

# Rancangan Sistem Informasi Penjadwalan Produksi pada Sistem *Shop Floor Control* \*

Nafiurridha, Rispianda, Cahyadi Nugraha

Jurusan Teknik Industri, Institut Teknologi Nasional (Itenas), Bandung

Email: nafiurridha@gmail.com

## ABSTRAK

*Penjadwalan adalah suatu proses pengambilan keputusan. Penjadwalan yang baik dapat meningkatkan efektivitas dan efisiensi produksi. Namun, seringkali pada aktualisasinya tidak semua jadwal dilaksanakan secara tepat, sehingga menimbulkan gap antara aktualisasi dengan rencana. Gap tersebut membutuhkan penanganan yang serius dan segera agar tidak menimbulkan kesalahan yang berkelanjutan. Oleh karena itu, pada penelitian ini dirancang suatu sistem informasi yang dapat menerima jadwal aktual pelaksanaan produksi dan mampu memperbaharui jadwal-jadwal yang belum terlaksana. Perancangan sistem informasi mengacu kepada model aktivitas sistem manufaktur Systems Integration of Manufacturing dan model Shop Floor Control yang selanjutnya dikembangkan dalam bentuk basis data dan perangkat lunak. Dengan adanya sistem ini, kebutuhan yang cepat dan tepat atas informasi aktual jadwal produksi dapat terpenuhi dan proses penjadwalan ulang terhadap jadwal yang belum dilaksanakan dapat dilakukan dengan mudah, cepat, dan tepat sehingga efektivitas dan efisiensi produksi dapat diraih secara optimal.*

**Kata kunci:** *Penjadwalan, Gap aktualisasi-rencana, Sistem Informasi*

## ABSTRACT

*Scheduling is a decision-making process. Good scheduling can improve the effectiveness and efficiency of production. Nevertheless, frequently in the actualization not all of the schedules are implemented correctly, thus causing a gap between the Actualization and plan. This gap requires a serious and immediate treatment in order to avoid ongoing gap. Therefore, this study designed an information system that can receive the actual execution of the production schedule and be able to update the schedules which are not yet actualized. Information system design refers to the activity models of manufacturing systems Systems Integration of Manufacturing and Shop Floor Control which are then followed by developing a database and software. With this system, the need for done rapid and precise information updates on the actual schedule of production and the process of rescheduling be done easily, quickly, and accurately so that the effectiveness and efficiency of production can be achieved optimally.*

**Keywords:** *Scheduling, Plan-Actualization Gap, Information System.*

---

\* Makalah ini merupakan ringkasan dari Tugas Akhir yang disusun oleh penulis pertama dengan pembimbingan penulis kedua dan ketiga. Makalah ini merupakan draft awal dan akan disempurnakan oleh para penulis untuk disajikan pada seminar nasional dan/atau jurnal nasional.

## 1. PENDAHULUAN

### 1.1 Pengantar

Penjadwalan adalah suatu proses pengambilan keputusan yang digunakan oleh sebagian besar industri manufaktur. Secara teoritis, penjadwalan merupakan usaha mengalokasikan sumber daya yang ada menjadi pekerjaan-pekerjaan yang harus diselesaikan dalam periode waktu tertentu, dan bertujuan untuk menghasilkan keluaran (*output*) yang optimal.

Penjadwalan yang baik tentunya menghasilkan jadwal yang mampu meningkatkan efektivitas dan efisiensi produksi, tetapi pada kenyataannya tidak semua jadwal yang digunakan dapat dilaksanakan secara tepat. Adanya perbedaan antara penjadwalan pada saat perencanaan dengan kondisi pelaksanaan tentunya membutuhkan penanganan yang serius dan segera.

Kesalahan-kesalahan yang terjadi pada saat pelaksanaan relatif mudah untuk diidentifikasi, tetapi usaha penanganan kesalahan-kesalahan ini tidak secepat proses identifikasiannya, khususnya untuk permasalahan penjadwalan itu sendiri. Contohnya, ketika penerapan jadwal tidak lagi sesuai dengan jadwal yang telah direncanakan, tetapi kegiatan produksi masih tetap merujuk pada jadwal tersebut. Kondisi semacam ini dapat menjadi pemicu permasalahan-permasalahan yang berkesinambungan serta permasalahan efektivitas dan efisiensi produksi. Untuk itu dibutuhkan suatu penanganan yang cepat dan tepat.

Kebutuhan yang cepat dan tepat ini, sesuai dengan yang dikemukakan oleh Badworth dan Bailey (1987) dalam bukunya *Integrated Production Control Systems* yang menyatakan, "Akurasi dan kecepatan mendapatkan data dari rantai produksi berpengaruh secara signifikan terhadap efisiensi manufaktur". Data yang dimaksud adalah informasi-informasi aktual yang berkaitan dengan pelaksanaan produksi. Dalam istilah penjadwalan informasi-informasi ini dapat berupa jadwal aktual suatu pekerjaan dimulai dan jadwal aktual suatu pekerjaan berakhir dilaksanakan. Kebutuhan yang cepat dan tepat dalam mendapatkan informasi aktual pelaksanaan produksi dapat dipenuhi dengan menggunakan suatu sistem informasi.

Perkembangan teknologi informasi dewasa ini, menjadi harapan baru yang mungkin saja dapat menjadi solusi dari permasalahan aliran informasi, dan sistem informasi pada aktivitas penjadwalan produksi. Sehingga proses pengambilan keputusan dalam penanganan masalah-masalah yang timbul pada proses penjadwalan dan pelaksanaan penjadwalan dapat dilakukan dengan segera, dan diharapkan dapat meningkatkan efektivitas dan efisiensi pada rantai produksi secara optimal.

