

Perancangan Aplikasi Penjadwalan untuk Optimalisasi Waktu Produksi dengan Metode Hill Climbing

Dwi Setyawan.¹, Heribertus Himawan, M. Kom.²

^{1,2} Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro

Jalan Nakula I No 5, Semarang, Indonesia

Email: finalhope7@gmail.com¹, himawan26@gmail.com²

PT. Sai Apparel Industries Ltd adalah perusahaan yang bergerak di bidang garment, setiap tahunnya perusahaan ini memproduksi sekitar 7 juta pakaian yang berupa celana, baju, jaket dan lain-lain yang semuanya adalah produk ekspor. Tetapi sering terjadi keterlambatan pengiriman barang pada konsumen (buyer) menurut survey penulis dalam persentase keterlambatan produksi barang mencapai 15% / tahun dari total semua produksi yang didapat dari catatan rekapitulasi produksi tahun 2012, hal ini disebabkan buruknya manajemen waktu. Oleh sebab itu perlu adanya penjadwalan yang dapat mengoptimalkan produksi tersebut dengan waktu dan sumber daya yang ada yaitu dengan aplikasi penjadwalan optimalisasi waktu produksi dengan metode hill climbing yang akan diterapkan di PT. Sai Apparel Industries Ltd. Metode hill climbing digunakan untuk melakukan pencarian order serta penempatannya ke dalam antrian produksi untuk mencari bentuk antrian yang paling pendek dalam waktu produksinya sehingga mampu mengoptimalkan waktu produksi. Dalam hal ini bagaimana merancang aplikasi penjadwalan untuk optimalisasi waktu produksi yang nantinya dapat menghasilkan sebuah penjadwalan produksi sesuai berdasarkan permasalahan yang dihadapi oleh PT. Sai Apparel Industries Ltd dengan mengoptimalkan waktu kerja serta antriannya. Hasilnya yaitu sebuah grafik jadwal yang dapat menginformasikan bagaimana panjang waktu produksi serta antrian pengerjaannya dengan urutan alur produksi yang lebih optimal dari pada sebelumnya.

Kata kunci : penjadwalan, optimal, hill climbing, perhitungan, grafik jadwal

1. Pendahuluan

PT. Sai Apparel Industries Ltd adalah perusahaan yang bergerak di bidang garment, setiap tahunnya perusahaan ini dapat memproduksi sekitar 7 juta pakaian yang berupa celana, baju, jaket dan lain-lain yang semuanya adalah produk ekspor, sayangnya sering terjadi keterlambatan pengiriman barang pada konsumenn yang telah memesanya (buyer) menurut survey yang penulis lakukan prosentasi keterlambatan produksi barang mencapai 15% / tahun dari total semua produksi yang didapat dari catatan rekapitulasi produksi pada tahun 2012, hal ini disebabkan karena manajemen waktu yang ada dan kurang optimalnya menghitung kalkulasi waktu yang dibutuhkan untuk memproduksi suatu produk. Oleh sebab itulah perlu adanya sistem aplikasi yang dapat mengoptimalkan produksi tersebut dengan waktu dan sumberdaya yang ada. Untuk itu maka diperlukanya aplikasi penjadwalan berdasarkan penerapan ilmu metode hill climbing sehingga dapat menghasilkan

penjadwalan produksi dengan mengoptimalkan waktu kerja serta antriannya

1.1 Penjadwalan (Scheduling)

Penjadwalan dalam buku Jeffrey W. Herrmann (2007) memuat tentang definisi penjadwalan rinci oleh Coxet al. (1992) adalah sebagai "tugas sebenarnya memulai dan / atau tanggal penyelesaian pada operasi atau kelompok operasi untuk menunjukkan kapan ini harus dilakukan jika urutan manufaktur akan selesai tepat waktu". Dari definisi di atas maka penjadwalan adalah suatu kegiatan untuk mencanangkan waktu kapan dimulainya suatu kegiatan yang dilaksanakan pada periode waktu tertentu sesuai dengan batasan - batasannya.

