BAB II

TINJAUAN PUSTAKA

Tinjauan pustaka yang digunakan meliputi faktor-faktor penentu prioritas penjadwalan, penjadwalan yang terjadi diproduksi pakaian, penjadwalan sistem operasi komputer dan penelitian sebelumnya yang berhubungan dengan penjadwalan sistem operasi.

Dalam industri pakaian dinamika perubahan pasar mempunyai pengaruh terbesar dalam menentukan produk apa yang harus diproduksi dengan tepat, hal ini menuntut perusahaan untuk menghasilkan produk yang tepat dan jumlah yang pas. Pengaturan produk yang akan diproduksi dan waktu yang tepat kapan produksi produk harus selesai merupakan hal yang penting sehingga penjadwalan produksi mempunyai peranan penting dalam hal ini. Dalam perencanaan penjadwalan produksi pembuatan produk akan disesuaikan dengan sejarah penjualan sebelumnya ataupun perkiraan jumlah penjualan yang harus dipenuhi pada waktu tertentu.

Meskipun jadwal produksi tiap bulan rutin dibuat namun, ketika ada perubahan pasar seperti suatu model produk laku keras atau dalam rangka persaingan bisnis, yang mengharuskan bagian produksi memproduksi selain yang dijadwalkan, maka bagian produksi harus menjadwalkan ulang jadwal produksi yang telah disepakati supaya produk yang akan dibuat dapat dibuat tepat waktu. Proses penjadwalan ulang terkadang berpotensi menguntungkan ataupun

merugikan, jika tidak diperhatikan dengan hati-hati maka akan berpotensi merugikan. Untuk memaksimalkan potensi keuntungan maka perlu diperhatikan faktor-faktor penentu prioritas produksi. faktor- faktor itu dijelaskan sebagai berikut:

2.1 Faktor-Faktor Penentu prioritas Produksi

Biaya didefinisikan sebagai produksi dan distribusi dengan biaya rendah [7]. Bersaing pada biaya berfokus pada kemampuan untuk secara efektif mengelola biaya produksi, termasuk aspek terkait seperti *overhead*, persediaan dan nilai tambah [8], lebih lanjut biaya juga bisa digambarkan sebagai kemampuan untuk mengurangi biaya produk dengan mengurangi biaya *overhead*, tenaga kerja, biaya bahan baku dan waktu siklus produksi.

Kualitas adalah salah satu cara untuk menentukan sejauh mana perusahaan manufaktur mampu menawarkan kualitas produk yang akan memenuhi harapan pelanggan [9]. Ada 8 dimensi kualitas, yaitu: kinerja, fitur, keandalan, kesesuaian, daya tahan, kemudahan servis, estetika, dan kualitas yang dirasakan [10]. Kualitas adalah hal yang harus dipenuhi perusahaan jika ingin menang dalam persaingan bisnis.

Fleksibilitas adalah kemampuan untuk bereaksi terhadap perubahan dalam produksi, perubahan dalam campuran produk, modifikasi dalam desain, fluktuasi bahan, dan perubahan secara berurutan [7]. Sedang ukuran lain dari fleksibilitas yang digunakan termasuk pengurangan *lead time*, pengaturan pengurangan waktu

dan kemampuan untuk mengubah prioritas pekerjaan di penjadwalan produksi metode *flow shop* [11].

Pengiriman biasanya mengacu pada kedua kecepatan dan ketergantungan, ketergantungan pengiriman sebagai jadwal pengiriman pertemuan dan kecepatan pengiriman sebagai bereaksi cepat terhadap pesanan pelanggan [12]. Pengiriman juga dapat menyertakan aspek purna jual sebagai ukuran pengiriman [13].

Inovasi didefinisikan sebagai pengenalan produk dan proses baru [7], tetapi juga berbicara kepada kemampuan perusahaan untuk melakukan perbaikan pada produk dan proses yang ada. Selain itu, mengacu pada kemampuan untuk menerapkan teknologi baru dan kemampuan untuk menciptakan pasar baru [14]. Inovasi juga termasuk adopsi teknologi baru, pengurangan waktu pengembangan produk dan pengenalan produk baru [13].

Pelayanan adalah bagaimana memberikan layanan terbaik bagi pelanggan atau pembahasan mengenai kepuasan pelanggan [15]. Layanan pelanggan di bagian utama dari manufaktur adalah satu bentuk ketergantungan [16].

Lama produksi adalah perkiraan waktu yang dibutuhkan bagian produksi untuk menghasilkan sejumlah barang, tujuannya adalah agar semua produk yang dipesan oleh pembeli dapat diatur dan dijadwalkan sedemikian rupa sehingga semua pesanan dapat dilayani dengan baik dan adil [4].

Lama produksi suatu model pakaian terkait dengan tingkat kerumitan pembuatan model pakaian tersebut. Untuk mengetahui lama produksi biasanya tim pembuat contoh pakaian (tim *sample*) akan mencoba membuat contoh model

pakaian yang akan di produksi, lalu dihitung perkiraan membuat produk tersebut, baik dari lama waktu penyiapan kain yang akan dipotong, pengumpulan bahan pendukungnya atau biasa di sebut aksesoris pakaian, proses menjahit pakaian, proses pengecekan pakaian setelah di jahit atau yang disebut kontrol kualitas. Dalam proses menjahit pakaian pun ada jenis- jenis kain yang lebih lama ketika proses jahitnya biasanya tergantung jenis kainnya, jika jenis kainnya termasuk jenis yang licin maka proses jahitnya pun akan lebih lama dibandingkan kain jenis yang tidak licin. Setelah proses perhitungan tersebut selesai maka bagian sample akan melaporkan ke bagian yang terkait lama produksi ($\boldsymbol{T_p}$) produk yang dibuat dan mendokumentasikannya.

Tenggat waktu merupakan waktu maksimal yang dapat diterima untuk menyelesaikan pekerjaan tersebut, kelebihan waktu yang telah ditentukan disebut keterlambatan [17]. Tenggat waktu penyelesaian produksi berkaitan dengan jadwal suatu model produk harus sudah dapat mulai dijual di toko, hal ini penting sekali jika dihubungkan dengan fokus pemenuhan produksi sehingga diperlukan pengaturan waktu yang efektif. Keterlambatan pemenuhan waktu produksi ini menyebabkan model tersebut terlambat sampai di toko, yang berpotensi menyebabkan model tersebut tidak laku terjual. Hal ini dapat disebabkan karena pelanggan menjadi lebih tertarik membeli pakaian dengan model serupa yang sudah siap di toko dari perusahaan pesaing.

