

SKRIPSI

**PENGEMBANGAN PERANGKAT LUNAK SIMULATOR
REAL-TIME SYSTEM MENGGUNAKAN ALGORITMA
*EARLIEST DEADLINE FIRST***



Kevin Pratama

NPM: 2014730073

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2018**

UNDERGRADUATE THESIS

**REAL-TIME SYSTEM SIMULATOR SOFTWARE
DEVELOPMENT USING THE EARLIEST DEADLINE FIRST
ALGORITHM**



Kevin Pratama

NPM: 2014730073

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2018**

LEMBAR PENGESAHAN



**PENGEMBANGAN PERANGKAT LUNAK SIMULATOR
REAL-TIME SYSTEM MENGGUNAKAN ALGORITMA
EARLIEST DEADLINE FIRST**

Kevin Pratama

NPM: 2014730073

Bandung, 13 Desember 2018

Menyetujui,

Pembimbing

Dott. Thomas Anung Basuki

Ketua Tim Penguji

Elisati Hulu, M.T.

Anggota Tim Penguji

Raymond Chandra Putra, S.T., M.T.

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng



PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

**PENGEMBANGAN PERANGKAT LUNAK SIMULATOR *REAL-TIME*
SYSTEM MENGGUNAKAN ALGORITMA *EARLIEST DEADLINE*
*FIRST***

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 13 Desember 2018



Kevin Pratama
NPM: 2014730073

ABSTRAK

Di zaman modern ini, semakin banyak pekerjaan yang perlu dijadwalkan dengan program komputer. Kadangkala muncul masalah, yaitu jumlah pekerjaan yang sangat banyak dan tidak dapat ditangani. Contohnya adalah beberapa perangkat *server* yang mengalami *crash* pada saat terjadi lonjakan permintaan yang tidak terduga. Oleh karena itu, dibutuhkan sebuah penjadwalan pada teknologi tersebut agar dapat melayani permintaan yang ada.

Tujuan dari skripsi ini adalah untuk menganalisis, merancang, dan mengembangkan perangkat lunak simulasi penjadwalan *real-time system*. Algoritma penjadwalan yang digunakan adalah *earliest deadline first*. Algoritma ini menerima input berupa tipe penjadwalan (*hard real time* atau *soft real time*), *Burst Time*, *Arrival Time*, dan *Deadline* dari masing-masing pekerjaan. *Output* dari perangkat lunak ini adalah hasil penjadwalan masing-masing pekerjaan dalam bentuk *gant chart*, *average waiting time*, *average turnaround time*, dan pekerjaan – pekerjaan yang melewati tenggat waktu.

Pengujian perangkat dilakukan dalam mode *hard real time* dan *soft real time* dengan kondisi ada / tidak ada *idle*, dan ada/tidak ada *miss*. Pengujian juga dilakukan secara eksperimental, dimana perangkat lunak dapat bekerja meskipun diberikan *test case* yang beragam. Serta dengan melakukan pengujian eksperimental lanjut, dipastikan bahwa perangkat lunak dapat bekerja dengan jumlah *test case* yang sangat banyak (ratusan atau ribuan pekerjaan).

Selain itu, dapat juga diambil kesimpulan bahwa semakin banyak pekerjaan yang dijadwalkan oleh perangkat lunak, maka semakin besar pula nilai *average waiting timenya*. Nilai *average turnaround time* bervariasi (naik dan turun) tergantung dari selisih kapan sebuah pekerjaan selesai dijadwalkan dan *arrival time* dari pekerjaan tersebut. Lalu, Semakin besar selisih dari *burst time* dan *deadline time* pada pekerjaan, maka semakin jauh perbedaan AWT dan ATTnya. Sebaliknya, semakin kecil selisih dari *burst time* dan *deadline time*, maka semakin seragam perbedaan AWT dan ATTnya.