Jadwal produksi yaitu mengkoordinasikan kegiatan-kegiatan untuk meningkatkan produktivitas dan meminimalkan biaya operasi. Sebuah jadwal produksi dapat mengidentifikasi sumber konflik,

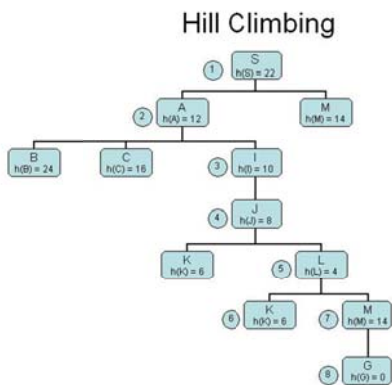
mengontrol pelepasan pekerjaan ke toko, memastikan bahwa bahan baku yang diperlukan memerintahkan dalam waktu, menentukan apakah janji pengiriman dapat dipenuhi, dan mengidentifikasi periode waktu yang tersedia untuk pemeliharaan preventif. Dalam hal ini yang akan dibuat adalah penjadwalan produksi.

1.2 Hill Climbing

Metode Hill Climbing Search adalah teknik pencarian untuk menyelesaikan masalah optimasi. Idenya adalah untuk memulai dengan solusi paling awal (yaitu, mulai dari dasar bukit) dan kemudian berulang kali meningkatkan solusi (berjalan ke atas bukit) sampai beberapa kondisi dimaksimalkan (puncak bukit tercapai). Atau lebih singkatnya menghasilkan dan coba ditambah dengan arah untuk bergerak.

Metodologi Hill Climbing adalah

- 1) Membuat sebuah solusi sub-optimal yang memenuhi kendala dari masalah
- 2) Mengambil solusi dan membuat perbaikan pada bagian atasnya
- 3) Berulang meningkatkan solusi sampai tidak ada lagi perbaikan yang diperlukan / mungkin.



Gambar 1.1 Gambaran dari teknik Hill Climbing dalam pencarian nilai terkecil

Contoh Algoritma Hill Climbing :

```
function HILL-CLIMBiNG(problem)
    returns a solution state
inputs: problem, a problem
static: current, a node ; next, anode
current ← MakeNode(initial-state[prblem])
loop do
    next ← a highest-valued successor of current
    if VALUE[next] < VALUE[current] then return
    current ← next
end
```

Rumus dari Hill Climbing adalah upaya untuk memaksimalkan (atau meminimalkan) fungsi target $f(\mathbf{x})$, di mana \mathbf{X} adalah vektor nilai kontinu dan / atau diskrit. Pada setiap iterasi, Hill Climbing akan menyesuaikan satu elemen dalam \mathbf{X} dan menentukan apakah perubahannya meningkatkan nilai $f(\mathbf{x})$. (Catatan bahwa ini berbeda dari metode gradient descent, yang mengatur semua nilai \mathbf{X} di setiap iterasi sesuai dengan gradien bukit.) Dengan Hill Climbing, setiap perubahan yang meningkatkan $f(\mathbf{x})$ diterima, dan proses berlanjut sampai tidak ada perubahan dapat ditemukan untuk meningkatkan nilai $f(\mathbf{x})$ \mathbf{X} kemudian dikatakan "lokal optimal".

Dalam ruang vektor diskrit, setiap nilai yang mungkin \mathbf{X} untuk dapat digambarkan sebagai simpul dalam grafik. Hill Climbing akan mengikuti grafik dari titik ke titik, selalu meningkat secara lokal (atau penurunan) nilai $f(\mathbf{x})$, sampai maksimum lokal (atau minimum lokal) tercapai.

Ada 2 macam tipe Hill Climbing yaitu

- 1) Simple Hill Climbing

Algoritma :

- a) Mulai dari keadaan awal, lakukan pengujian: jika merupakan tujuan, maka berhenti; dan jika tidak, lanjutkan dengan keadaan sekarang sebagai keadaan awal.
- b) Kerjakan langkah-langkah berikut sampai solusinya ditemukan, atau sampai tidak ada operator baru yang akan diaplikasikan pada keadaan sekarang :
 - 1) Cari operator yang belum pernah digunakan; gunakan operator ini untuk mendapatkan keadaan yang baru.
 - 2) Evaluasi keadaan baru tersebut.
- c) Jika keadaan baru merupakan tujuan, keluar.
 - 1) Jika bukan tujuan, namun nilainya lebih baik daripada keadaan sekarang, maka jadikan keadaan baru tersebut menjadi keadaan sekarang.
 - 2) Jika keadaan baru tidak lebih baik daripada keadaan sekarang, maka lanjutkan iterasi.
 - 3) Pada simple hill climbing ini, ada 3 masalah yang mungkin, yaitu:
 - a. Algoritma akan berhenti kalau mencapai nilai optimum lokal.
 - b. Urutan penggunaan operator akan sangat berpengaruh pada penemuan solusi.
 - c. Tidak diijinkan untuk melihat satupun langkah sebelumnya.

