

IMPLEMENTASI *ADVANCED CALL DISTRIBUTIONS* (ACD) PADA INBOUND CALLS BERBASIS VoIP

Khusnul Sholikah, Mike Yuliana, ST MT, Arifin, ST.MT
Jurusan Teknik Telekomunikasi - Politeknik Elektronika Negeri Surabaya
Institut Teknologi Sepuluh Nopember (ITS) Surabaya
Kampus PENS-ITS, Keputih, Sukolilo, Surabaya.
Telp : +62+031+5947280; Fax. +62+031+5946011
Email : khusnul.sholikah@gmail.com

Abstrak

Call Centre merupakan suatu tempat informasi yang terpusat digunakan dengan tujuan menerima panggilan masuk dan memberi informasi terkait kepada pelanggan. Sekarang sudah banyak perusahaan di luar negeri mengembangkan *Call Centre* yang dihubungkan dengan jaringan komputer dan LAN. Teknologi ini disebut jaringan VoIP (Voice over Internet Protocol).

Pada proyek akhir ini dibuat suatu sistem yang dapat mendistribusikan panggilan telepon masuk dari seorang pelanggan kepada sebuah call center. Sistem ini nantinya akan menampung semua panggilan telepon masuk kedalam suatu IP PBX server. Didalam IP PBX server ini terdapat fitur IVR yang akan menjawab panggilan secara otomatis dan fitur *advance call distributions* yang akan mendistribusikan panggilan kepada seluruh agent yang aktif.

Dari hasil pengujian didapatkan sistem IVR telah berhasil dengan baik, dimana rata – rata waktu eksekusi dalam mengakses sebuah pilihan menu yaitu 4-5 detik. Untuk mengakses informasi layanan yang disediakan membutuhkan waktu 7- 8 detik. Pada implementasi sistem ACD dengan empat metode diperoleh waktu eksekusi rata – rata untuk masing – masing metode yaitu metode *Round Robin* sebesar 7.45 detik, metode *Fewest Calls* sebesar 2.2 detik, metode *Round Robin Memory* sebesar 6.8 detik dan yang terakhir metode *Least Recent* sebesar 2 detik. Dari empat metode ACD tersebut, metode *Least Recent* mempunyai waktu eksekusi tercepat dibanding ketiga metode lainnya. Hasil pengujian responden didapat sebanyak 60% memberi penilaian bagus untuk sistem yang telah dibuat dan sebanyak 30% dari responden memilih metode *Least Recent* sebagai metode yang paling bagus untuk diterapkan dalam *Call Center*.

Kata kunci : *Call Centre*, IP PBX server, *Advance Call Distributions*, *Round Robin*, *Least Recent*, *Fewest Calls*, *RRMemory*.

1. PENDAHULUAN

Saat ini setiap orang menginginkan cara pengaksesan sistem informasi yang cepat dan mudah. Salah satu cara yang bisa dilakukan adalah dengan penambahan layanan IVR pada *call centre*. Dengan penambahan layanan tersebut, maka masyarakat dapat mengakses informasi tanpa perlu berhubungan langsung dengan operator, sedangkan manfaat bagi operator sendiri adalah tugasnya menjadi lebih ringan karena tidak perlu semua pertanyaan dijawab oleh operator.

Pada tugas akhir ini, akan dibuat sebuah sistem *call centre* yang dilengkapi dengan fitur ACD dan pengintegrasian dengan AMI (Asterisk Manager Interface). Pemberian fitur ACD bertujuan untuk mengatur *user* agar di distribusikan secara otomatis menuju *agent* yang tepat sesuai dengan model yang ditetapkan, dimana pada proyek akhir ini terdapat dua kelompok agent dengan metode yang diterapkan adalah *round robin* dan *fewest calls*. Sedangkan pengintegrasian dengan AMI dapat menghasilkan status terkini dari masing - masing agent.

Dengan dibuatnya IVR server yang telah diintegrasikan dengan fitur ACD pada masing-masing *agent*, diharapkan dapat diterapkan untuk menjaga kualitas pelayanan *call center*, baik bagi *user* maupun *agent*, sehingga menunjukkan efektivitas dan perbaikan efisiensi pelayanan *call center*.

