	L	M	L	LM	P	P	P	S	L	S	M	M	M	LM	L	P	P	S	S	MS	L	M	L	L	S	P	P	S	S	
5	105	S	S	S	L	M	M	LM	L	P	P	S	S	P	L	M	M	LM	M	P	L	S	S	L	S	M	M	LM	L	P	P	S
6	106	P	P	L	S	S	L	M	M	LM	S	L	P	S	S	L	S	M	L	LM	S	P	P	S	S	P	P	M	M	LM	P	L
7	201	LM	L	P	P	P	S	L	L	M	L	LM	L	L	P	S	L	S	MS	M	M	LM	M	P	P	S	S	MS	MS	M	L	LM
8	108	P	P	L	P	P	L	P	P	L	L	P	S	P	P	P	P	L	P	P	P	P	P	P	M	P	P	P	P	P	P	M

### Hasil Optimalisasi Bulan Desember 2017 Unit Ruang Bayi dan NICU

No	Kode Staf	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	201	P	P	L	P	P	P	P	P	P	MS	P	P	P	P	P	P	L	P	P	P	P	P	P	L	P	P	P	P	P	P	L
2	202	M	M	LM	L	P	P	S	S	MS	L	M	M	LM	L	P	P	S	S	L	L	M	M	LM	L	P	P	S	S	MS	MS	M
3	203	MS	MS	M	M	LM	L	P	P	S	S	MS	L	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	L
4	204	S	S	MS	L	M	M	LM	L	P	P	S	S	MS	L	M	M	LM	L	P	P	S	L	LM	L	M	M	LM	P	P	P	S
5	205	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	L	L	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P
6	206	LM	L	P	P	S	S	L	MS	M	M	LM	L	P	P	S	S	L	MS	M	M	LM	L	P	L	S	S	MS	L	M	M	LM
7	207	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	L	M	M	L	L	P	P	S	S	L	MS	M
8	208	MS	MS	M	M	LM	L	P	P	S	S	MS	L	M	M	LM	MS	P	P	S	S	L	P	M	M	LM	L	P	MS	S	S	MS
9	209	S	S	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	L	L	M	M	LM	L	P	P	S
10	210	P	P	S	S	MS	L	M	M	LM	L	P	P	S	S	MS	P	M	M	LM	L	P	L	S	S	L	MS	M	M	LM	L	P
11	211	LM	L	P	P	S	S	MS	L	M	M	LM	L	P	P	S	S	L	MS	M	M	LM	S	P	P	S	S	MS	M	M	LM	M
12	212	M	M	LM	L	P	P	S	S	MS	L	M	M	LM	L	P	L	S	S	L	MS	M	M	LM	P	P	P	S	S	L	MS	M
13	213	MS	MS	M	M	LM	L	P	P	S	S	MS	MS	M	M	LM	L	P	P	S	S	L	MS	M	M	LM	L	P	P	S	S	L

### Hasil Optimalisasi Bulan Desember 2017 Unit SIM Rekam Medis dan Registrasi

No	Kode Staf	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
1	301	P	S	S	S	MS	MS	M	M	LM	L	P	L	P	S	S	S	L	MS	M	M	LM	L	P	P	P	S	S	S	S	MS	L	M
2	302	P	P	P	S	S	S	L	L	M	M	LM	P	P	P	P	S	S	S	MS	L	M	M	LM	L	P	P	P	S	S	S	L	M
3	303	LM	L	P	P	P	S	S	L	P	M	M	LM	L	P	P	P	S	S	S	P	L	M	M	LM	L	P	P	P	S	S	S	
4	304	M	M	LM	P	P	P	P	S	S	S	L	S	M	M	LM	L	P	P	P	S	S	S	L	L	M	M	LM	L	P	P	P	
5	305	L	MS	M	M	LM	L	P	P	P	S	S	S	L	MS	M	M	LM	L	P	L	L	S	S	S	S	MS	M	M	LM	L	P	
6	306	S	S	P	L	M	M	LM	L	P	L	P	L	S	S	P	P	M	M	LM	P	P	P	P	S	L	S	P	P	M	M	LM	



### Hasil Optimasi Bulan Desember 2017 Unit Rawat Jalan dan IGD

No	Kode Staf	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
		31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61
1	501	M	M	LM	L	P	L	S	S	S	S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S	L	MD	L	S	M	
2	502	MS	S	M	M	LM	P	P	P	S	S	S	S	L	MD	L	S	M	M	LM	L	P	P	S	S	S	S	L	MD	L		
3	503	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S	L	
4	504	S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S	L	MD	MS	S	M	M	L	S	P	P	S	S	S		
5	505	S	S	S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	L	MD	L	S	M	M	LM	L	L	P	S	S		
6	506	P	S	S	S	S	L	MD	L	S	M	M	LM	L	P	P	S	S	S	S	S	S	MD	L	L	M	M	LM	L	P	P	
7	507	L	P	P	S	S	S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	S	S	S	LM	MD	L	S	M	LM	L	
8	508	M	LM	L	P	P	S	S	S	S	S	L	MD	MS	S	M	M	LM	L	P	P	S	S	S	L	MD	P	S	M	M		

### Hasil Optimasi Bulan Desember 2017 Unit Kamar Operasi

No	Kode Staf	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	Sb	M
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	601	P	P	L	P	S	L	P	P	S	P	P	P	S	P	S	L	P	P	S	S	P	P	S	L	S	L	P	P	S	S	P
2	602	S	P	L	S	P	S	L	P	P	P	S	P	P	S	P	S	P	P	S	S	P	P	L	P	S	L	P	P	S	P	
3	603	S	S	P	P	S	P	S	L	P	P	S	S	P	P	S	P	L	L	P	L	S	S	P	P	S	P	S	L	P	P	P
4	604	P	S	P	P	P	S	P	S	L	P	P	S	P	P	S	L	S	L	P	P	S	P	P	P	P	S	P	S	L	P	P
5	605	P	P	P	P	P	S	P	S	L	P	P	S	P	P	P	P	P	S	P	P	P	P	S	P	P	P	S	P	S	L	L
6	606	L	P	P	S	P	P	P	S	P	L	L	P	P	S	P	P	S	P	S	L	P	P	P	P	P	P	P	S	P	S	L