Karena model pakaian itu dipengaruhi oleh tren model yang ada di suatu daerah atau pun berhubungan dengan hari-hari khusus seperti hari raya ataupun kegiatan sekolah maka penyediaan barang di toko perlu dilakukan tepat waktu

sehingga potensi produk terjual akan tinggi. Potensi kerugian perusahaan juga dapat terjadi apabila saat penyelesaian produksi suatu model pakaian terlalu cepat, jauh dari tenggat waktu yang ditentukan oleh bagian penjualan. Hal ini diakibatkan karena biaya pergudangan yang diperlukan untuk menyimpan model pakaian tersebut membengkak, ketersediaan tempat untuk penyimpanan produk yang lain menjadi berkurang dan juga akan menyebabkan *cash-flow* terganggu jikalau ada produk yang tidak langsung dijual setelah diproduksi, Semakin dekat tenggat waktu yang tersisa untuk suatu model (T_d), maka semakin tinggi prioritas produksi model tersebut.

Keuntungan adalah selisih yang diperoleh dari pendapatan dikurangi semua biaya [18], keuntungan atau laba merupakan salah satu tujuan utama didirikannya perusahaan, keuntungan yang diperoleh akan dipergunakan dalam memperluas kegiatan usahanya. Semakin besar keuntungan yang diperoleh perusahaan, semakin besar pula peluang perluasan usahanya.

Potensial adalah sebuah istilah dari kata potensi yang berarti memiliki kemampuan namun belum tercapai untuk saat ini, secara sederhana kata potensial bisa diartikan dengan memiliki potensi. Potensi adalah sesuatu yang dapat kita kembangkan [19]. **Keuntungan potensial produksi** adalah keuntungan produk pada saat produk tersebut diproduksi tanpa menghitung berapa banyak produk yang akan laku di pasaran.

Strategi *marketing* adalah suatu nilai yang ditentukan oleh bagian penjualan sebagai hasil dari pengamatan pola pembelian pelanggan maupun pola produksi model pakaian dari perusahaan pesaing.

Dalam pembahasan strategi *marketing* akan dibagi menjadi 2 kategori yaitu: make to order (MTS), make to stock (MTS).

MTO adalah Sebuah strategi bisnis produksi yang biasanya memungkinkan konsumen untuk membeli produk yang disesuaikan dengan spesifikasi mereka, bagian produksi akan memproduksi sesuai dengan pesanan pelanggan. Konsep pembuatan produk berdasarkan pesanan pelanggan dikenal dengan istilah *make to order* (MTO). Strategi dengan konsep ini hanya memproduksi produk akhir setelah pelanggan menempatkan pesanan. Hal ini menciptakan proses menunggu waktu tambahan bagi konsumen untuk menerima produk, tetapi memungkinkan untuk kustomisasi lebih fleksibel dibandingkan dengan membeli dari rak pengecer ataupun dari toko.

Konsep MTO juga merupakan strategi dapat mengurangi masalah persediaan yang berlebihan yang umum dengan konsep produksi tradisional untuk pembuatan produk berdasarkan persediaan dikenal dengan istilah *make to stock* (MTS) strategi. Komputer Dell, konsep toko butik adalah contoh bisnis yang menggunakan strategi produksi MTO.

Strategi MTO mempunyai konsep persediaan tetapi hanya dalam bentuk desain produk dan beberapa bahan baku standar, sesuai dengan produk yang telah dibuat sebelumnya. Aktivitas proses berdasarkan pesanan konsumen, yang berarti konsep produksi ini menitikberatkan terhadap pesanan konsumen. Aktivitas proses

dimulai pada saat konsumen menyerahkan spesifikasi produk yang dibutuhkan dan perusahaan akan membantu konsumen menyiapkan spesifikasi produk, beserta harga dan waktu penyerahan. Apabila telah dicapai kesepakatan, maka perusahaan akan mulai membuat komponen dan merakitnya menjadi produk dan kemudian menyerahkan kepada konsumen. Pada strategi ini, resiko terhadap investasi persediaan kecil, operasionalnya lebih fokus pada keinginan konsumennya. Contoh produk: komponen mesin, komputer untuk riset, dan lain-lain.

MTS adalah salah satu starategi bisnis dalam produksi yang mempunyai tujuan konsep pembuatan produknya untuk disimpan di gudang, ada proses jeda jika produk selesai dibuat maka akan disimpan terlebih dahulu di gudang

Adapun karakteristik MTS:

- Adanya proses penyimpanan produk hasil produksi di gudang.
- Tingkat persediaan produk di gudang tergantung pada waktu respon permintaan pelanggan dan tingkat analisa kebutuhan pelanggan oleh *marketing*.
- Lebih mudah mengatur jadwal produksi berdasarkan perkiraan permintaan.
- Jikalau permintaan barang tinggi maka tingkat persediaan barang lebih sedikit, penanganan permintaan lebih cepat karena barang telah tersedia sebelumnya.
- Diperlukan modal untuk memproduksi produk tersebut yang menjamin perkiraan permintaan konsumen terpenuhi.

- Jenis pelanggan MTS ini tidak bersedia menunggu lama untuk mendapatkan produk yang dipesannya.
- Adanya kemungkinan barang hasil produksi menumpuk di gudang karena barang tersebut tidak laku dijual di pasaran.
- Bagian penjualan harus menjual produk yang sudah ada di gudang.

Dari karakteristik produksi, baik berkaitan dengan strategi produksi yang berhubungan dengan permintaan konsumen dan permintaan untuk pemenuhan barang untuk stok maka dirumuskan bahwa strategi *marketing* berkaitan erat dengan :

- Inovasi.
- Pesanan konsumen yang harus segera dipenuhi.
- Pemenuhan terhadap prediksi bagian *marketing* terhadap *event* tertentu seperti hari raya, ataupun musim awal anak sekolah.
- Mengamankan jumlah stok untuk strategi produksi *make to stock.*
- Persaingan dengan pesaing bisnis.

Dari poin-poin di atas bagian *marketing* akan mengurutkan sesuai kepentingannya, Jika suatu model pakaian dianggap perlu untuk segera dibuat, maka nilai strategi *marketing* (**m**) akan dibuat tinggi oleh bagian penjualan, demikian pula sebaliknya.

Setelah faktor-faktor telah ditelaah maka selanjutnya akan dirumuskan berdasarkan faktor-faktor yang telah disesuaikan berdasarkan karakteristik perusahaan untuk selanjutnya disusun penjadwalan produksi.