2. Teori Penunjang

a. Asterisk

Asterisk, yang merupakan salah satu sistem server PBX open source, saat ini juga mendukung jangkauan yang luas dari protokol VOIP mencakup SIP, MGCP dan H.323. Asterisk dapat beroperasi dengan kebanyakan telepon SIP, seolah-olah sebagai gateway antara IP telepon dan PSTN.[1] Developer Asterisk juga telah mendesain protokol baru, yaitu Inter-Asterisk eXchange, untuk melakukan efisiensi panggilan trunking antara banyak Asterisk PBX. Beberapa telepon memberi dukungan terhadap protokol IAX,

yaitu protokol yang secara langsung berkomunikasi dengan server Asterisk.[2]

b. Asterisk for Java

Asterisk for Java

Paket *Asterisk for java* terdiri dari satu set *class* Java yang memungkinkan untuk membuat aplikasi java yang terintegrasi dengan *Asterisk PBX server*. *Asterisk for java* mendukung masing – masing *interface* yang ditetapkan *Asterisk* untuk hal ini. Yaitu protokol FastAGI dan Manager AGI [3].

Asterisk Gateway Interface (AGI)

Asterisk Gateway Interface adalah *interface* yang digunakan untuk menambahkan kemampuan pada *Asterisk* dengan menggunakan beberapa bahasa pemrograman yang mampu dipahami oleh *asterisk*. Beberapa fungsi AGI :

- AGI memungkinkan mengontrol dialplan, yang dipanggil pada *extensions.conf*.
- EAGI memberikan aplikasi untuk mengakses dan mengontrol *channel sounds* sebagai tambahan terhadap interaksi dengan dialplan.
- FastAGI dapat digunakan untuk melakukan pengolahan di suatu *remote machine* melalui suatu koneksi jaringan.
- DeadAGI memberikan kemampuan untuk mematikan *channel* setelah *hang up*.

Implementasi FastAGI akan mendukung semua *commands* yang biasanya tersedia dari *Asterisk*. Implementasi MangerAPI mendukung penerimaan *events* dari *Asterisk server* misal: *call progress*, keadaan *channel* dan mengirimkan *action* ke *Asterisk* misal : *agent login/ logoff, start/stop* rekaman suara.

Asterisk Manager Interface (AMI)

Manager API merupakan cara lain untuk berkomunikasi dengan server *Asterisk*. Manager API terdiri dari 3 konsep : *Action*, *Responses* dan *Events*. *Action* dapat dikirim ke *Asterisk* dan menginstruksikan untuk melakukan sesuatu.

Events dikirim oleh *Asterisk* tanpa berhubungan langsung dengan *Action* yang mengirimkan aplikasi. *Events* akan mengirimkan informasi yang terdapat dalam *Asterisk*.

Asterisk manager mengijinkan kepada client untuk terhubung dengan hal *asterisk* , *command* dan membaca *events* yang diterima melalui jalur TCP/IP.

c. IVR

Interactive Voice Response merupakan sistem komputer yang berinteraksi dengan telepon pemanggil, dimana pemanggil membirakan input terhadap sistem dengan cara menekan keypad pada telepon (DTMF *dual-tone multi-frequency*) atau dengan mengatakan sesuatu (bahasa yang natural untuk dikenali) [4].

Prinsip dasar dari semua sistem IVR adalah bagaimana pemanggil membaca menu dan memilih pilihan dari menu untuk melakukan tindakan, atau sebagai alternatif memasukkan informasi (pada format nomor melalui penekanan *keypad*).

d. ACD (*Automatic Call Distribution*)

Automatic call distribution (ACD) merupakan jantung sistem *call centre*, yang berfungsi melakukan penanganan terhadap sejumlah besar *incoming call* dan mendistribusikannya ke sekelompok *agent* secara otomatis sesuai model distribusi yang ditetapkan. Sehingga user tersambung dengan cepat (*call processing*) ke *agent* atau kelompok *agent* sesuai tingkat kualifikasinya (*skill level*). ACD menjaga agar panggilan terdistribusi secara merata ke semua *agent* sesuai waktu kerja dan beban kerjanya (*load balancing*). Serta menjaga agar waktu antrian *call* lebih singkat dan memberikan *treatment* yang sesuai dengan kapasitas antrian yang disediakan (*waiting time threatment*)[5].