Kata-kata kunci: Real time system, earliest deadline first algorithm

ABSTRACT

In this modern era, there are more jobs that need to be scheduled using computer programs. Sometimes problems arise, i.e. the number of jobs is very large and cannot be handled by the computer. The example is a number of servers that experience crashes when an unexpected surge in demand occurs. Therefore, scheduling technology is needed in order to serve the demands. The purpose of this undergraduate thesis is to analyze, design, and develop a real-time system simulation software. The scheduling algorithm used by this software is earliest deadline first. This algorithm receives an input in the form of scheduling type (hard real time or soft real time), burst time, arrival time, and deadline time from each job. The output of the software is the result of the scheduling of each job in the form of gantt chart, average waiting time, average turnaround time, and number of jobs that already exceed the deadline time.

The software testing is done in hard real time and soft real time mode with idle / not idle, and miss / not miss condition. Testing is also carried out experimentally, where software can work even though various test case tests are given. As well as by conducting further experimental testing, it is ensured that the software can work with a large number of test cases (hundreds or thousands of jobs).

In addition, it can also be concluded that the more jobs scheduled by the software, the greater the value of average waiting time. Besides, the value of average turnaround time varies (up and down) depending on the difference between completion time and arrival time of the job. Then, the greater the difference between the burst time and deadline time on the job, the further the difference between AWT and ATT. Conversely, the smaller the difference between burst time and deadline time, the more uniform the difference between AWT and ATT.

Keywords: Real time system, earliest deadline first algorithm

*Untuk Mama, Papa, Dennis, Bapak Bagoes, dan teman - teman
terbaik...*

KATA PENGANTAR

Puji Syukur kepada Tuhan Yang Maha Esa atas selesainya skripsi yang berjudul *Real Time System* dengan *Earliest Deadline First* ini. Atas dukungan moral yang diberikan pada saat penyusunan skripsi ini, saya mengucapkan terima kasih kepada :

1. Mama dan Papa yang selalu mendukung saya menyelesaikan skripsi saya ini
2. Bapak Bagoes yang membimbing saya selama mengerjakan skripsi ini.
3. Ibu Mariskha yang sudah memberikan petunjuk - petunjuk penting penyelesaian skripsi ini, dan;
4. Teman - teman saya di jurusan informatika tercinta yang terus mendukung saya.

Tanpa adanya pihak - pihak diatas, penyusunan skripsi ini mungkin belum sempurna sekarang. Skripsi ini disusun untuk memenuhi syarat memperoleh gelar Sarjana Teknik (S.T.) dalam bidang keahlian Teknik Informatika pada Fakultas Teknologi Informasi dan Sains Universitas Katolik Parahyangan, Bandung.

Dengan keterbatasan pengalaman, waktu menulis dan ilmu pustaka yang ditinjau, penulis menyadari bahwa masih banyak kekurangan yang ada pada skripsi ini sehingga dibutuhkan pengembangan lebih lanjut agar benar - benar bermanfaat. Oleh karena itu, penulis sangat menerima kritik dan saran agar skripsi ini menjadi lebih baik lagi.

Akhir kata, penulis berharap skripsi ini dapat memberikan manfaat bagi kita semua terutama untuk pengembangan ilmu pengetahuan pada bidang teknologi informasi.

Bandung, Desember 2018

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	2
1.6 Sistematika Pembahasan	3
2 DASAR TEORI	5
2.1 <i>Real Time System</i> dan <i>Scheduling</i> [1]	5
2.1.1 Definisi <i>Real Time System</i> [2]	5
2.1.2 Definisi <i>Scheduling</i> [2]	5
2.2 Cara kerja dan jenis – jenis sistem penjadwalan [2]	6
2.3 Contoh nyata sistem penjadwalan <i>real time</i>	7
2.4 <i>Gantt Chart</i>	8
2.5 Algoritma <i>Earliest Deadline First</i>	9
2.6 Contoh penjadwalan <i>soft real time</i> dengan algoritma <i>earliest deadline first</i>	9
2.7 Contoh penjadwalan <i>hard real time</i> dengan algoritma <i>earliest deadline first</i>	11
3 ANALISIS	13
3.1 Cara Kerja Algoritma	13
3.2 Analisis cara kerja <i>Hard Real Time</i> dan <i>Soft Real Time</i> [1]	14
3.3 Analisis Perangkat Lunak Sejenis	14
3.3.1 Pemodelan <i>Use Case</i>	19
3.3.2 Diagram Kelas	22
4 PERANCANGAN	25
4.1 Rancangan algoritma <i>earliest deadline first</i>	25
4.2 Perancangan antarmuka pengguna	27
4.3 Rancangan diagram aktivitas	29
4.4 Rancangan diagram kelas	31
5 IMPLEMENTASI DAN PENGUJIAN	37
5.1 Lingkungan Implementasi	37
5.1.1 Lingkungan Perangkat Keras	37
5.1.2 Lingkungan Perangkat Lunak	37