### 1.2 Identifikasi Masalah

Penanganan masalah-masalah yang timbul pada proses penjadwalan harus dilakukan dengan cepat dan tepat, agar masalah yang ada tidak mengakibatkan timbulnya masalah baru, khususnya pada jadwal itu sendiri. Upaya penanganan yang cepat dan tepat dapat dilaksanakan apabila informasi yang berkaitan dengan jadwal dapat dihimpun dan diproses secara terkomputerisasi. Oleh karena itu perlu dirancang suatu sistem informasi yang dapat melakukan penjadwalan serta menerima laporan aktual dari pelaksanaan jadwal tersebut, dan mampu melakukan pembaharuan terhadap jadwal-jadwal yang belum dilaksanakan.

## 2. STUDI LITERATUR

### 2.1. Aktivitas Penjadwalan Pada Sistem Manufaktur

Aktivitas sistem manufaktur merupakan keseluruhan kegiatan yang dikerjakan dari diterimanya suatu pesanan hingga produk sampai kepada konsumen. Aktivitas pada sistem

manufaktur secara generik telah dikembangkan oleh *National Institute of Standards* (1995) sebagai bagian dari program SIMA (*System Integration for Manufacturing Applications*).

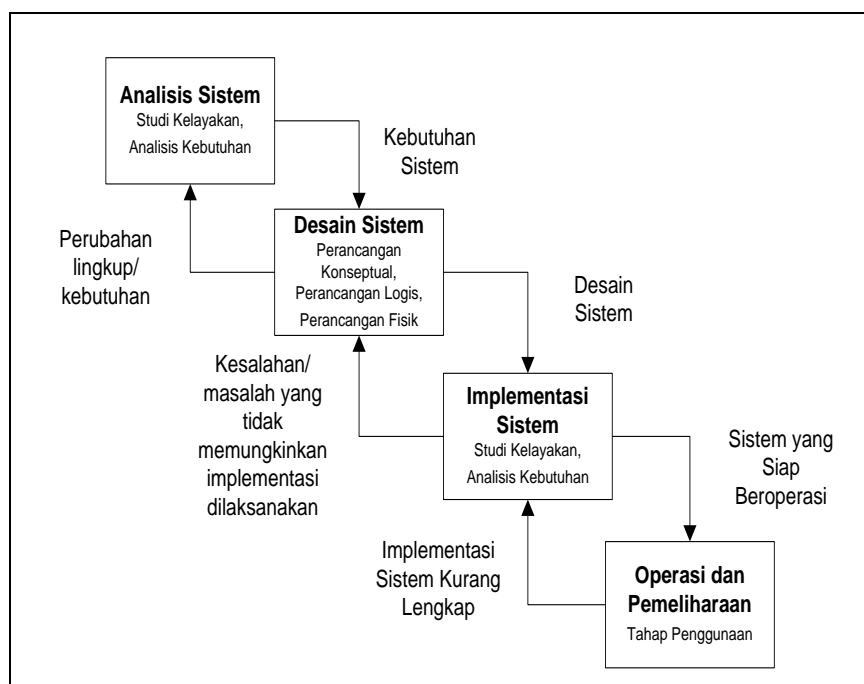
Aktivitas penjadwalan pada SIMA merupakan bagian dari aktivitas membuat produk fisik. Pada tahapan tersebut ditentukan jadwal produksi *detail* untuk semua *job* yang telah dirilis, yang menentukan suatu mesin atau stasiun kerja akan mengerjakan operasi apa, pada *part* apa, dan kapan.

## 2.2. Sistem Informasi

Sistem sering didefinisikan sebagai komponen-komponen yang saling berinteraksi untuk mencapai tujuan tertentu dan dengan beberapa batasan-batasan yang dimilikinya, sedangkan sistem informasi adalah "Kombinasi antara prosedur kerja, informasi, orang, dan teknologi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi" (Alter, 1992).

## 2.3. Mengembangkan Sistem Informasi

Metodologi klasik yang sering digunakan untuk mengembangkan sistem informasi adalah SDLC (*System Development Life Cycle*). Metode ini mencakup kegiatan yang mengawali proyek, menganalisis kebutuhan data, merancang sistem, membuat sistem, dan memelihara sistem. Model SDLC secara umum sering digambarkan dengan model air terjun, seperti diperlihatkan pada Gambar1.



Gambar 1. Model Air Terjun (Sumber: Kadir, 2009)

## 2.4. Database

*Database* atau basis data adalah Kumpulan data yang umumnya menjabarkan aktivitas-aktivitas dari satu atau lebih suatu organisasi yang terkait (Ramakrishnan dan Gehrke, 2000), atau *Database* merupakan kumpulan data yang terorganisir dari data-data yang secara nalar saling terkait (Hoffer *et al.*, 2005).

## 2.5. Konsep Object Oriented

*Object-Oriented* (OO) merupakan salah satu metode yang digunakan dalam pengembangan *software* atau perangkat lunak. Dengan menggunakan OO usaha untuk membuat

perangkat lunak menjadi modular (*Modularity*), sehingga kemungkinan untuk digunakannya kembali suatu perintah perangkat lunak menjadi besar (*Reusability*).