Metode ACD

- *Round Robin*

Antrian panggilan akan berputar kepada seluruh *agent* yang sedang tidak menerima panggilan hingga menemukan satu dari *agent* yang siap untuk menerima panggilan.[6] Juga karena *roundrobin* akan selalu mulai dengan *agent* pertama yang terdapat pada antrian, metode ini cocok hanya pada lingkungan yang menginginkan *agent* dengan rangking tertinggi/ *agent* berkompeten untuk menghadapi semua panggilan kecuali kalau *agent* lagi sibuk, di situasi ini *agent* dengan rangking rendah yang akan mendapat giliran terakhir untuk menerima panggilan.

- *Fewest Calls*

Panggilan akan ditujukan kepada *agent* yang telah menerima panggilan dalam jumlah sedikit. Ini akan mempertimbangkan hanya pada banyak panggilan yang mereka terima.

- *Least Recent*

Panggilan akan diberikan kepada *agent* yang tidak menerima panggilan selama periode terlalu pada suatu waktu.

- *RRMemory*

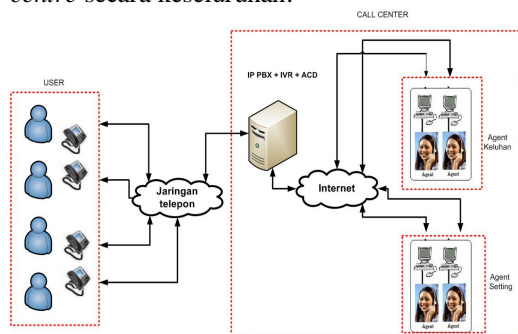
Siklus antrian akan melalui daftar antrian *agent*, yang melacak anggota terakhir

menerima panggilan. Saat terdapat panggilan masuk lagi, *Round Robin* akan memulai dari *agent* berikutnya yang akan menerima. Hal ini menjamin bahwa siklus penerimaan panggilan akan lebih adil.

3. Metodologi

a. Perancangan Sistem

Pada sistem dilakukan integrasi antara jaringan IP dan jaringan telepon analog, sehingga penelpon yang terhubung dari jalur PSTN melalui telepon analog dan *handphone* akan terhubung dengan IP PBX melalui jaringan internet menggunakan ekstensi seperti IP phone atau *softphone* yang dilengkapi dengan headset. Berikut diagram sistem *call centre* secara keseluruhan.



Gambar 1 Arsitektur Sistem

Gambar 1 merupakan arsitektur sistem yang dirancang dalam tugas akhir ini. Arsitektur tersebut terdiri dari sebuah IP PBX server yang memiliki fitur sebagai IVR dan ACD. Sedangkan ACD berfungsi sebagai perutean atau penyambung antara seorang *user* dengan *agent* apabila seorang *user* menginginkan komunikasi dengan *agent* tersebut. Dimana *agent* akan dikelompokkan menjadi menjadi *agent* keluhan dan *agent* setting

Cara kerja dari sistem tersebut diawali dari *user* yang melakukan panggilan kemudian setelah sampai pada sistem call centre akan diterima oleh IP PBX yang dalam hal ini IP PBX juga sebagai IVR server. sehingga user akan menerima jawaban otomatis yang selalu berbunyi ketika *user* menghubungi *call centre*, selain itu didalam IP PBX juga dilengkapi dengan ACD yang akan menghubungkan *user* dengan *agent* sesuai model distribusinya. Pada proyek akhir ini, penulis akan mengerjakan bagian IVR server yang di integrasikan dengan ACD yang menggunakan bahasa pemrograman Java. Perancangan sistem pada proyek akhir ini terdiri dari:

1. Perancangan IP PBX server untuk layanan *Call centre*
2. Perancangan aplikasi IVR berbasis Java

3. Perancangan aplikasi ACD metode *Round Robin*, *Fewest Calls*, *Round Robin Memory* dan *Least Recent*.

4. Hasil Pengukuran dan Analisa

a. Pengujian dan Analisa Ketepatan Digit DTMF

Pengujian ketepatan digit dengan informasi yang diberikan IVR merupakan ketepatan digit yang ditekan penelpon dengan informasi yang diharapkan. Pada pengujian kali ini dilakukan penekanan digit pada masing – masing layanan informasi yang dimiliki IVR server.