2.2 Penjadwalan Produksi

Penjadwalan dapat diartikan sebagai proses pengalokasian sumber daya yang terbatas untuk memilih tugas dalam jangka waktu tertentu. Definisi umum ini memiliki dua arti. Pertama, penjadwalan merupakan fungsi dari pembuat keputusan yang merupakan proses untuk menetapkan suatu jadwal sehingga dapat diterapkan pada proses pembuatan keputusan yang lain. Kedua, Penjadwalan sebagai suatu teori yang terdiri dari pengumpulan dasar, model, teknik dan konklusi logis yang memberikan gambaran dalam fungsi penjadwalan.

Penjadwalan dapat juga merupakan suatu usaha menggunakan waktu agar diperoleh efisiensi terhadap waktu yang sesuai dengan kebutuhan. Permasalahan dalam penjadwalan adalah bagaimana mengalokasikan pekerjaan ke mesin, pada kondisi mesin mempunyai kapasitas dan jumlah yang terbatas [20].

Penjadwalan juga dapat didefinisikan menjadi dua hal, yakni: meresepkan kapan dan di mana masing-masing operasi manufaktur dari suatu produk perlu untuk dilaksanakan. Menentukan waktu yang digunakan untuk memulai dan/atau melengkapi masing-masing peristiwa atau operasi yang berisikan suatu prosedur [21].

Penjadwalan dalam suatu organisasi membahas cara untuk menetapkan pengaturan waktu pemakaian sumber daya yang spesifik dalam organisasi tersebut [22]. Hal ini berhubungan dengan pemakaian peralatan, fasilitas-fasilitas, dan aktivitas manusia. Tujuan dari penjadwalan, yaitu:

- Memanfaatkan seefisien mungkin staf, peralatan dan fasilitas-fasilitas.

- Meminimalkan waktu menunggu konsumen, inventaris dan proses waktu.

Dalam penjadwalan atau *scheduling* ada beberapa tujuan yang harus dicapai [23] seperti:

- Efisiensi yang tinggi.
- Persediaan atau inventori sedikit.
- Kepuasan pelanggan.

Ketika membahas penjadwalan produksi, maka tidak terlepaskan dengan pembahasan mengenai kapasitas produksi. Kapasitas merupakan jumlah keluaran yang dapat dihasilkan oleh suatu sistem produksi dalam cakrawala waktu tertentu, yaitu selama satu tahun atau dalam beberapa tahun mendatang. Kapasitas dapat pula diartikan sebagai jumlah unit produk yang dapat ditangani, diterima, disimpan, atau diakomodasi dalam waktu tertentu [24]. Pengertian yang pertama terpakai dalam menyatakan kemampuan menghasilkan produk dari sebuah pabrik.

Sedangkan makna kapasitas yang kedua, lazim dipakai untuk menyatakan kemampuan sebuah fasilitas jasa. Kapasitas teoritis atau maksimum atau desain (*theoritical/maximum/design capacity*) adalah kapasitas maksimal yang dimungkinkan dari sistem produktif [25].

Dalam menentukan prioritas penjadwalan ada beberapa teknik penjadwalan yang biasa digunakan [26] yaitu:

*2.2.1 First Come First Served (*FCFS*)*

Pengerjaan pesanan berdasarkan urutan kedatangan yaitu pertama datang, pertama dilayani. Pekerjaan yang datang dahulu ke pusat kerja akan diproses terlebih dahulu Kelebihan. Metode ini adalah yang paling sederhana, dan menjamin bahwa permintaan yang datang terlebih dahulu akan terlayani, sehingga metode ini terlihat adil untuk pelanggan suatu hal yang dianggap penting dalam rangka kepuasan pelanggan. kelemahannya jika terjadi *convoy effect,* jika ada sebuah proses yang kecil tetapi mengantri dengan proses yang membutuhkan waktu yang lama maka mengakibatkan proses tersebut tidak akan untuk dikerjakan.

Kelemahan ini akan menyebabkan pesanan yang baru datang dan berpotensi berkeuntungan tinggi tidak dikerjakan sehingga menyebabkan perusahaan mengalami kerugian.

*2.2.2 Early Due Date (*EDD*)*

Merupakan pengurutan pengerjaan pesanan berdasarkan batas waktu yang paling awal, yaitu pekerjaan dengan batas waktu yang paling awal akan dikerjakan terlebih dahulu. Kelebihannya meminimalkan waktu keterlambatan ketika penyerahan produk jadi ke pelanggan. Kelemahan dari metode ini adalah belum tentu produksi suatu produk yang mempunyai waktu penyerahan terpendek merupakan produksi yang dinilai menguntungkan oleh perusahaan.

*2.2.3 Shortest Processing Time (*SPT*)*

Metode pengerjaan pesanan berdasarkan waktu pemrosesan terpendek. Pekerjaan yang memiliki waktu proses tercepat atau terpendek diselesaikan terlebih dahulu. Kelebihannya metode ini paling optimal, karena metode ini memberikan waktu tunggu pengerjaan yang kecil untuk kumpulan proses yang mengantri. Kelemahannya tidak dapat digunakan untuk penjadwalan dengan tipe penjadwalan pendek.

*2.2.4 Longest Processing Time (*LPT*)*

Merupakan proses pengurutan pesanan berdasarkan waktu pemrosesan terpanjang, yaitu pekerjaan dengan waktu proses yang panjang akan diutamakan dan didahulukan. Memiliki kelemahan karena berpotensi akan mengakibatkan produk yang mempunyai tenggat waktu terpendek akan terabaikan dan menyebabkan keterlambatan produksi, yang dapat menimbulkan keluhan dari pelanggan.

2.2.5 Rush Order Based Scheduling

Merupakan penjadwalan yang terjadi diakibatkan oleh adanya permintaan suatu produk mendadak yang harus segera diselesaikan [27], baik itu permintaan oleh bagian *marketing* ataupun pihak yang berpengaruh dalam manajemen perusahaan. Penggunaan metode ini biasanya tergantung dari kebijakan manajemen, biasanya akan menggantikan proses produksi yang mempunyai tingkat kepentingan dianggap yang paling rendah dari proses lainnya. Kelemahan dari metode ini adalah tidak memperhitungkan nilai keuntungan dari produksi produk tersebut.