### **2.6. UML (*Unified Modeling Language*)**

UML merupakan salah satu *tools* yang dapat digunakan untuk merancang perangkat lunak (NIIT, 2004). UML sendiri memiliki sembilan diagram yang dapat digunakan dalam perancangan perangkat lunak. Sembilan diagram itu adalah: *Use case diagrams*, *Class diagrams*, *Object diagrams*, *Collaboration diagrams*, *Sequence diagrams*, *State diagrams*, *Activity diagrams*, *Component diagrams*, *Deployment diagrams*.

## **3. METODOLOGI PENELITIAN**

Metodologi penelitian yang digunakan pada penelitian ini terbagi kedalam tiga bagian utama yaitu: tahapan pendahuluan, tahapan perancangan dan implementasi sistem, serta kesimpulan dan saran.

### **3.1. Tahapan Pendahuluan**

Tahapan ini melakukan indentifikasi permasalahan penjadwalan pada sistem manufaktur, dan indentifikasi aktivitas-aktivitas yang terjadi pada sistem manufaktur berdasarkan referensi model SIMA dan model *Shop Floor Control* (SFC). Berdasarkan model SIMA didapatkan aliran informasi yang terjadi pada sistem manufaktur, dan dari model SFC didapatkan elemen-elemen yang terlibat pada sistem penjadwalan.

### **3.2. Tahapan Perancangan dan Implementasi**

Tahapan ini terdiri dari dua tahapan utama, tahapan pertama adalah perancangan dan implementasi basis data dan tahapan kedua adalah perancangan dan implementasi perangkat lunak. Rincian dari tahapan-tahapan tersebut adalah sebagai berikut:

#### **1. Basis Data**

Pada tahapan ini dilakukan pengembangan model berdasarkan model referensi yang digunakan, pengembangan model ini dilakukan dengan menggunakan *Data Flow Diagram*. DFD tersebut memperlihatkan entitas eksternal yang terlibat dengan sistem beserta aliran data yang terjadi antar entitas dan sistem. Berdasarkan entitas dan aliran data yang terjadi dilakukan perancangan basis data, perancangan ini dimulai dari perancangan logis basis data, normalisasi data, dan perancangan fisik basis data. Pada proses perancangan fisik basis data, penggunaan RDBM (*Relational Database Managemen System*) ikut dipertimbangkan. Adapun RDBMS yang digunakan pada penelitian ini adalah MySQL versi 5.0.

#### **2. Perangkat Lunak**

Perancangan perangkat lunak dilakukan dengan membuat arsitektur perangkat lunak, arsitektur ini menggambarkan bagian-bagian pada perangkat lunak yang digunakan pada sistem. Selanjutnya dilakukan indentifikasi operasi yang dilakukan masing-masing bagian dengan *tools* berupa UML yaitu *use case diagrams* kemudian dari masing-masing *use case* dilakukan indentifikasi tahapan-tahapan yang terjadi dengan menggunakan *sequence diagrams*. Selanjutnya dilakukan desain tampilan dari perangkat lunak berdasarkan operasi-operasi yang dilakukan oleh masing-masing bagian yang terdapat pada perangkat lunak yang akan diimplementasikan dengan menggunakan bahasa pemrograman Java.

### **3.3. Tahapan Pengujian dan Analisis**

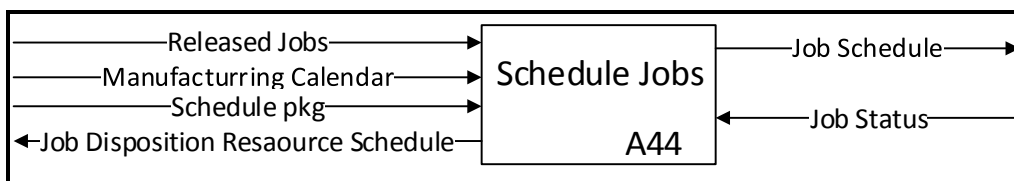
Pada tahapan ini dilakukan pengujian dan analisis terhadap keakuratan *output* yang dihasilkan perangkat lunak terhadap rancangan sistem.

### 3.4. Tahapan Perumusan Kesimpulan dan Saran

Pada tahapan ini, dipaparkan beberapa kesimpulan dari penelitian yang telah dilakukan beserta beberapa saran yang dapat digunakan pada penelitian-penelitian selanjutnya.

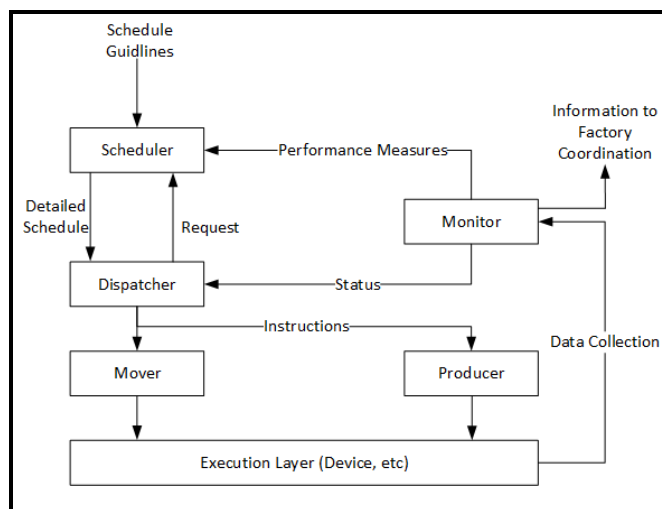
## 4. PERANCANGAN SISTEM

Sistem informasi yang dikembangkan pada penelitian ini mengikuti aktivitas-aktivitas sistem manufaktur yang terdapat pada model umum sistem manufaktur yang dikembangkan oleh *National Institute of Standards*. Model ini merupakan model referensi dari *Systems Integration of Manufacturing Applications (SIMA)* yang dikembangkan oleh *National Institute of Standards and Technology*. Informasi yang masuk dan keluar dari aktivitas yang menjadi fokus penelitian ini dapat dilihat pada Gambar 2. Selain model tersebut, penelitian ini juga menggunakan model dasar sistem *shop floor control* yang dikembangkan oleh Bauer *et al.* (1994). Model dasar tersebut dapat dilihat pada Gambar 3.