- 2) Stepest Ascent Hill Climbing

Algoritma :

- a) Mulai dari keadaan awal, lakukan pengujian: jika merupakan tujuan, maka berhenti; dan jika tidak, lanjutkan dengan keadaan sekarang sebagai keadaan awal.
- b) Kerjakan hingga tujuan tercapai atau hingga iterasi tidak memberikan perubahan pada keadaan sekarang.
 1. Tentukan SUCC sebagai nilai heuristik terbaik dari successor-successor.
 2. Kerjakan untuk tiap operator yang digunakan oleh keadaan sekarang:
 - a) Gunakan operator tersebut dan bentuk keadaan baru.
 - b) Evaluasi keadaan baru tersebut.
 - c) Jika merupakan tujuan, keluar. Jika bukan, bandingkan nilai heuristiknya dengan SUCC. Jika lebih baik, jadikan nilai heuristik keadaan baru tersebut sebagai SUCC. Namun jika tidak lebih baik, nilai SUCC tidak berubah.
 3. Jika SUCC lebih baik daripada nilai heuristik keadaan sekarang, ubah node SUCC menjadi keadaan sekarang.

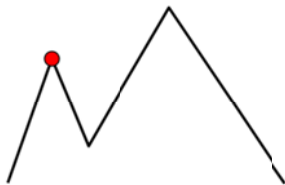
Kelebihan Hill Climbing:

- 1) Lebih efisien dari segi memori, bila dibandingkan dengan metode steepest - ascent hill climbing
- 2) boleh digabungkan dengan metode pencarian yang lain, memberikan permulaan yang baik

Kekurangan Hill Climbing:

- 1) Tidak boleh kembali ke state semula atau backtracking
- 2) Bisa terjebak di local maxima, plateaus atau ridges.

Pengertian lebih mendalam pada kelemahannya yang pertama adalah Local maxima sebuah states yang lebih baik dari semua tetangganya, tapi tidak lebih baik daripada beberapa states lain jauh. Jadi hasil yang dituju tidak maksimal karena kurang luasnya dan dalamnya pencarian.



Gambar 1.2 Kondisi terjadinya Local maxima

Plateu adalah sebuah daerah datar dari ruang pencarian di mana semua states tetangga memiliki nilai yang sama. Hampir sama seperti Local mxima

tetapi tujuan yang dicapai memiliki kesamaan dengan tetangga dan akhirnya kondisi paling awal tertinggi yang dihasilkan.



Gambar 1.3 Kondisi terjadinya Plateu

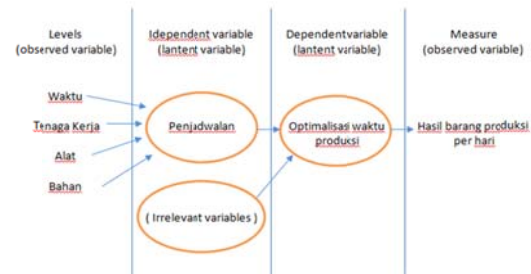
Ridge adalah local optimum yang lebih disebabkan karena ketidakmampuan untuk menggunakan 2 operator sekaligus.

Ways Out adalah

- 1) Mundur ke beberapa simpul sebelumnya dan mencoba berjalan ke arah yang berbeda.
- 2) Membuat lompatan besar untuk mencoba untuk mendapatkan di bagian baru.
- 3) Pindah ke beberapa arah sekaligus.

1.3 Kerangka Pemikiran

Kerangka pemikiran pada penelitian ini adalah “Bagaimana pengaruh penjadwalan produksi pada optimalisasi produksi?”. Dalam hal ini variable bebas adalah penjadwalan dan variabel terkait adalah optimalisasi produksi.



Gambar 1.4 Bagan kerangka penelitian

		Variable Type	
Mewakili efek → V. Bebas	Efisiensi waktu produksi	Optimal Produksi	
	Kualitas produk	Antara waktu yang digunakan 100% dengan waktu legans (refreshing)	
Mewakili sebab → V. Terikat	Kebutuhan perubahan	Waktu	
	Penjadwalan	Tenaga kerja	
		Alat	
		Bahan	
		Konsep	Diteliti