2.3 Penjadwalan Ulang Produksi

Penjadwalan ulang produksi adalah praktik umum di perusahaan manufaktur di seluruh dunia [28]. Dalam pembahasan yang berhubungan Dalam bidang pemasaran saat ini, pesanan terburu-buru (*rush order*) sering muncul dalam sistem manufaktur dan harus dijadwal ulang pada waktunya sehingga mereka akan selesai secepat mungkin. Selain itu, beberapa pekerjaan asli, yang belum diproses dapat diproses dengan kriteria yang baik [29]. Untuk membantu menyelesaikan masalah ini diperlukan penggunaan metode penjadwalan yang tepat.

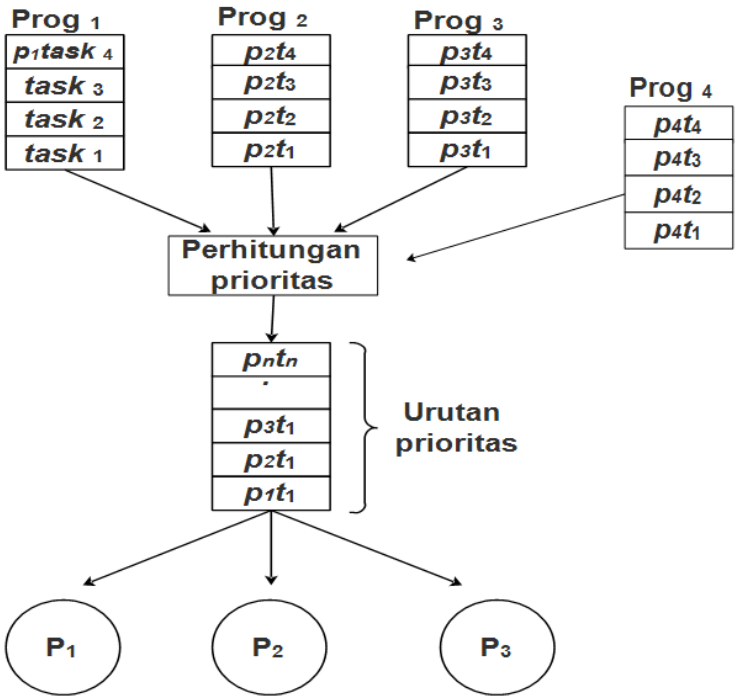
Dilihat dari karakteristik penjadwalan produksi yang mengerjakan produk di lini produksi dalam waktu tertentu dan adanya gangguan penjadwalan oleh pesanan produk baru ternyata hal-hal tersebut terjadi juga di penjadwalan *task* sistem operasi komputer.

Jika di dalam penjadwalan produksi pakaian pengerjaan produk dilakukan di lini produksi tertentu, maka analogi di sistem operasi pengerjaan *task* dilakukan di suatu prosesor. Jika di dalam penjadwalan produksi adanya gangguan penjadwalan dikarenakan ada produk baru yang harus segera dikerjakan, maka analogi di sistem operasi adalah ketika adanya gangguan jadwal yang disebabkan ada *task* yang mempunyai prioritas tinggi yang perlu dijalankan oleh prosesor. Analogi-analogi tersebut menyebabkan sehingga permasalahan pemaksimalan keuntungan potensial produksi yang terjadi penjadwalan di produksi dapat diselesaikan dengan menggunakan pendekatan penjadwalan *task* di sistem operasi. Sejumlah teknik penjadwalan *task* di sistem operasi dapat dipilih untuk menentukan

strategi penjadwalan produksi pakaian yang dapat memaksimalkan keuntungan potensial produksi.

2.4 Penjadwalan *Task* Sistem Operasi

Secara umum proses penjadwalan di sistem operasi



Gambar II-1. Proses Penjadwalan Sistem Operasi

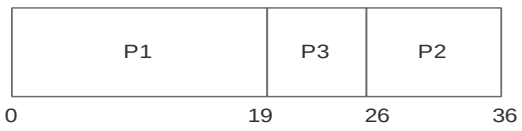
Gambar II-1 menunjukan proses penjadwalan *task* di sistem operasi, setiap *task* yang merupakan bagian terkecil dari program akan dihitung nilai prioritasnya, selanjutnya akan diurutkan sesuai nilai prioritasnya yang selanjutnya *task* tersebut akan di proses oleh prosesor yang sedang kosong ataupun *task scheduler* akan menggantikan *task* yang sedang berjalan di suatu prosesor dengan *task* yang baru dikarenakan prioritas yang baru mempunyai tingkat prioritas yang lebih tinggi.

Proses penjadwalan di sistem operasi jika dilihat dari sisi penentuan prioritasnya terbagi menjadi 2 [30] yaitu:

1. *Nonpreemptive* artinya sekali *task* berada dalam status berjalan, maka *task* ini akan terus dijalankan sampai selesai atau blok oleh dirinya sendiri atau I/O atau permintaan dari *service* sistem Operasi.
2. *Preemptive* artinya *task* yang sedang berjalan dapat diganggu dan dipindahkan ke status persiapan oleh sistem operasi. keputusan untuk *preemptive* mungkin dijalankan ketika *task* baru datang.

2.4.1 Penjadwalan *Nonpreemptive*

Konsep penjadwalan yang setelah prosesor diberikan kepada suatu *task* untuk mengerjakan *task* tersebut, maka *task* tersebut tidak boleh diganggu sampai *task* tersebut selesai dikerjakan oleh prosesor tersebut



Gambar II-2. Penjadwalan Nonpreemptive

Gambar II-2 menunjukan proses penjadwalan *nonpreemptive* dengan rincian waktu pengerjaan prosesor P1, P2, dan P3 adalah 19, 10, dan 7 masing-masing. P1, P2, dan P3 hanya memiliki satu prosesor. Waktu kedatangan *task*: P1 (0), P2 (2), P3 (4), dan prioritasnya masing-masing adalah 3 (terendah), 2, dan 1 (tertinggi).

*2.4.1.1 First Come First Served (*FCFS*)*

First come first served (FCFS) atau sering dikenal juga dengan nama *first in first out* (FIFO) merupakan penjadwalan tidak berprioritas. FCFS adalah penjadwalan paling sederhana, yaitu *task-task* diberi jatah waktu pemroses berdasarkan waktu kedatangan. Pada saat *task* mendapat jatah waktu pemroses, *task* dijalankan sampai selesai.