**Gambar 2. Diagram IDEF0 Aliran Data Aktivitas Schedule Jobs (Sumber: National Institute of Standards, 1995)**

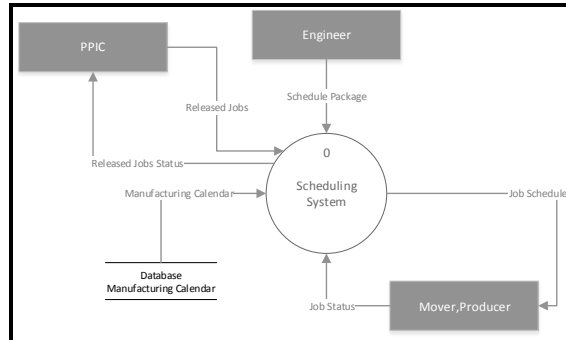
*Manufacturing Calendar* adalah data yang berkaitan dengan waktu yang tersedia untuk melakukan kegiatan produksi, seperti *shift* yang tersedia dan waktu yang tersedia untuk setiap *shift*-nya. *Schedule Package* adalah data yang berisi informasi spesifik tentang urutan stasiun kerja atau mesin yang dilalui oleh suatu benda kerja. *Released Jobs* adalah data yang berisi informasi mengenai produk yang dirilis beserta *earliest scheduling date*, jumlah produk (*a batch*) beserta *duedate*-nya. *Job Schedule* adalah data yang berisi informasi tentang pelaksanaan suatu operasi dari job tertentu di mesin tertentu beserta batasan waktu pengerjaannya, informasi ini sering juga disebut sebagai *Dispatch list*. *Job Status* adalah data yang berupa laporan tentang pelaksanaan produksi apakah suatu pekerjaan hendak dilaksanakan, sedang dilaksanakan, atau telah selesai dilaksanakan. *Job Disposition Resource Schedule* adalah data yang berisi informasi bahwa *released jobs* sesuai atau tidak sesuai dengan sumber daya yang tersedia.



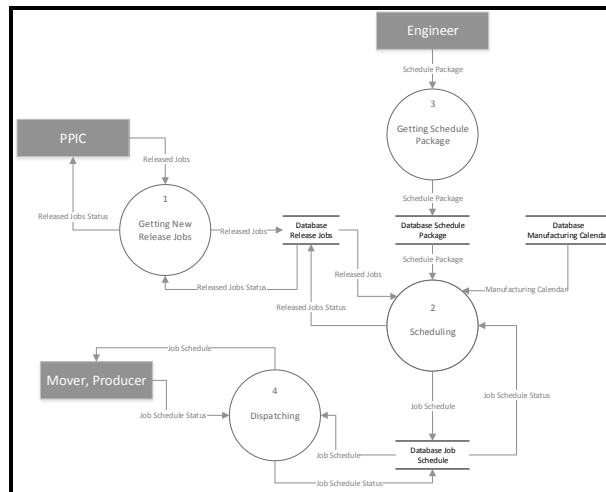
**Gambar 3. Model Dasar Sistem Shop Floor Control (Sumber: Bauer et al., 1994)**

#### 4.1. Pengembangan Model Konseptual

Model konseptual dari proses penjadwalan beserta aliran data yang terjadi pada sistem penjadwalan digambarkan dengan menggunakan DFD (*Data Flow Diagram*). Untuk DFD level-0 terlihat pada Gambar 4, dan DFD level-1 pada Gambar 5.



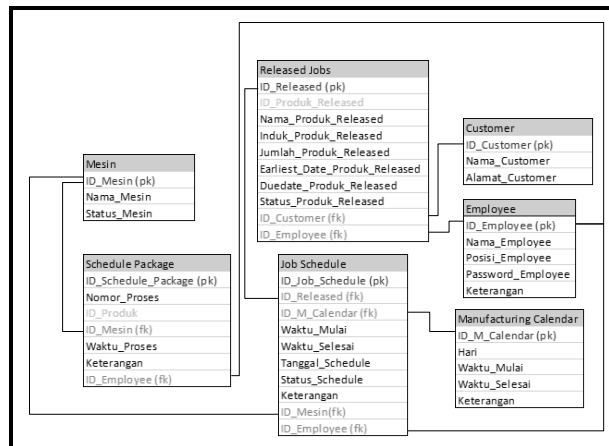
Gambar 4. DFD Level-0 Sistem Penjadwalan



Gambar 5. DFD Level 1 Sistem Penjadwalan

#### 4.2. Rancangan Basis Data

Rancangan logis basis data dari pengembangan model konseptual dan telah melalui proses normalisasi dapat dilihat pada Gambar 6.