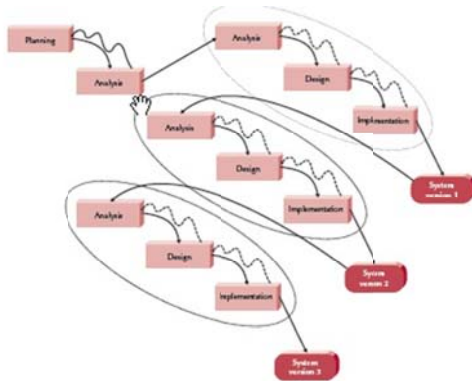
Gambar 1.5 Bagan kerangka tipe variable

Dari gambar di atas maka diketahui level yaitu waktu, tenaga kerja, alat dan bahan dimana hal tersebut yang mempengaruhi dibuatnya suatu jadwal dalam hal ini termasuk variabel bebas. Jadwal digunakan untuk mengatur kapan dan berapa lamanya waktu produksi serta kegiatan apa saja yang dilakukan selama produksi. Dan hubungannya dengan optimalisasi produksi (variabel terikat) adalah dari berapa jumlah produksi per hari (variabel yang dicari). Dari itu semua maka penelitian ini dilakukan.

2. METODE PENELITIAN

2.1. Metode Pengembangan Sistem

Metode yang digunakan untuk dalam penelitian adalah model *Rapid Application Development (RAD)*, sehingga model inilah yang akan dibahas, langkah – langkah metode RAD yaitu



Gambar 2.1 Model RAD

1. Planing

Tahap dimana dibuatnya perancangan aplikasi yang akan dibuat yang awalnya tujuan mengapa aplikasi ini dibuat. Kemudian dari itu dibuat gambaran besar dari apa yang akan diinputkan, proses yang diinginkan serta keluaran yang dibutuhkan.

2. Analisis/Definisi Kebutuhan

Ada dua aktifitas pada tahap ini yaitu:

- Analisis Kebutuhan, yang menghasilkan garis besar kebutuhan.
- Definisi Kebutuhan, yang menghasilkan dokumen kebutuhan. Dalam tahap ini, sistem analis harus menggali informasi mengenai fungsi, sifat, tujuan dan kendala-kendala yang ada di dalam sistem, yang kemudian dituangkan menjadi definisi kebutuhan yang jelas.

Tujuan dari tahap ini adalah untuk mengetahui bagaimana sebenarnya sistem yang akan dikembangkan.

3. Desain Sistem dan *Software*

Tahap ini terdiri atas:

- Desain Sistem, yang menghasilkan konfigurasi sistem secara keseluruhan termasuk penjelasan kebutuhan *hardware* dan *software*.
- Desain *Software*, yang merupakan proses *multistep* berfokus pada 4 atribut pemrograman: struktur data, arsitektur *software*, spesifikasi *interface*, dan algoritma yang digunakan.

Tujuan tahap ini adalah untuk menerjemahkan kebutuhan menjadi representasi *software* yang bisa diukur, sebelum dilakukan pemrograman/pengkodean.

4. Implementasi dan *Testing Unit*

Dalam tahap ini, desain yang telah dibuat diterjemahkan dalam bentuk kode program yang dapat dieksekusi dan dimengerti oleh mesin. Kemudian dilakukan pengujian tiap-tiap program atau unit program untuk memperbaiki *error* dalam penulisan kode dan untuk meyakinkan bahwa fungsi-fungsi yang dibentuk dapat berjalan sesuai keinginan. Tujuan tahap ini adalah untuk menghasilkan unit program yang dapat dieksekusi dan valid.

5. Sistem Versi

Adalah versi dari program yang dibuat dari awalnya jadi tapi masih memiliki banyak kekurangan hingga versi yang paling baru yang sudah memperbaiki kekurangannya sesuai keinginan user.

2.2 Penelitian korelasi

Penelitian korelasi adalah suatu penelitian yang melibatkan tindakan pengumpulan data guna menentukan, apakah ada hubungan dan tingkat hubungan antara dua variabel yaitu antara Penjadwalan dengan Optimalisasi Waktu Produksi. Adanya hubungan dan tingkat variabel ini penting, karena dengan mengetahui tingkat hubungan yang ada, peneliti akan dapat mengembangkannya sesuai dengan tujuan penelitian.

Penelitian korelasi mempunyai tiga karakteristik penting, yaitu:

- Penelitian korelasi tepat jika variabel kompleks dan peneliti tidak mungkin melakukan manipulasi dan mengontrol variabel seperti dalam penelitian eksperimen.
- Memungkinkan variabel diukur secara intensif dalam setting (lingkungan) nyata.
- Memungkinkan untuk mendapatkan derajat asosiasi yang signifikan.