Penilaian penjadwalan ini berdasarkan kriteria optimasi :

1. Adil, dalam arti resmi (*task* yang datang duluan akan dilayani lebih dulu), tapi dinyatakan tidak adil karena *task-task* yang perlu waktu lama membuat *task-task* pendek menunggu. *Task-task* yang tidak penting dapat membuat *task-task* penting menunggu lama.
2. Efisiensi, sangat efisien.
3. Waktu tanggap sangat jelek, tidak cocok untuk sistem interaktif apalagi untuk sistem waktu nyata.
4. *Turn around time* kurang baik.
5. *Throughtput* kurang baik, FCFS jarang digunakan secara mandiri, tetapi dikombinasikan dengan skema lain.
6. Baik untuk sistem *batch* yang sangat jarang berinteraksi dengan pemakai. Contoh : aplikasi analisis numerik, maupun pembuatan tabel.
7. Sangat tidak baik (tidak berguna) untuk sistem interaktif, karena tidak memberi waktu tanggap yang baik.
8. Tidak dapat digunakan untuk sistem waktu nyata (*real-time applications*).

Kelemahannya:

1. *Waiting time* rata-ratanya cukup lama.
2. Terjadinya *convoy effect*, yaitu *task-task* menunggu lama untuk menunggu 1 *task* besar yang sedang dieksekusi oleh prosesor. Metode ini juga menerapkan konsep *nonpreemptive*, yaitu setiap *task* yang sedang dieksekusi oleh prosesor tidak dapat di-*interrupt* oleh *task* yang lain.

2.4.1.2 *Shortest Job First* (SJF)

Penjadwalan ini mengasumsikan waktu berjalannya *task* sampai selesai telah diketahui sebelumnya. Mekanismenya adalah menjadwalkan *task* dengan waktu jalan terpendek lebih dulu sampai selesai, sehingga memberikan efisiensi yang tinggi dan *turn around time* rendah dan penjadwalannya tak berprioritas [6].

Karena SJF selalu memperhatikan rata-rata waktu respon terkecil, maka sangat baik untuk proses interaktif. Umumnya *task* interaktif memiliki pola, yaitu menunggu perintah, menjalankan perintah, menunggu perintah dan menjalankan perintah, begitu seterusnya. Masalah yang muncul adalah tidak mengetahui ukuran pekerjaan saat pekerjaan tersebut masuk. Untuk mengetahui ukuran pekerjaan adalah dengan membuat estimasi berdasarkan kelakuan sebelumnya. Prosesnya tidak datang bersamaan, sehingga penetapannya harus dinamis. Penjadwalan ini jarang digunakan karena merupakan kajian teoritis untuk pembandingan *turn around time*.

2.4.1.3 *Highest Response Ratio Next* (HRRN)

Highest response ratio next merupakan strategi penjadwalan dengan prioritas *task* tidak hanya berdasarkan fungsi waktu layanan tetapi juga jumlah

waktu tunggu *task*. Begitu *task* mendapat jatah pemroses, *task* berjalan sampai selesai. Prioritas dinamis HRRN dihitung berdasarkan rumus:

Prioritas = (waktu tunggu + waktu layanan) / waktu layanan

Karena waktu layanan muncul sebagai pembagi, maka pekerjaan lebih pendek berprioritas lebih baik, karena waktu tunggu sebagai pembilang maka *task* yang telah menunggu lebih lama juga mempunyai kesempatan lebih bagus. Disebut HRRN, karena waktu tunggu ditambah waktu layanan adalah waktu tanggap, yang berarti waktu tanggap tertinggi yang harus dilayani [30].

Strategi penjadwalan dengan prioritas proses tidak hanya merupakan fungsi waktu layanan tetapi juga jumlah waktu tunggu *task*. Begitu *task* mendapat jatah pemroses, *task* berjalan sampai selesai, metode ini adalah pengembangan dari SJF–*nonpreemptive*, merupakan: Penjadwalan berprioritas dinamis, penjadwalan untuk mengoreksi kelemahan SJF .

HRRN adalah strategi penjadwalan dengan prioritas *task* tidak hanya merupakan fungsi waktu layanan tetapi juga jumlah waktu tunggu *task.* Begitu proses mendapat jatah pemroses, proses berjalan sampai selesai. HRRN dibuat untuk memperbaiki kelemahan metode SJF. Disebut HRRN, karena waktu tunggu ditambah waktu layanan adalah waktu tanggap, yang berarti waktu tanggap tertinggi yang harus dilayani. Sifat penjadwalan:

1. *Nonpreemptive* yaitu Jika ada proses prosesor telah dialokasikan untuk sebuah *task* maka tidak dapat diberhentikan sampai *task* tersebut selesai.

2. Prioritas Dinamis, merupakan mekanisme menanggapi perubahan lingkungan sistem operasi. Prioritas awal yang diberikan ke *task* mungkin hanya berumus pendek disesuaikan ke nilai yang lebih tepat sesuai dengan lingkungan.

Kelebihan dari penjadwalan ini adalah:

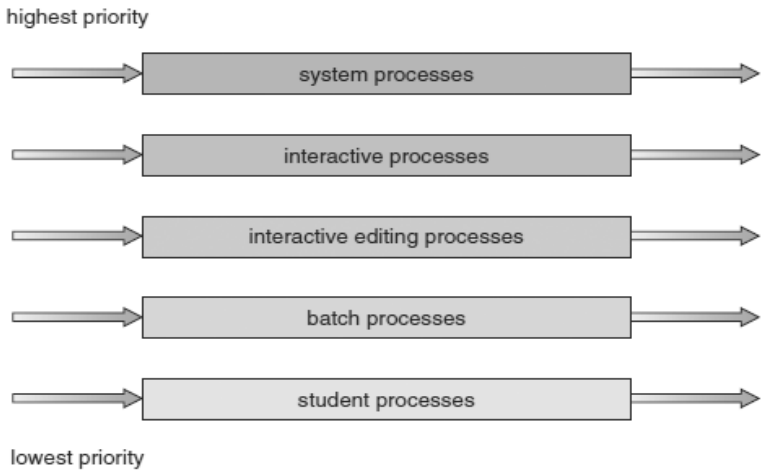
1. Respon time cepat.
2. Setiap *task* akan mendapatkan layanan yang seimbang.
3. Tidak terjadi s*tarvation*, karena *task* yang memiliki prioritas dinamis

Kelemahannya adalah terjadinya *overhead* akibat *scheduler* harus mengetahui atau memperkirakan service time *task- task* yang akan dieksekusi.