Gambar 6. Rancangan Basis Data

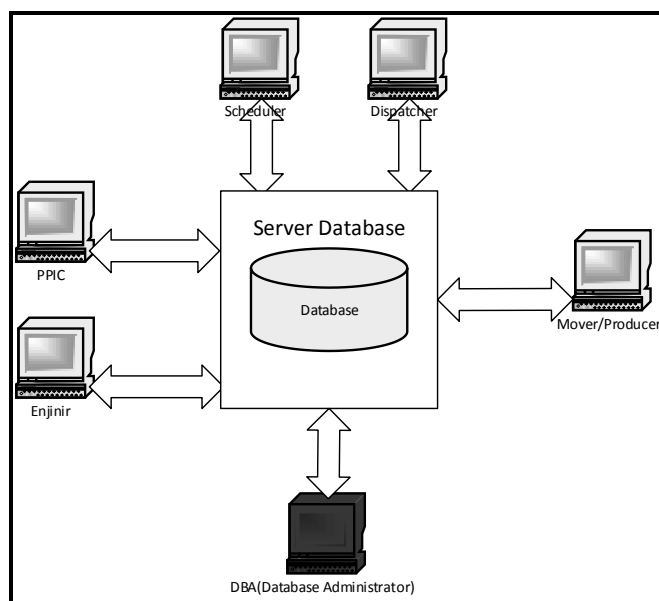
Berdasarkan rancangan basis data dilakukan perancangan fisik basis data dan salah satu contoh rancangan fisik basis data dapat dilihat pada Tabel 1.

**Tabel 1. Contoh Rancangan Fisik Basis Data *Job Schedule***

| Nama Field       | Tipe Data | Lebar | Deskripsi   |
|------------------|-----------|-------|---|
| ID_Job_Schedule  | CHAR      | 10    | Primary key table <i>job schedule</i>             |
| ID_Released      | CHAR      | 10    | Referensi <i>ID released</i>                      |
| ID_M_Calendar    | CHAR      | 10    | Referensi <i>manufacturing calendar</i>           |
| Waktu_Mulai      | TIME      |       | Waktu mulai                                       |
| Waktu_Selesai    | TIME      |       | Waktu selesai                                     |
| Tanggal_Schedule | DATE      |       | Tanggal Pelaksanaan                               |
| Status_Schedule  | VARCHAR   | 10    | Status "SCHEDULE", "PROSES", "FINISH".            |
| Keterangan       | TEXT      |       | Keterangan <i>job schedule</i>                    |
| ID_Mesin         | CHAR      | 10    | Mesin yang digunakan                              |
| ID_Employee      | CHAR      | 10    | Menandakan <i>scheduler</i> yg mengerjakan jadwal |

### 4.3. Rancangan Aplikasi Perangkat Lunak

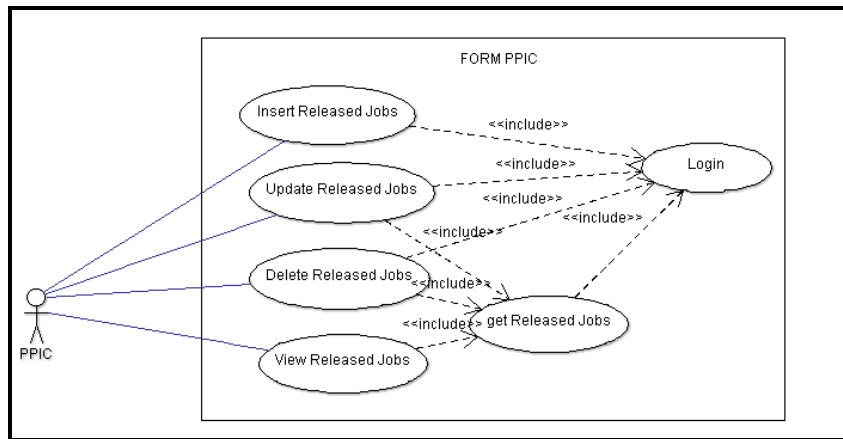
Perangkat lunak berperan dalam pengolahan basis data, agar dapat mengubah data menjadi informasi yang dibutuhkan. Perangkat lunak yang dikembangkan pada penelitian ini terbagi menjadi beberapa bagian yang bekerja sesuai dengan kebutuhannya masing-masing. Arsitektur perangkat lunak yang dikembangkan pada penelitian ini dapat dilihat pada Gambar 7.



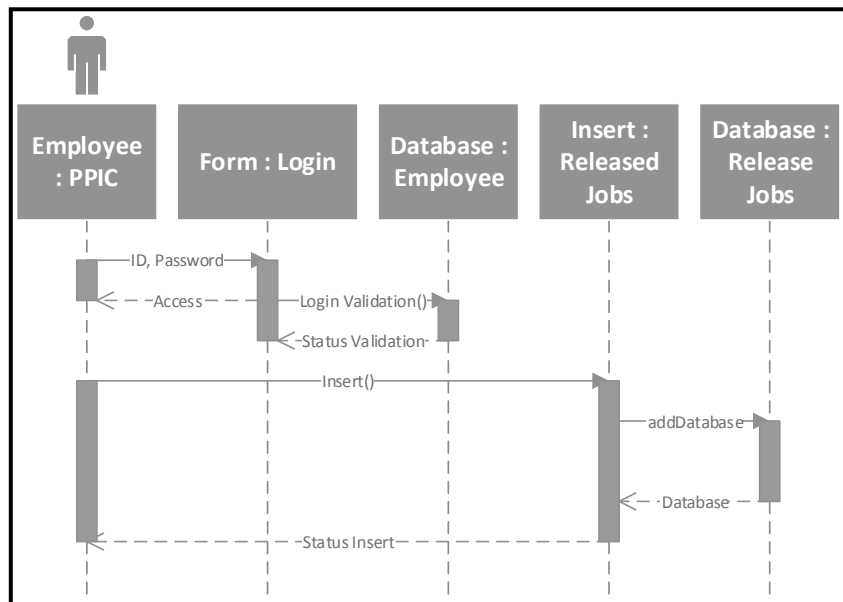
**Gambar 7. Arsitektur Perangkat Lunak**