Tujuan penelitian korelasional adalah untuk :

1. Mendeteksi sejauh mana variasi-variasi pada suatu faktor berkaitan dengan variasi-variasi pada satu atau lebih faktor lain berdasarkan pada koefisien korelasi.
2. Menentukan hubungan antara variabel, atau untuk menggunakan hubungan tersebut untuk membuat prediksi.

2.3 Perancangan

Untuk menentukan optimalisasi waktu produksi, ada beberapa teknik pengumpulan data dan variable yang digunakan sebagai berikut:

a. Studi Lapangan

Yaitu data secara langsung dapat diperoleh dengan meninjau dan mengamati secara langsung terhadap obyek yang diteliti melalui :

1. Wawancara (*Interview*)

Selama pengamatan langsung penulis mengadakan tanya jawab secara langsung untuk mendapatkan kelengkapan data. Penulis mengajukan beberapa pertanyaan yang berkaitan dengan masalah yang dibahas dan prosedur sistem yang dilakukan.

2. Survey

Pengumpulan data dengan pencatatan secara cermat terhadap proses sistem produksi yang ada pada PT. SAI APPAREL INDUSTRIES Ltd Semarang

b. Studi Pustaka

Untuk membantu penulis dalam pembuatan tugas akhir ini, maka penulis melakukan pendekatan – pendekatan dengan tujuan yaitu dengan membaca buku-buku dan referensi-referensi yang ada agar tidak menyimpang dari ketentuan yang ditetapkan

3. ANALISA DAN PERANCANGAN SISTEM

3.1 Kebutuhan Perangkat Keras

Spesifikasi hardware pendukung yang akan digunakan dalam proses pengembangan aplikasi adalah :

- a. Processor Intel Pentium Dual Core Centrino
- b. Memory 2 GB
- c. Hardisk 120 GB

3.2 Kebutuhan Perangkat Lunak

Dalam proses pembangunan aplikasi penjadwalan produksi pada PT. Sai Apparel Industries Ltd. Beberapa perangkat lunak yang digunakan untuk membangun aplikasi webserver.

a. XAMPP

Merupakan akronim dari cross-platform, Apache HTTP Server, MySQL, PHP, Perl, Tomcat. XAMPP didesain untuk memudahkan programmer dalam mengembangkan aplikasi web

berbasis PHP dengan menyediakan server lokal sehingga tidak dibutuhkan akses internet untuk menguji aplikasinya.

b. Adobe Dreamweaver

Merupakan aplikasi bantuan untuk membangun aplikasi penjadwalan produksi pada PT. Sai Apparel Industries Ltd berdasarkan bahasa pemrograman CSS, PHP, Javascript dengan menggunakan framework Code Ignited.

c. Firefox dan Google Chrome

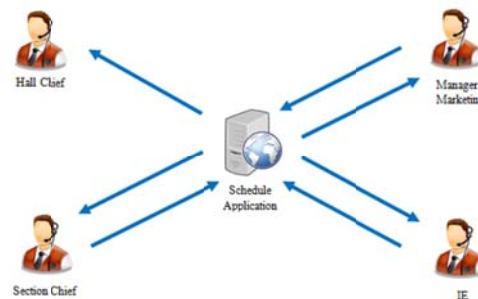
Merupakan aplikasi pendukung untuk membuka aplikasi penjadwalan produksi dan untuk menguji aplikasinya.

d. Sistem Operasi

Penulis menggunakan sistem operasi Windows 7 untuk mengembangkan aplikasi web. Akan tetapi pada proses pengerjaannya tidak menutup kemungkinan terkadang penulis menggunakan sistem operasi yang berbeda.