2.4.1.4 Multi Level Queue

Metode penjadwalan yang digunakan dalam skenario ini adalah penjadwalan *task* yang melakukan pengklasifikasian menjadi kelompok berdasarkan properti seperti jenis proses, waktu prosesor, akses IO, ukuran memori, dll. Satu klasifikasi umum dari *task* adalah *task* latar depan dan *task* latar belakang. Dalam metode penjadwalan *multi level queue*, akan ada n jumlah antrian, di mana n adalah jumlah kelompok *task* diklasifikasikan ke dalam. Setiap antrian akan diberi prioritas dan akan memiliki metode penjadwalan sendiri seperti penjadwalan *round-robin* atau FCFS. Untuk *task* dalam antrian untuk dieksekusi, semua antrian prioritas lebih tinggi dari itu harus kosong, yang berarti *task* dalam antrian berprioritas tinggi seharusnya telah menyelesaikan eksekusinya. Dalam metode penjadwalan ini, setelah ditetapkan ke antrian, *task* tidak akan berpindah ke antrian lain [6].

Ide dasar dari metode ini berdasarkan pada sistem prioritas *task.* Prinsipnya, jika setiap *task* dapat dikelompokkan berdasarkan prioritasnya. Jika ada 5 *task* yang dengan proses: *system processes, interactive processes, interactive editing processes, batch processes, student processes*, maka akan didapati queue seperti pada gambar berikut:



Gambar II-3 Metode Multi Level Queue

Gambar II-3 menunjukan bahwa setiap antrian memiliki prioritas mutlak atas antrian dengan prioritas lebih rendah. Tidak ada *task* dalam antrian dapat berjalan sehingga yang prioritasnya lebih tinggi telah dikerjakan semuanya.

Metode ini pun memiliki kelemahan, yaitu sangat mungkin bahwa suatu proses pada *queue* dengan prioritas rendah dapat saja tidak mendapat jatah prosesor. Untuk mengatasi hal tersebut, salah satu caranya adalah dengan memodifikasi metode ini dengan adanya jatah waktu maksimal untuk tiap antrian, sehingga jika suatu antrian memakan terlalu banyak waktu, maka prosesnya akan dihentikan, dan

digantikan oleh antrian dibawahnya, dan tentu saja batas waktu untuk tiap antrian dapat saja sangat berbeda tergantung pada prioritas masing-masing antrian.

2.4.1.5 Multi Level Feed Back Queue

Metode penjadwalan ini merupakan pengembangan dari metode *multilevel queue.* Perbedaannya ialah metode ini mengizinkan *task* untuk pindah antrian. Jika suatu *task* menyita prosesor terlalu lama, maka *task* itu akan dipindahkan ke antrian yang lebih rendah. Hal ini menguntungkan proses interaksi karena proses ini hanya memakai waktu prosesor yang sedikit. Demikian pula dengan *task* yang menunggu terlalu lama. *Task* ini akan dinaikkan tingkatannya. Biasanya prioritas tertinggi diberikan kepada *task* dengan prosesor *burst* (waktu pengerjaan proses) terkecil, dengan begitu prosesor akan terutilisasi penuh dan M/K dapat terus sibuk. Semakin rendah tingkatannya, panjang prosesor *burst task* juga semakin besar [6].

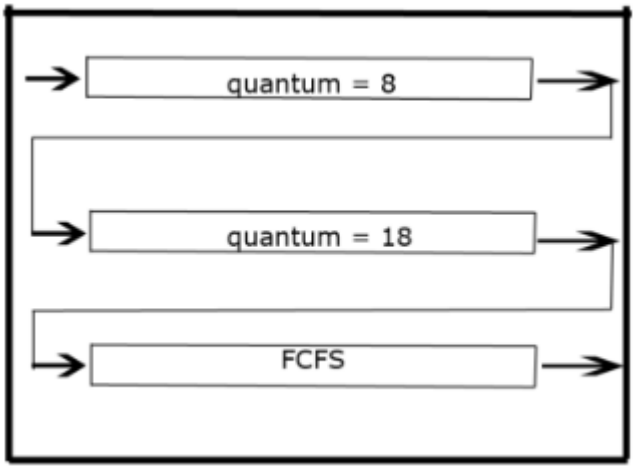
Metode penjadwalan ini didefinisikan melalui beberapa parameter, antara lain:

1. Jumlah antrian.
2. Metode penjadwalan tiap antrian.
3. Kapan menaikkan *task* ke antrian yang lebih tinggi.
4. Kapan menurunkan *task* ke antrian yang lebih rendah.
5. Antrian mana yang akan dimasuki *task* yang membutuhkan.

Dengan pendefinisian seperti tadi membuat algoritma ini sering dipakai, karena metode ini mudah dikonfigurasi ulang supaya cocok dengan sistem.

Penggunaan *multi level feed back queue* misalnya pada contoh berikut :

1. Semua *task* yang baru datang akan diletakkan pada *queue* 0 (*quantum*= 8 ms).
2. Jika suatu *task* tidak dapat diselesaikan dalam 8 ms, maka *task* tersebut akan dihentikan dan dipindahkan ke *queue* 1 (*quantum* = 16 ms).
3. *Queue* 1 hanya akan dikerjakan jika tidak ada lagi *task* di queue 0, dan jika suatu *task* di *queue* 1 tidak selesai dalam 16 ms, maka *task* tersebut akan dipindahkan ke *queue* 2.
4. *Queue* 2 akan dikerjakan bila *queue* 0 dan 1 kosong, dan akan berjalan dengan algoritma FCFS.



Gambar II-4 Metode Multi Level Feed Back Queue

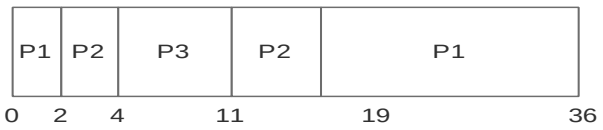
Pada Gambar II-4 terlihat bahwa ada kemungkinan terjadinya perpindahan *task* antar *queue.*

Definisi dari penjadwalan *multi level feed back queue* membuat metode ini paling umum digunakan metode penjadwalan prosesor. Karena metode ini dapat

dikonfigurasi untuk mencocokkan spesifik sistem yang sedang dirancang. Sayangnya, ini juga merupakan metode yang paling rumit, karena mendefinisikan penjadwalan terbaik yang membutuhkan beberapa cara untuk memilih nilai yang cocok untuk semua parameter

2.4.2 Penjadwalan *Preemptive*

Dalam penjadwalan *preemptive* artinya setiap *task* yang dikerjakan dalam prosesor dapat diganggu oleh *task* baru ketika ada *task* yang lebih tinggi prioritasnya, sehingga *task* lama akan diistirahatkan sementara waktu sampai *task* baru selesai dikerjakan ataupun *task* ini menempati prioritas tinggi dalam suatu waktu berdasarkan aturan tertentu yang diberlakukan dalam penjadwalan