Tabel 1. Hasil Pengujian Ketepatan Digit

Urutan digit yang ditekan	Jenis layanan yang diakses	Status	
		Berhasil	Gagal
1	Pilihan Informasi	✓	
1-1	Informasi tipe modem GSM beserta harga	✓	
1-2	Informasi tipe modem CDMA beserta harga	✓	
2	Pilihan informasi	✓	
2-1	Informasi lokasi daerah jabodetabek	✓	
2-2	Informasi lokasi daerah surabaya	✓	
2-3	Informasi lokasi daerah yogyakarta	✓	
3	Operator Bantuan <i>Setting</i>	✓	
4	Operator Kritik dan Saran	✓	
9	Kembali ke Menu Utama	✓	
6	Tidak dapat mengakses layanan informasi		✓
7	Tidak dapat mengakses layanan informasi		✓
8	Tidak dapat mengakses layanan informasi		✓
#	Tidak dapat mengakses layanan informasi		✓
*	Tidak dapat mengakses layanan informasi		✓
0	Keluar dari layanan	✓	

Dari pengujian ketepatan digit dengan informasi yang diberikan IVR, menunjukkan bahwa IVR telah bekerja dengan benar, hal ini ditandai dengan keberhasilan memainkan file suara sesuai dengan jenis layanan yang diakses berdasarkan urutan digit yang dimasukkan. Sedangkan gagal disini berarti, apabila digit yang ditekan tidak ada dalam sistem informasi maka telepon akan terputus (Hangup).

b. Pengujian dan Analisa Ketepatan Implementasi layanan ACD

• Pengujian Metode *Round Robin*

Dari pengujian metode *round robin* dapat dianalisa bahwa panggilan dapat didistribusikan secara baik berdasarkan prioritas masing – masing *agent*. *agent* yang

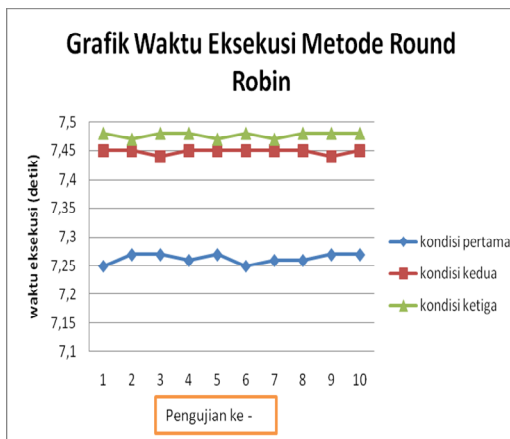
berada pada prioritas pertama adalah *agent* Khusnul, *agent* Zai pada prioritas kedua, dan *agent* Firdaus pada prioritas ketiga. Berikut tabel hasil pengujian ketepatan metode *Round Robin*:

Tabel 2. Hasil pengujian Ketepatan Metode *Round Robin*

Nomor yang ditelpon	Digit yang ditekan	Kondisi <i>agent</i> yang Login			<i>Agent</i> yang berdering
		<i>Agent</i> Khusnul	<i>Agent</i> Zai	<i>Agent</i> Firdaus	
300	3	On Hook	On Hook	On Hook	Khusnul
300	3	Off Hook	On Hook	On Hook	Zai
300	3	Off Hook	Off Hook	On Hook	Firdaus
300	3	On Hook	On Hook	Off Hook	Khusnul
300	3	Off Hook	On Hook	Off Hook	Zai
300	3	On Hook	Off Hook	On Hook	Khusnul
300	3	On Hook	Off Hook	Off Hook	Khusnul
300	3	On Hook	On Hook	Off Hook	Khusnul