3.3 Perancangan Sistem.

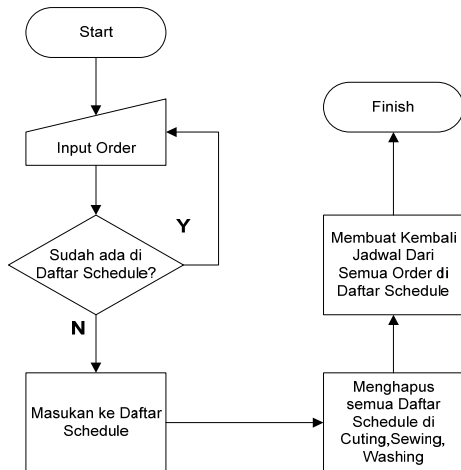
3.3.1 Arsitektur Umum



Gambar 3.1 Arsitektur Sistem

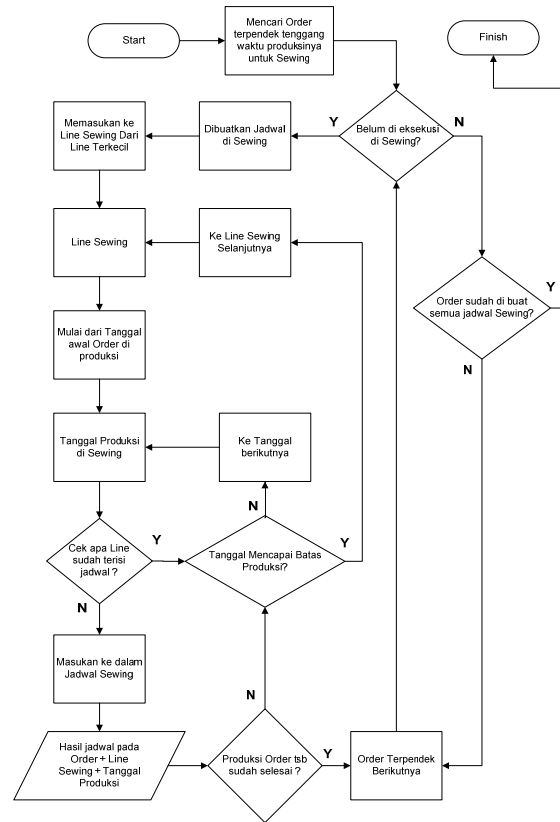
Alur proses dimulai dari input data *cuting* dan *sewing* dari Section Chief .Setelah itu di ikuti input *data motive, color* dan *fabric* yang menjadi acuan untuk menginput *style* yang berisi estimasi waktu produksinya untuk jabatan IE. Dilanjutkan jabatan Manager Marketing untuk menginput *buyer* serta order yang meliputi jenis *style* nya yang diproduksi oleh suatu *buyer*. Dan pada akhirnya semua itu akan dilaporkan kepada Hall Chief.

3.3.2 Perancangan Diagram



Gambar 3.2 Flowchart Pembentukan Jadwal secara umum dalam Aplikasi Optimalisasi Penjadwalan

Flowchart di atas menggambarkan fase pembentukan jadwal dari menambahkan order yang ingin dibuatkan jadwal produksinya untuk masuk daftar jadwal. Kemudian menggapus seluruh jadwal produksi selama 1 tahun (rencana 1 tahun penuh kedepan) pada *cutting*, *sewing*, *washing* serta keseluruhannya. Kemudian membuat kembali jadwal produksi pada *cutting*, *sewing*, *washing* serta keseluruhannya dengan acuan daftar jadwal yang baru ditambah order baru.



Gambar 3.3 Flowchart Pembentukan Jadwal pada *sewing* dalam Aplikasi Optimalisasi Penjadwalan

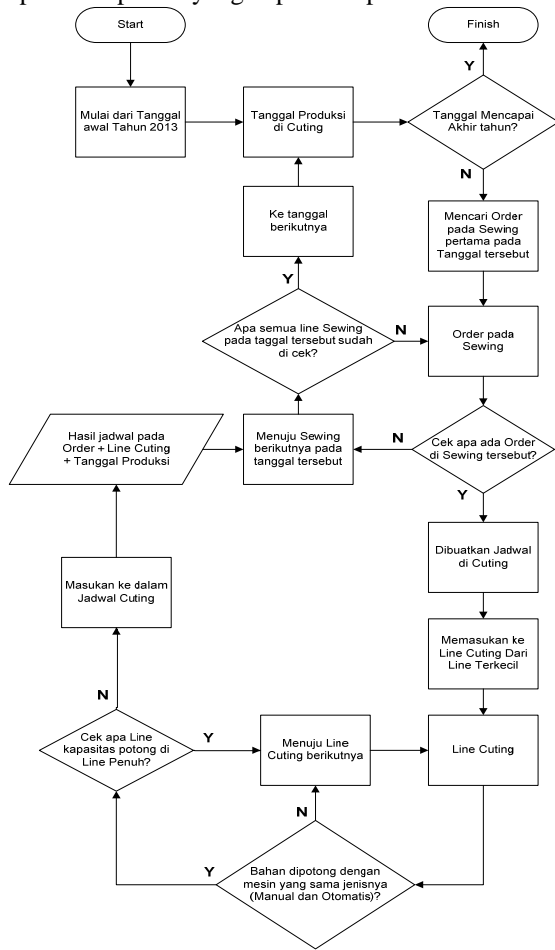
Penjelasan secara lebih luas pada saat pembentukan jadwal baru, terutama pada penjadwalan *sewing*. Karena produksi pada *sewing* menjadi acuan utama untuk membuat berbagai jadwal produksi di bagian yang lainnya. Pembentukan Jadwal tersebut dapat dijabarkan sebagai berikut:

1. Mencari *order* dengan batas tenggang waktu perjanjian dengan *buyer* yang paling pendek dan belum dibuat jadwal untuk *sewing* nya.
2. Memasukan *order* ke dalam jadwal *sewing*.
3. Mencari line paling pertama untuk dibuatkan jadwal untuk line yang belum memiliki jadwal produksi.
4. Memasukan *order* ke dalam jadwal line.
5. Mengecek tanggal apa saja line memiliki kekosongan jadwal selama tenggang waktu produksi *order*.
6. Memasukan *order* ke dalam jadwal line pada tanggal - tanggal tersebut jika ada.
7. Mengecek apa sudah jumlah *quantity order* sudah terpenuhi.
8. Jika belum maka mencari line berikutnya lalu melakukan kegiatan dari nomer 4 dan seterusnya

atau jika sudah terpenuhi maka mencari *order* berikutnya jika masih ada yang belum dibuatkan jadwal lalu melakukan kegiatan dari nomer 2 dan seterusnya.

9. Kegiatan akan berakhir jika semua *order* sudah di buatkan jadwal atau *sewing* sudah tidak dapat menampung produksi *order* berikutnya.

Maka dari aturan pencarian dengan metode Hill Climbing akan tercipta suatu jadwal dengan cepat dan optimal yang dapat di capai.



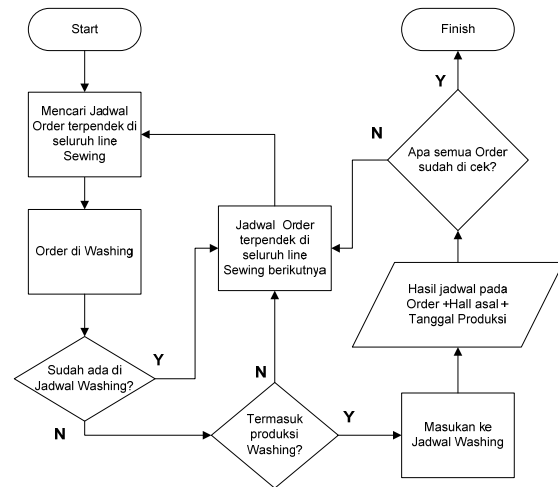
Gambar 3.4 Flowchart Pembentukan Jadwal pada *cutting* dalam Aplikasi Optimalisasi Penjadwalan

Penjelasan secara lebih luas pada saat pembentukan jadwal baru, terutama pada penjadwalan *cutting* setelah membentuk penjadwalan *sewing*. Karena produksi pada *sewing* akan menjadi acuan utama untuk membuat jadwal produksi di bagian *cutting*. Pembentukan Jadwal tersebut dapat dijabarkan sebagai berikut:

1. Mengecek tanggal dari awal tahun.
2. Dari tanggal tersebut untuk mengecek line *sewing* paling pertama pada *sewing*.

3. Mengecek *order* pada line dan tanggal tersebut.
4. Memasukan *order* ke dalam jadwal *cutting*.
5. Mencari line atau mesin *cutting* paling pertama untuk dicek.
6. Apakah line atau mesin *cutting* cocok dengan type mesin pada *motive* yang akan dibuat.
7. Jika tidak maka mencari line atau mesin *cutting* berikutnya lalu melakukan kegiatan dari nomer 6 kembali dan seterusnya.
8. Apakah mesin tersebut masih mampu membuat sejumlah besar *order* yang akan di produksi di line *sewing* tersebut.
9. Jika mampu maka masukan ke jadwal line atau mesin *cutting* tersebut dan jika tidak mampu maka cari line atau mesin *cutting* selanjutnya atau melakukan kegiatan nomer 6 dan seterusnya.
10. Apakah semua line *sewing* sudah di cek *order* produksinya.
11. Jika sudah maka mengecek tanggal selanjutnya dan lakukan kegiatan nomer 2 dan seterusnya atau jika belum maka mengecek line *sewing* selanjutnya pada *sewing* kemudian melakukan kegiatan nomer 3 dan seterusnya.
12. Kegiatan akan berakhir jika semua *order* di semua line *sewing* dari semua tanggal sudah di buatkan jadwal di *cutting* atau line atau mesin pada *cutting* sudah tidak dapat menampung produksi *order* berikutnya.