Gambar II-5. Penjadwalan Preemptive

Gambar II-5 menunjukan adanya *task* (p) yang terganggu oleh *task* yang lebih tinggi prioritasnya dalam hal ini p1 terganggu dengan kedatangan p2, begitu juga *task* p2 terganggu oleh *task* p3 dan seterusnya. Berikut jenis – jenis metode *Preemptive* sistem operasi:

2.4.2.1 Round-Robin (RR)

Metode penjadwalan *round-robin* (RR) dirancang khusus untuk sistem pembagian waktu. Ini mirip dengan penjadwalan FCFS, tetapi proses *preemptive* ditambahkan untuk mengaktifkan sistem untuk beralih antar *task*. Sebuah satuan

waktu kecil, yang disebut waktu *quantum* atau irisan waktu, didefinisikan. Waktu *quantum* umumnya dari 10 hingga 100 milidetik panjangnya. Antrian siap diperlakukan sebagai antrian melingkar. Penjadwal prosesor berjalan mengelilingi antrean siap, mengalokasikan prosesor ke setiap *task* untuk interval waktu hingga 1 kali *quantum* [6].

Metode ini menggilir *task* yang ada di antrian. *Task* akan mendapat jatah sebesar *time quantum*. Jika *time quantum*-nya habis atau *task* sudah selesai, prosesor akan dialokasikan ke *task* berikutnya. Tentu proses ini cukup adil karena tak ada *task* yang diprioritaskan, semua *task* mendapat jatah waktu yang sama dari prosesor yaitu (1/n), dan tak akan menunggu lebih lama dari (n-1) q dengan q adalah lama 1 *quantum*.

Metode ini sepenuhnya bergantung besarnya waktu *quantum*. Jika terlalu besar, algoritma ini akan sama saja dengan algoritma *first come first served*. Jika terlalu kecil, akan semakin banyak peralihan *task* sehingga banyak waktu terbuang.

Permasalahan utama pada *round-robin* adalah menentukan besarnya *time quantum*. Jika *time quantum* yang ditentukan terlalu kecil, maka sebagian besar *task* tidak akan selesai dalam 1 *quantum*. Hal ini tidak baik karena akan terjadi banyak *switch*, padahal prosesor memerlukan waktu untuk beralih dari suatu *task* ke *task* lain (disebut dengan *context switches time*). Sebaliknya, jika *time quantum* terlalu besar, metode *round-robin* akan berjalan seperti metode *first come first served*. Waktu *quantum* yang ideal adalah jika 80% dari total *task* memiliki prosesor *burst time* yang lebih kecil dari 1 *quantum*.

Kelebihan dari metode ini adalah mempunyai respon lebih cepat terhadap *user* sedangkan kelemahan metode ini yaitu mempunyai waktu lebih lama dalam *turn around* dibandingkan dengan SJF (*Shortest Job First*).

2.4.2.2 Shortest Remaining First (SRF)

Metode s*hortest remaining first* juga dikenal sebagai s*hortest remaining time first* (SRTF), adalah metode penjadwalan yang merupakan versi preemptive dari *shortest job first* (SJF). *Task scheduler* selalu memilih *task* yang sisa waktu pengerjaannya paling pendek. Untuk itu, waktu proses harus diketahui terlebih dahulu. Ketika pekerjaan baru tiba, waktu totalnya dibandingkan dengan waktu *task* saat ini. Jika pekerjaan baru membutuhkan waktu lebih sedikit untuk menyelesaikan daripada *task* saat ini, *task* saat ini ditangguhkan dan pekerjaan baru dimulai. Skema ini memungkinkan pekerjaan baru yang mempunyai waktu pengerjaan singkat untuk mendapatkan layanan yang baik [31]. Karena proses eksekusi saat ini adalah yang dengan jumlah waktu tersingkat yang tersisa berdasarkan definisi, dan sejak saat itu hanya akan berkurang seiring berlangsungnya proses, *task* akan selalu berjalan sampai *task* selesai atau *task* baru ditambahkan yang membutuhkan waktu yang lebih sedikit.

Dengan konsep penjadwalan berdasarkan waktu tersisa terpendek maka hal ini menguntungkan karena *task* singkat ditangani dengan sangat cepat. Sistem ini juga membutuhkan *overhead* yang sangat sedikit karena hanya membuat keputusan ketika *task* selesai atau *task* baru ditambahkan, dan ketika *task* baru ditambahkan, metode hanya perlu membandingkan *task* yang sedang dieksekusi dengan *task* baru,

mengabaikan semua *task* lainnya yang sedang menunggu untuk dieksekusi. Metode ini merupakan:

1. Penjadwalan berprioritas dinamis.
2. *Preemptive* untuk *time sharing.*
3. Melengkapi SJF.

Pada SRF, *task* dengan sisa waktu jalan di estimasi terendah dijalankan, termasuk *task-task* yang baru tiba. Pada SJF, begitu *task* dieksekusi, *task* dijalankan sampai selesai. Pada SRF, *task* yang sedang berjalan (*running*) dapat diambil alih oleh *task* baru dengan sisa waktu jalan yang di estimasi lebih rendah.

Kelemahan:

1. Mempunyai *overhead* lebih besar dibanding SJF. SRF perlu penyimpanan waktu layanan yang telah dihabiskan *task* dan kadang-kadang harus menangani peralihan.
2. Tibanya *task-task* kecil akan segera dijalankan.
3. *Task-task* lebih lama berarti dengan lama dan variasi waktu tunggu lebih lama dibanding pada SJF.

2.4.2.3 *Priority Scheduling* (PS)

Priority scheduling merupakan metode penjadwalan yang mendahulukan *task* yang memiliki prioritas tertinggi. Setiap *task* memiliki prioritasnya masing-masing. Prioritas suatu *task* dapat ditentukan melalui beberapa karakteristik antara lain:

1. *Time limit*.

2. *Memory requirement.*
3. Akses *file*.
4. Perbandingan antara *burst* masukan/ keluaran (M/K) dengan prosesor *burst*.
5. Tingkat kepentingan *task.*

Priority scheduling juga dapat dijalankan secara *preemptive* maupun *nonpreemptive*. Pada *preemptive*, jika ada suatu *task* yang baru datang memiliki prioritas yang lebih tinggi daripada *task* yang sedang dijalankan, maka *task* yang sedang berjalan tersebut dihentikan, lalu prosesor dialihkan untuk *task* yang baru datang tersebut. Sementara itu, pada *task nonpreemptive* yang baru datang tidak dapat mengganggu *task* yang sedang berjalan, tetapi hanya diletakkan di depan *queue*.