Pengujian Waktu Akses Metode *Round Robin* Kondisi Pertama, Kedua dan Ketiga

Pengujian kondisi pertama dilakukan pada saat satu kelompok operator yang terdiri dari 3 *agent* sedang on hook, dan pada saat itu ekstensi *agent* khusnul berada yang akan berdering terlebih dahulu. Dari 10 kali percobaan diperoleh rata-rata waktu akses 7.26 detik. Pengujian kondisi diperoleh rata-rata waktu eksekusi 07.45 detik. Pengujian kondisi ketiga diperoleh rata-rata waktu akses 07.48 detik. Berikut grafik waktu eksekusi untuk kondisi *agent* prioritas pertama, kedua dan ketiga:



Gambar 2. Grafik Waktu Eksekusi Metode *Round Robin*

- **Pengujian Metode *Fewest Calls***

Dari pengujian metode *fewest calls* dapat dianalisa bahwa panggilan disalurkan berdasarkan jumlah panggilan yang diterima oleh *agent*. Pada kelompok operator pengaduan keluhan, kritik dan saran, *agent* yang berada pada prioritas pertama adalah *agent* yang mempunyai jumlah panggilan paling sedikit, yaitu *agent* Dipta, *agent* Rara

berada pada prioritas kedua, dan *agent* Nanda berada pada prioritas ketiga. Berikut gambar tabel hasil pengujian :

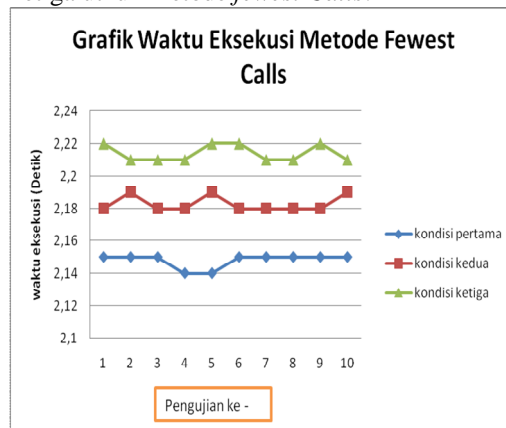
Tabel 3. Hasil Pengujian Ketepatan Metode *Fewest Calls*

Nomor yang ditelpon	Digit yang ditekan	Kondisi <i>agent</i> dengan jumlah panggilan paling sedikit			<i>Agent</i> yang berdering
		Rara (4)	Nanda (7)	Dipta (2)	
300	4	On Hook	On Hook	On Hook	Dipta
300	4	OnHook	On Hook	Off Hook	Rara
300	4	OffHook	On Hook	Off Hook	Nanda
300	4	On Hook	On Hook	On Hook	Dipta
300	4	On Hook	On Hook	Off Hook	Rara
300	4	On Hook	Off Hook	On Hook	Dipta
300	4	OffHook	On Hook	OffHook	Nanda
300	4	On Hook	Off Hook	On Hook	Dipta

Pengujian Waktu Eksekusi Metode *Fewest Calls* Kondisi Pertama, Kedua dan Ketiga

Pengujian kondisi pertama dilakukan pada saat satu kelompok operator yang terdiri dari *agent* sedang on hook, dan pada saat itu *agent* dipta yang akan berdering terlebih dahulu. Dari 10 kali percobaan diperoleh rata-rata waktu eksekusi 2.15 detik. Pengujian kondisi kedua diperoleh rata-rata waktu Eksekusi 2.19 detik. Pengujian kondisi diperoleh rata-rata waktu Eksekusi 2.21 detik.

Berikut grafik waktu eksekusi untuk kondisi *agent* prioritas pertama, kedua dan ketiga untuk metode *fewest Calls*:



Gambar 2. Grafik Waktu Eksekusi Rata-Rata Metode *Fewest Calls*

- **Pengujian Metode *Round Robin Memory***

Pengujian ACD metode *round robin memory* dilakukan pada saat penelpon terhubung dengan operator yang menangani bantuan *setting* yaitu dengan menekan digit 3. Dengan metode *Round Robin Memory*, panggilan dapat didistribusikan kepada antrian *agent*. *Agent* yang berada pada antrian pertama adalah *agent* Dani, *agent* Beni pada urutan kedua, dan *agent* Hasbi pada antrian

ketiga. Berikut tabel hasil pengujian ACD metode *round robin memory*..