Maka dari aturan pencarian dengan metode Hill Climbing akan tercipta suatu jadwal dengan cepat dan optimal yang dapat di capai.



Gambar 3.5 Flowchart Pembentukan Jadwal pada *cutting* dalam Aplikasi Optimalisasi Penjadwalan

Penjelasan secara lebih luas pada saat pembentukan jadwal baru berikutnya yaitu penjadwalan *washing* setelah membentuk penjadwalan *sewing* dan *cutting*. Karena produksi

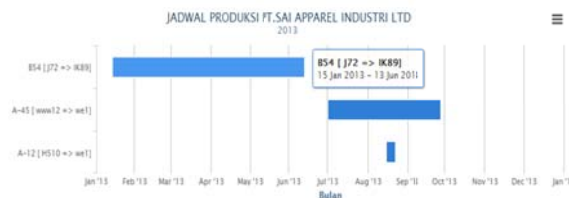
pada *sewing* akan menjadi acuan utama untuk membuat jadwal produksi di bagian *washing*. Pembentukan Jadwal tersebut dapat dijabarkan sebagai berikut:

1. Mencari *order* dengan batas tenggang waktu produksinya pada *sewing* yang paling pendek dan belum dibuat jadwal untuk *washing* nya pada .
2. Mengecek *order* ke dalam jadwal *washing*.
3. Apa *order* sudah ada di jadwal *washing*.
4. Jika sudah maka mengecek *order* dengan batas tenggang waktu produksinya pada *sewing* berikutnya lalu melakukan kegiatan di nomer 2 dan seterusnya.
5. Apa *order* termasuk produksi yang harus melalui proses *washing*.
6. Jika benar maka masukan *order* ke jadwal *washing* tapi jika tidak maka mengecek *order* dengan batas tenggang waktu produksinya pada *sewing* berikutnya lalu melakukan kegiatan di nomer 2 dan seterusnya.
7. Kegiatan akan berakhir jika semua *order* yang termasuk produksi melalui proses cuci di semua line *sewing* dari semua tanggal sudah di buatkan jadwal di *washing*.

Maka dari aturan pencarian dengan metode Hill Climbing akan tercipta suatu jadwal dengan cepat dan optimal yang dapat di capai.

3.3.3 Perancangan Interface

Pada tahapan ini antarmuka aplikasi akan didesain untuk masing – masing user dan di bawah ini adalah contoh interface dari diagram penjadwalan yang menunjukkan order, buyer dan style yang digunakan.



Gambar 3.6 Rancangan Aplikasi Penjadwalan

4. Kesimpulan

Berdasarkan pembahasan yang telah dipaparkan sebelumnya maka dapat ditarik kesimpulan diantaranya adalah :

1. Prototype yang telah dikembangkan, dapat memperhitungkan dalam memperkirakan antara kemampuan dengan permintaan yang terbentuk dalam jadwal.
2. Prototype yang penulis kembangkan berhasil menghasilkan bentuk jadwal berupa grafik untuk menunjukkan jadwal di setiap tempat produksi.

3. Mampu mengetahui kebutuhan kemampuan / jumlah line yang dibutuhkan di cutting dan *sewing* untuk memenuhi permintaan *order* dengan kata lain dapat mengetahui jumlah line yang harus aktif dalam rentan waktu produksinya.

Daftar Pustaka

- File Jeffrey W. Herrmann , Website education, Home page on-line. Available from http://drum.lib.umd.edu/bitstream/1903/7488/4/25813_cov.pdf; Internet; accessed 1 April 2013.
- File Wikibooks , Website organization, Home page on-line. Available from http://en.wikibooks.org/wiki/Algorithms/Hill_Climbing; Internet; accessed 1 April 2013.
- File Universitas Kristen Petra , Website academic, Home page on-line. Available from <http://puslit.petra.ac.id/files/published/journals/INF/INF020302/INF02030203.pdf>; Internet; accessed 1 April 2013.
- File Romisatriawahono, Website entitas, Home page on-line. Available from <http://romisatriawahono.net/2012/08/07/kiat-menyusun-kerangka-pemikiran-penelitian/Internet> ; accessed 1 April 2013.
- File Institut Teknologi Bandung , Website academic, Home page on-line. Available from <http://mail.stei.itb.ac.id/~soni/EL5133/Materi/03-heuristic%20search.pdf>; Internet; accessed 1 April 2013.
- File Wikipedia , Website organization, Home page on-line. Available from https://en.wikipedia.org/wiki/Hill_climbing; Internet; accessed 1 April 2013.