Bagian-bagian pada arsitektur perangkat lunak melakukan aktivitas sesuai dengan fungsinya masing-masing. PPIC bertugas memberikan data *released jobs*, dan dapat menerima informasi status pengerjaan *released jobs* yang diberikan, *Engineer* bertugas memberikan dan memanipulasi data *schedule package*, *Scheduler* menentukan *released jobs* yang akan diubah menjadi *job schedule*, serta dapat melakukan manipulasi pada *job schedule* yang telah dibuat, *Dispatcher* melakukan penugasan terhadap *mover/producer* berdasarkan *job schedule* yang ada. *Mover/producer* mengerjakan *job schedule* yang diberikan oleh *Dispatcher* serta melaporkan hasil aktual pelaksanaan *job schedule* tersebut. Aktivitas dari masing-masing bagian tersebut digambarkan dengan menggunakan *use case diagrams* dan urutan proses dari masing-masing *use case* digambarkan dengan menggunakan *sequence*

diagrams. Contoh *use case diagram* untuk bagian PPIC dan *sequence diagram* dari *use case insert released job* dapat dilihat pada Gambar 8 dan Gambar 9.

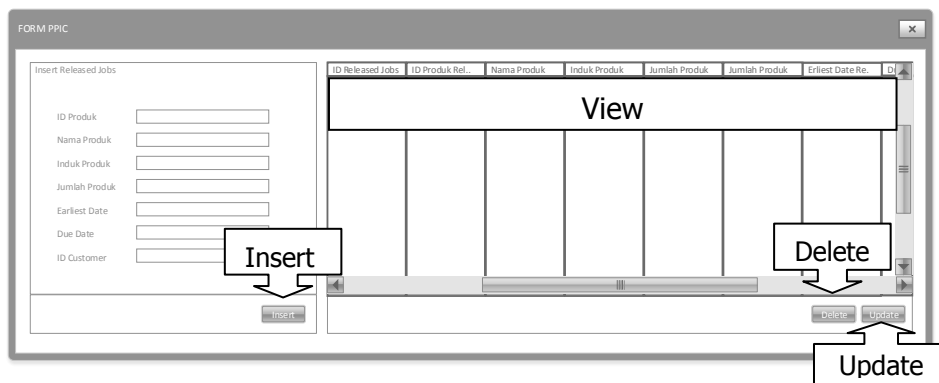


**Gambar 8. Contoh Use Case Diagram PPIC**



**Gambar 9. Contoh Sequence Diagram dari Insert Released Jobs**

Berdasarkan *use case diagram* PPIC pada Gambar 8 dilakukan perancangan GUI (*Graphical User Interface*) dari PPIC dan contoh rancangan GUI tersebut dapat dilihat pada Gambar 10.



**Gambar 10. Contoh Desain GUI dari Form PPIC**



## 5. PENGUJIAN DAN ANALISIS

Pengujian perangkat dilakukan untuk memastikan perangkat lunak bekerja sesuai dengan tujuan sistem. Pengujian perangkat lunak dilakukan dengan mengikuti tiga skenario utama. Tiga skenario utama yaitu skenario *input*, *generate job schedule*, *receive actual job schedule*.

### 5.1. Skenario *Input*

Proses *input* terjadi di beberapa bagian seperti, PPIC yang memberikan *input* berupa *released jobs*, *Engineer* memberikan *input* berupa *schedule package* dan DBA memberikan data *employee* dan *manufacturing calendar*. PPIC memberikan *released jobs* dan setiap *released jobs* yang masuk berstatus "BARU" hingga *released jobs* tersebut diperiksa oleh bagian *scheduler*. Apabila *released jobs* yang masuk tidak dapat dilaksanakan sesuai batas waktu yang diinginkan maka *released jobs* akan berubah status menjadi "RENEGOSIASI(n-Hari)" dan aplikasi perangkat lunak memberikan tanda bahwa terdapat *released jobs* yang berstatus "RENEGOSIASI(n-Hari)", sebaliknya apabila *released jobs* berhasil dijadwalkan oleh *scheduler* maka *released jobs* akan berubah status menjadi "FINISH". Untuk *schedule package* yang diberikan oleh *engineer* harus dipastikan berasal dari produk yang berbeda, sehingga tidak ada produk yang memiliki *schedule package* ganda. Oleh karena itu, perangkat lunak harus dapat memberikan peringatan ketika *engineer* memasukkan *schedule package* untuk produk yang sama. Sedangkan data *employee* dan *manufacturing calendar* yang diberikan oleh DBA digunakan untuk memberikan izin kepada *employee* tertentu untuk melakukan aktivitas-aktivitas pada bagian-bagian seperti yang terlihat pada arsitektur perangkat lunak (Gambar 7) dan *manufacturing calendar* digunakan sebagai dasar penyusunan *job schedule*. Antara satu *employee* dengan *employee* lainnya dibedakan berdasarkan ID-nya. Untuk *manufacturing calendar* selain dibedakan berdasarkan ID-nya juga dibedakan berdasarkan waktu mulai dan waktu selesainya. Oleh karena itu, tidak ada *manufacturing calendar* yang memiliki waktu yang sama bahkan waktu yang beririsan. Skenario *input* pada saat melakukan *input* data *released job* yang dilakukan oleh PPIC dapat dilihat pada Tabel 2, Tabel 3, dan Gambar 11.