Kelemahan pada *priority scheduling* adalah dapat terjadinya *indefinite blocking (starvation)* yaitu adanya *task* yang sedang berjalan di *task* akan digantikan dengan *task* yang baru dan berkesempatan tidak di jalankan karena selalu dikalahkan oleh *task* baru yang mempunyai prioritas lebih tinggi. Suatu *task* dengan prioritas yang rendah memiliki kemungkinan untuk tidak dieksekusi jika terdapat *task* lain yang memiliki prioritas lebih tinggi darinya.

Solusi dari permasalahan ini adalah *aging*, yaitu meningkatkan prioritas dari setiap *task* yang menunggu dalam queue secara bertahap. Contoh: Setiap 10 menit, prioritas dari masing-masing *task* yang menunggu dalam queue dinaikkan satu tingkat. Maka, suatu *task* yang memiliki prioritas 127, setidaknya dalam 21 jam 20

menit, *task* tersebut akan memiliki prioritas 0, yaitu prioritas yang tertinggi (semakin kecil angka menunjukkan bahwa prioritasnya semakin tinggi).

2.4.2.4 *Guaranteed Scheduling* (GS)

Penjadwalan ini memberikan janji yang realistis (memberi daya pemroses yang sama) untuk membuat dan menyesuaikan *performance* adalah jika ada N pemakai, sehingga setiap *task* (pemakai) akan mendapatkan 1/N dari daya pemroses prosesor. Untuk mewujudkannya, sistem harus selalu menyimpan informasi tentang jumlah waktu prosesor untuk semua *task* sejak *login* dan juga berapa lama pemakai sedang login. Kemudian jumlah waktu prosesor, yaitu waktu mulai login dibagi dengan n, sehingga lebih mudah menghitung rasio waktu prosesor. Karena jumlah waktu pemroses tiap pemakai dapat diketahui, maka dapat dihitung rasio antara waktu pemroses yang sesungguhnya harus diperoleh, yaitu 1/N waktu pemroses seluruhnya dan waktu pemroses yang telah diperuntukkan *task* itu. Rasio 0,5 berarti sebuah *task* hanya punya 0,5 dari apa yang waktu prosesor miliki dan rasio 2,0 berarti sebuah *task* hanya punya 2,0 dari apa yang waktu prosesor miliki.

Metode penjadwalan ini akan menjalankan *task* dengan rasio paling rendah hingga naik ke tingkat lebih tinggi di atas pesaing terdekatnya. Ide sederhana ini dapat diimplementasikan ke sistem real-time dan memiliki penjadwalan berprioritas dinamis.

Kelemahan metode ini jika ada pekerjaan yang memiliki keuntungan yang lebih tinggi akan disamakan penggunaan utilitas sumber daya yang ada.

2.5 Penelitian Sebelumnya

Beberapa *paper* telah dipublikasikan terkait permasalahan dalam penentuan prioritas dalam penjadwalan produksi :

1. Pada tahun 2011 Yonatan Nugrahanto melakukan penelitian dengan judul Rancang Bangun Sistem Penjadwalan Produksi Dengan Kombinasi Algoritma *Shortest Job First* Dan *Dynamic Priority Scheduling*, penelitian yang dilakukan adalah dengan mengombinasikan metode *shortest job first* dengan *dynamic priority* yang bertujuan efisiensi penyelesaian pesanan, kurang maksimal jika digunakan terhadap pesanan yang benar-benar baru [4].
2. Tahun 2008 Theresia Lilyanawati Tanjung melakukan penelitian dengan judul Rancang Bangun Sistem Penjadwalan Produksi Dengan Metode *Earliest Due Date* Pada CV. Sumber Artha, penelitian yang dilakukan dengan mengurutkan pesanan produk berdasarkan tenggat waktu penyerahan pesanan, hasilnya berupa penjadwalan yang berkonsentrasi terhadap waktu penyerahan pesanan [5].
3. Tahun 1991 Thomas J Crowe dan Pablo Nuno melakukan penelitian dengan judul *Deciding Manufacturing Priorities: Flexibility, Cost, Quality, Service,* penelitian ini menggali faktor yang berpengaruh dalam proses manufaktur [2].
4. Tahun 2014 Suzana N. Russell dan Harvey H. Millar melakukan penelitian dengan judul *Competitive Priorities of Manufacturing Firms in The Caribbean*, penelitian ini menggali faktor yang berpengaruh dalam penentuan prioritas dalam proses manufaktur di negara Karibia, hasilnya berupa daftar

faktor yang dapat menjadi pertimbangan dalam penentuan prioritas produksi [3].

Jika disusun ke dalam tabel maka akan menjadi sebagai berikut:

Tabel II-1 Rangkuman Penelitian sebelumnya

Judul, Peneliti	**Algoritma**	**Aspek-Aspek Prioritas**	**Fokus Waktu**	**Fokus Keuntungan**
Deciding Manufacturing Priorities (1991) ,Thomas J Crowe dan Pablo Nuno	Tidak ada	Biaya, kualitas, fleksibilitas, pelayanan	Ya	Ya
Rancang Bangun Sistem Penjadwalan Produksi Dengan Metode *Earliest Due Date* Pada CV. Sumber Artha (2008),Theresia Lilyanawati Tanjung	EDD	Waktu penyerahan produk	Ya	Tidak
Rancang Bangun Sistem Penjadwalan Produksi Dengan Kombinasi Algoritma *Shortest Job First* Dan *Dynamic Priority Scheduling* (2011),Yonatan Nugrahanto	SJF + DPS	Waktu penyerahan produk	Ya	Tidak
Competitive Priorities of Manufacturing Firms in The Caribbean (2014), Suzana N. Russell dan Harvey H. Millar	Tidak ada	*Biaya, kualitas, fleksibilitas, pengiriman, inovasi*	Ya	Ya

Berdasarkan table II-1, penelitian ini akan mencangkup penentuan faktor-faktor yang berpengaruh terhadap penjadwalan produksi pakaian sesuai dengan karakteristik perusahaan, pembuatan formula untuk penentuan nilai prioritas produksi dan penyusunan algoritma penjadwalan pakaian untuk memaksimalkan keuntungan potensial produksi.