5.2	Pengujian	38
5.2.1	Pengujian Fungsional	38
5.2.2	Pengujian Eksperimental	40
6	KESIMPULAN DAN SARAN	53
6.1	Kesimpulan	53
6.2	Saran	54
	DAFTAR REFERENSI	55
A	KODE PROGRAM PERANGKAT LUNAK SEJENIS	57
B	KODE PROGRAM PERANGKAT LUNAK <i>EDF Simulator</i>	63
C	<i>Test case</i> PENGUJIAN EKSPERIMENTAL LANJUT	75
C.1	<i>Soft Real Time Test Case</i>	75
C.2	<i>Hard Real Time Test Case</i>	76

DAFTAR GAMBAR

2.1	Seismogram (Sumber : ilmugeografi.com)	7
2.2	Contoh Implementasi <i>Gantt Chart</i> (Sumber : asq.org).	8
2.3	<i>Pseudocode</i> algoritma EDF	9
2.4	<i>Gantt chart</i> hasil penjadwalan	10
2.5	<i>Gantt chart</i> hasil penjadwalan	12
3.1	<i>Gantt Chart</i> SRTF	14
3.2	<i>Gantt Chart</i> EDF	14
3.3	Antarmuka program sebelum penjadwalan dieksekusi	18
3.4	Antarmuka program setelah dieksekusi	18
3.5	<i>Gantt Chart</i> dari hasil eksekusi program	19
3.6	Diagram <i>use case</i> SRTF	19
3.7	Diagram <i>use case</i> EDF	21
3.8	Diagram kelas SRTF	22
3.9	Diagram kelas EDF	24
4.1	<i>Main window</i> perangkat lunak EDF	27
4.2	<i>Main window</i> perangkat lunak EDF setelah penjadwalan.	28
4.3	Diagram aktivitas menghitung penjadwalan	30
4.4	Diagram kelas perangkat lunak EDF	31
4.5	Diagram kelas EDFCore	32
4.6	Diagram kelas EDFUserInterface	33
4.7	Diagram kelas Job	34
4.8	Diagram kelas JobComparator	35
4.9	Diagram kelas BurstTimeJobComparator	35
5.1	Grafik hasil pengujian <i>soft real time system</i> (tidak ada <i>miss</i> dan <i>idle</i>)	48
5.2	Grafik hasil pengujian <i>hard real time system</i> (ada <i>miss</i> dan tidak ada <i>idle</i>)	49
5.3	Hasil penjadwalan <i>soft real time test case</i>	50
5.4	Hasil penjadwalan <i>hard real time test case</i>	50
5.5	Grafik pengujian eksperimental lanjut	51