**Tabel 2. *Input Released Job (BARU)***

| ID Released(pk) | ID Produk Released | Nama Produk Released | Induk Produk Released | Jumlah Produk Released | Earliest Date Produk Released | Duedate Produk Released | Status Produk Released |
|-----------------|--------------------|----------------------|-----------------------|------------------------|-------------------------------|-------------------------|------------------------|
| R1              | P1                 | A                    |                       | 10                     | 2013-05-04                    | 2013-05-05              | BARU                   |

Pada saat *released job* yang masuk dijadwalkan oleh *Scheduler* dan ternyata belum dapat dijadwalkan berdasarkan batasan waktu yang diberikan (*Duedate\_Produk\_Released*) maka *released job* akan berubah status menjadi "RENEGOSIASI" seperti terlihat pada Tabel 3, dan pemberitahuan adanya *released job* yang memerlukan renegotiasi terlihat pada Gambar 11.

**Tabel 3. *Input Released Job (RENEGOSIASI)***

| ID Released(pk) | ID Produk Releas | Nama Produk Releas | Induk Produk Releas | Jumlah Produk Releas | Earliest Date Produk Released | Duedate Produk Released | Status Produk Released |
|-----------------|------------------|--------------------|---------------------|----------------------|-------------------------------|-------------------------|------------------------|
| R1              | P1               | A                  |                     | 10                   | 2013-05-04                    | 2013-05-05              | RENEGOSIASI(2)         |

Gambar 11. Form PPIC

### 5.2. Skenario *Generate Job Schedule*

*Generate Job Schedule* adalah aktivitas mengubah *released jobs* menjadi *job schedule* berdasarkan aturan FCFS (*First Come First Served*). FCFS yang dimaksud berdasarkan *Earliest Date* Produk *Released* terkecil dan dengan status "BARU". Setelah didapatkan *released jobs* dengan *Earliest Date* Produk *Released* terkecil, selanjutnya perangkat lunak melakukan pemeriksaan *schedule package* berdasarkan *ID\_Produk\_Released* dari *released jobs* yang terpilih. Ketika *schedule package* (Tabel 4) yang sesuai tersedia maka penjadwalan atau pembuatan *job schedule* dapat dilakukan. Ketika belum terdapat *job schedule* sebelumnya maka penjadwalan dapat segera dilakukan namun bila telah ada *job schedule* sebelumnya aplikasi perangkat lunak akan mencari waktu yang tepat lebih dahulu dengan cara memeriksa *job schedule* untuk mesin yang sama untuk setiap *manufacturing calendar* (Tabel 5) yang tersedia, sedangkan jadwal yang terikat dengan jadwal sebelumnya dimulai pada waktu berakhirnya jadwal pendahulunya. *Released job* yang digunakan pada skenario ini adalah *released job* pada Tabel 2. Berdasarkan *released job*, *schedule package*, dan *manufacturing calendar* tersebut maka akan menghasilkan *job schedule* seperti yang terlihat pada Tabel 6.

Tabel 4. *Schedule Package* (TERSEDIA)

| ID Schedul Package (pk) | Nomor Proses | ID Produk | ID Mesin (fk) | Waktu Proses | Keterangan     | ID Employee (fk) |
|-------------------------|--------------|-----------|---------------|--------------|----------------|------------------|
| SP1                     | 1            | P1        | M1            | 5            | <i>Cutting</i> | E6               |
| SP2                     | 2            | P1        | M3            | 10           | <i>Tooling</i> | E6               |
| SP3                     | 3            | P1        | M2            | 7            | <i>Brazing</i> | E6               |
| SP4                     | 1            | P2        | M1            | 8            | <i>Tooling</i> | E6               |
| SP5                     | 2            | P2        | M2            | 5            | <i>Tooling</i> | E6               |
| SP6                     | 3            | P2        | M3            | 2            | <i>Tooling</i> | E6               |

Tabel 5. *Manufacturing Calendar* (TERSEDIA)

| ID M Calendar | Hari   | Waktu Mulai | Waktu Selesai | Keterangan |
|---------------|--------|-------------|---------------|------------|
| MC1           | SENIN  | 8:00:00     | 12:00:00      | Shift A    |
| MC2           | SELASA | 8:00:00     | 12:00:00      | Shift B    |
| MC3           | RABU   | 8:00:00     | 12:00:00      | Shift C    |
| MC4           | KAMIS  | 7:00:00     | 16:00:00      | Shift D    |

**Tabel 6. Job Schedule**

| ID Manufacturing Calendar | Waktu Mulai | Waktu Selesai | Tanggal Schedule | Keterangan | ID Mesin | ID Employee |
|---------------------------|-------------|---------------|------------------|------------|----------|-------------|
| MC1                       | 8:00:00     | 8:50:00       | 2013-05-06       | Terjadwal  | M1       | E2          |
| MC1                       | 8:50:00     | 10:30:00      | 2013-05-06       | Terjadwal  | M3       | E2          |
| MC1                       | 10:30:00    | 11:40:00      | 2013-05-06       | Terjadwal  | M2       | E2          |

### 5.3 Skenario *Receive Actual Job Schedule*

*Receive actual job schedule* adalah kondisi dimana *Mover/Producer* mengirimkan informasi mengenai jadwal aktual pelaksanaan produksi berdasarkan jadwal atau job schedule yang telah dihasilkan oleh *Scheduler* dan telah ditugaskan oleh *Dispatcher*. Kemungkinan-kemungkinan yang terjadi adalah jadwal dilaksanakan lebih awal dari jadwal yang telah ditetapkan, jadwal dilaksanakan terlambat dari jadwal yang telah ditetapkan, atau jadwal telah dilaksanakan sesuai dengan jadwal yang telah ditetapkan namun selama prosesnya tiba-tiba terjadi kerusakan mesin atau penyebab lainnya yang mengakibatkan proses tertunda selama beberapa waktu. Ketiga skenario tersebut bila dialami *job schedule* pada Tabel 6, akan menghasilkan *job schedule* seperti yang terlihat pada Tabel 7, Tabel 8, dan Tabel 9.