DAFTAR TABEL

2.1	Detil masukan pekerjaan pada sistem penjadwalan	10
2.2	Detil pekerjaan setelah selesai dijadwalkan	10
2.3	Detil masukan pekerjaan pada sistem penjadwalan	11
2.4	Detil pekerjaan setelah selesai dijadwalkan	12
3.1	Tabel pekerjaan	13
3.2	Tabel skenario meng-input detil proses	20
3.3	Tabel skenario melakukan penjadwalan	21
5.1	Hasil pengujian fungsional	39
5.2	Tabel pengujian eksperimental perangkat lunak (kondisi : Tidak ada <i>miss</i> dan <i>idle</i> .)	42
5.3	Tabel pengujian eksperimental perangkat lunak (kondisi : Ada <i>miss</i> dan tidak ada <i>idle</i> .)	43
5.4	Tabel pengujian eksperimental perangkat lunak (kondisi : Tidak ada <i>miss</i> dan ada <i>idle</i> .)	44
5.5	Tabel pengujian eksperimental perangkat lunak (kondisi : Ada <i>miss</i> dan ada <i>idle</i> .)	45
5.6	Tabel pengujian eksperimental perangkat lunak (kondisi : Mode <i>real time</i> aktif dan ada <i>miss</i> .)	47

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Di zaman modern ini, semakin banyak pekerjaan yang perlu dijadwalkan dengan program komputer. Masalah yang dapat terjadi adalah jumlah pekerjaan yang dikerjakan terkadang melebihi batas kemampuan dari program komputer itu sendiri. Oleh karena itu, dibutuhkan sebuah penjadwalan pada komputer agar dapat melayani permintaan yang ada. Salah satu contoh dari implementasi teknologi penjadwalan ini adalah sistem waktu nyata atau *Real Time System*. Sistem ini sangat diperlukan untuk berbagai bidang, contoh implementasinya adalah pada sistem pengontrol pesawat terbang. Jika sistem tersebut terlambat untuk mengerjakan sebuah aksi, maka akan membahayakan semua kru dan penumpang di dalam pesawat tersebut. Contoh tersebut menunjukkan pentingnya *Real Time System* pada jaman sekarang ini, dimana sistem harus menghasilkan respon yang tepat dalam batas waktu yang telah ditentukan. Oleh karena itu, pada skripsi ini dibuat sebuah perangkat lunak penjadwalan dengan tujuan untuk mensimulasikan bagaimana sistem *real time* tersebut bekerja.

Pada skripsi ini, penjadwalan diselesaikan dengan menggunakan algoritma *earliest deadline first*, dimana algoritma ini secara garis besar mengurutkan proses yang masuk berdasarkan batas waktu penyelesaiannya *deadline*. Jika proses melewati batas waktu penyelesaiannya, ada 2 buah kemungkinan, yang pertama adalah penjadwalan dilanjutkan dan yang kedua adalah penjadwalan berhenti secara menyeluruh.

1.2 Rumusan Masalah

Masalah yang diangkat dalam skripsi ini adalah :

1. Bagaimana menganalisis algoritma *Earliest Deadline First* ?
2. Bagaimana merancang perangkat lunak dengan menggunakan algoritma *Earliest Deadline First* ?
3. Bagaimana mengembangkan perangkat lunak dengan menggunakan algoritma *Earliest Deadline First* ?

1.3 Tujuan

Tujuan yang ingin dicapai dari penulisan skripsi ini adalah :

1. Menganalisis algoritma *Earliest Deadline First*.
2. Merancang perangkat lunak dengan menggunakan algoritma *Earliest Deadline First*.
3. Mengembangkan perangkat lunak dengan menggunakan algoritma *Earliest Deadline First*.

1.4 Batasan Masalah

Pada skripsi ini, dibuat perangkat lunak yang mendemokan sebuah CPU (*Central Processing Unit*) yang ditugaskan untuk mengerjakan serangkaian pekerjaan yang diberikan oleh pengguna melalui input-input berupa *Burst Time*, *Arrival Time*, dan *Deadline* dari masing-masing pekerjaan. Adapun beberapa batasan masalah yang dibuat yaitu :

1. Jenis eksekusi dari perangkat lunak ini adalah *offline scheduling* yang berarti perangkat lunak ini tidak dapat menerima masukan ketika sedang bekerja (dalam hal ini adalah melakukan penjadwalan), melainkan menerima detail input pekerjaan sebelum penjadwalan dijalankan.
2. Pergantian proses/pekerjaan yang dieksekusi pada saat penjadwalan sedang berlangsung tidak membutuhkan waktu. Pergantian ini biasa disebut dengan *context switching*, dimana CPU (*central processing unit*) akan menyimpan keadaan proses yang sedang dieksekusi ke dalam memori, sehingga dapat diakses kembali ketika proses/pekerjaan tersebut kembali dieksekusi pada waktu mendatang.

1.5 Metodologi

Metodologi yang digunakan dalam skripsi ini adalah :

1. Studi Literatur
Mempelajari teori-teori yang berhubungan dengan *Real Time System* dan algoritma *Earliest Deadline First*, yaitu definisi *real-time system* dan *scheduling*, cara kerja algoritma *earliest deadline first*, komponen pekerjaan (*arrival time*, *burst time*, dan *deadline*), dan *output* dari penjadwalan yang sudah dilakukan (*gantt chart*, *average turnaround time*, dan *average waiting time*).
2. Analisis Kebutuhan dan Sistem
Menganalisis perangkat lunak serupa, yaitu perangkat lunak penjadwalan *Real Time* dengan menggunakan algoritma *Shortest Remaining Time First* untuk mempelajari apa saja yang harus dibuat, lalu setelah itu menganalisis kebutuhan perangkat lunak yang dibuat yaitu penjadwalan *Real Time* dengan menggunakan algoritma *Earliest Deadline First* . Kemudian membuat beberapa diagram untuk menjelaskan perangkat lunak tersebut yaitu diagram *Use Case*, dan diagram kelas.
3. Perancangan Perangkat Lunak
Perancangan diagram kelas lengkap, diagram *sequence* dan antarmuka pengguna dengan memperhatikan hasil analisis yang ada.
4. Pengembangan Perangkat Lunak
Melakukan pengkodean perangkat lunak dengan menggunakan hasil rancangan perangkat lunak yang telah dibuat dan dianalisis.

5. Pengujian Perangkat Lunak

Melakukan pengujian untuk melihat keabsahan dan kebenaran dari perangkat lunak yang telah dibuat. Pengujian dilakukan secara fungsional, dimana penulis sebagai *tester* memfokuskan pengujian pada masukan dan keluaran dari perangkat lunak yang ada. Pengujian juga dilakukan secara eksperimental untuk memastikan bahwa perangkat lunak dapat menerima dan memproses masukan yang keluarannya bervariasi dengan benar. Pengujian eksperimental lanjut juga dilakukan untuk memastikan bahwa perangkat lunak dapat berjalan dengan baik dengan *test case* yang berjumlah ratusan atau ribuan.

6. Pelaporan Hasil Perangkat Lunak

Melaporkan hasil dari perangkat lunak yang sudah dirancang. Pelaporan ini mencakup demo dari perangkat lunak serta hasil dari pengujian perangkat lunak. Banyaknya pekerjaan yang diperlukan untuk melakukan pengujian eksperimental lanjut juga dilaporkan.

1.6 Sistematika Pembahasan

Sistematika pembahasan dari skripsi ini adalah sebagai berikut :

1. Bab 1 berisi tentang latar belakang, rumusan masalah, batasan masalah, dan sistematika pembahasan pada skripsi ini.
2. Bab 2 berisi tentang dasar teori dari perangkat lunak yang dikembangkan. Teori yang dijelaskan adalah cara kerja dan jenis – jenis sistem penjadwalan, Definisi *real time system* dan *scheduling*, *gant chart*, algoritma *earliest deadline first*, dan contoh penjadwalan dengan menggunakan algoritma *earliest deadline first*.
3. Bab 3 berisi tentang analisis dari algoritma dan perangkat lunak sejenis.
4. Bab 4 berisi tentang perancangan perangkat lunak yang terdiri dari rancangan algoritma *earliest deadline first*, rancangan antarmuka pengguna, rancangan diagram aktivitas, dan rancangan diagram kelas.
5. Bab 5 berisi tentang pengujian dan implementasi dari perangkat lunak yang telah dikembangkan. Pengujian yang dilakukan adalah pengujian fungsional dan eksperimental.
6. Bab 6 berisi tentang kesimpulan dan saran.