**Tabel 7. Job Schedule Dilaksanakan Lebih Awal**

| ID Manufacturing Calendar | Waktu Mulai | Waktu Selesai | Tanggal Schedule | Keterangan | ID Mesin | ID Employee |
|---------------------------|-------------|---------------|------------------|------------|----------|-------------|
| MC1                       | 7:55:00     | 8:45:00       | 2013-05-06       | Lebih Awal | M1       | E2          |
| MC1                       | 8:45:00     | 10:30:00      | 2013-05-06       | Lebih Awal | M3       | E2          |
| MC1                       | 10:30:00    | 11:40:00      | 2013-05-06       | Lebih Awal | M2       | E2          |

**Tabel 8. Job Schedule Dilaksanakan Terlambat**

| ID Manufacturing Calendar | Waktu Mulai | Waktu Selesai | Tanggal Schedule | Keterangan | ID Mesin | ID Employee |
|---------------------------|-------------|---------------|------------------|------------|----------|-------------|
| MC1                       | 8:05:00     | 8:55:00       | 2013-05-06       | Terlambat  | M1       | E2          |
| MC1                       | 8:55:00     | 10:35:00      | 2013-05-06       | Terlambat  | M3       | E2          |
| MC1                       | 10:35:00    | 11:45:00      | 2013-05-06       | Terlambat  | M2       | E2          |

**Tabel 9. Job Schedule Mengalami Penundaan**

| ID Manufacturing Calendar | Waktu Mulai | Waktu Selesai | Tanggal Schedule | Keterangan       | ID Mesin | ID Employee |
|---------------------------|-------------|---------------|------------------|------------------|----------|-------------|
| MC1                       | 8:00:00     | 8:55:00       | 2013-05-06       | Tertunda 5 menit | M1       | E2          |
| MC1                       | 8:55:00     | 10:35:00      | 2013-05-06       | Terlambat        | M3       | E2          |
| MC1                       | 10:35:00    | 11:45:00      | 2013-05-06       | Terlambat        | M2       | E2          |

## 6. KESIMPULAN

### 6.1 Ringkasan

Pada penelitian ini telah dibangun suatu sistem informasi yang dapat digunakan untuk melakukan penjadwalan berdasarkan metode FCFS (*First Come First Served*), serta dapat melakukan pembaharuan jadwal berdasarkan waktu aktual yang didapatkan dari lantai produksi. Selain itu, sistem dapat memberikan peringatan kepada bagian PPIC yang

memasukkan data *released jobs* bahwa suatu *released jobs* diterima atau perlu untuk di renegotiasi.

Sistem informasi yang telah dikembangkan dapat digunakan sebagai solusi dari keinginan mendapatkan informasi dari rantai produksi secara cepat dan tepat serta dapat digunakan sebagai sarana peningkatan produktivitas rantai produksi.

## 6.2 Saran

Pengembangan pada penelitian selanjutnya, salah satunya dapat dilakukan dengan cara menggunakan metode penjadwalan yang dapat menghasilkan jadwal yang lebih baik, seperti penyusunan jadwal dengan metode-metode heuristik atau meta-heuristik, validasi setiap proses *input* data dapat dilakukan dengan lebih rinci, pemrograman yang berkaitan dengan manipulasi data dapat dilakukan secara langsung pada RDBMS (*Relational Database Management Systems*) agar meminimisasi penggunaan *computer resource* (RAM), dan menggunakan perangkat otomatisasi dalam proses akuisisi data atau mendapatkan data aktual pada rantai produksi, sehingga keterlambatan-keterlambatan yang diakibatkan kesalahan atau kelalaian manusia dapat dihindari.

## REFERENSI

Alter, Steven. (1992). *The Information Systems: A Management Perspective*. The Benjamin/Cummings Publishing Company Inc., USA.

*National Institute of Standards*. (1995). *SIMA Reference Architecture, Part1: Activity Models, National Institute of Standards and Technology*, United States.

Bedworth, David D. dan Bayley, James E. (1987). *Integrated Production Control Systems*, Edisi Kedua. John Wiley & Sons., Singapore.

Beuer, A., Bowden, R., Browne, J., Duggan., dan Lyons, G. (1994). *Shop Floor Control Systems, From Design to Implementation*. Chapman & Hall., Great Britain.

Hoffer, Jeffrey A., Prescott, Mary B, McFadden, Fred B. (2005). *Modern Systems Analysis and Design*, 4th ed. Pearson Prentice Hall., New Jersey.

Kadir, Abdul. (2009). *Dasar Perancangan dan Implementasi Database Relasional*. ANDI., Yogyakarta.

NIIT. (2004). *Object Oriented Analysis and Design using UML: UML and Requierments Modeling, Studen Guide*. Sona Printers Pvt. Ltd. New Delhi.

Ramakrishnan, Raghu. dan Grekhe, Johannes. (2000). *Database Management Systems*. The McGraw-Hill Companies Inc., USA.