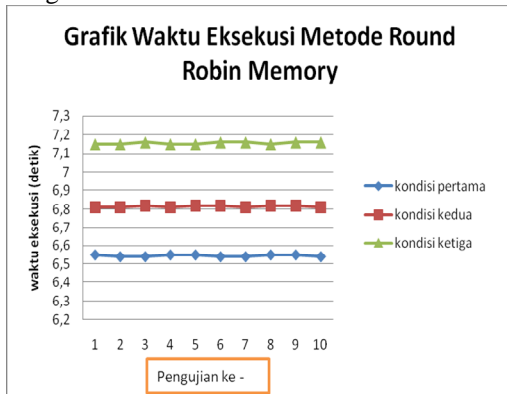
Tabel 4. Hasil Pengujian Ketepatan Metode *Round Robin Memory*

Nomor yang ditelpon	Digit yang ditekan	Kondisi <i>agent</i> yang Login			<i>Agent</i> yang berdering
		<i>Agent</i> Dani	<i>Agent</i> Beni	<i>Agent</i> Hasbi	
301	3	On Hook	On Hook	On Hook	Dani
301	3	OffHook	On Hook	On Hook	Beni
301	3	OffHook	OffHook	On Hook	Hasbi
301	3	On Hook	On Hook	OffHook	Dani
301	3	OffHook	On Hook	On Hook	Beni
301	3	On Hook	OffHook	On Hook	Hasbi
301	3	On Hook	On Hook	OffHook	Dani

Pengujian Waktu Eksekusi Metode *fewest calls* Kondisi Pertama, Kedua dan Ketiga

Pengujian kondisi pertama dilakukan pada saat satu kelompok operator yang terdiri dari 3 *agent* sedang on hook, dan agent dani berada dalam antrian pertama sehingga panggilan akan ditujukan kepada agent Dani. Dari 10 kali percobaan diperoleh rata-rata waktu eksekusi 06.54 detik. Pengujian kondisi kedua diperoleh rata-rata waktu eksekusi 06.815 detik. Pengujian kondisi ketiga diperoleh rata-rata waktu Eksekusi 07.15 detik.

Berikut grafik waktu eksekusi untuk kondisi *agent* prioritas pertama, kedua dan ketiga:



Gambar 4. Grafik Waktu Eksekusi Metode *Round Robin Memory*

- **Pengujian Metode *Least Recent***

Pengujian ACD metode *least Recent* dilakukan pada saat penelpon terhubung dengan operator yang menangani pengaduan keluhan, kritik dan saran yaitu dengan menekan digit 4. Pada metode *Least Recent* panggilan dapat didistribusikan menuju *agent* yang tepat. *Agent* yang berada pada prioritas pertama adalah *agent* yang mempunyai waktu tunggu / waktu idle terlama , yaitu *agent* Ais , *agent* Ayu berada pada prioritas kedua, dan

agent Asri berada pada prioritas ketiga. Berikut tabel hasil pengujian ACD metode *Least Recent* .

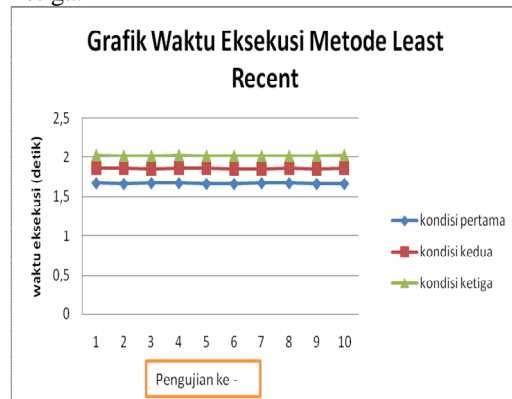
Tabel 5. Hasil Pengujian Ketepatan Metode *Least Recent*

Nomor yang ditelpon	Digit yang ditekan	Kondisi <i>agent</i> dengan waktu tunggu terlama			<i>Agent</i> yang berdering
		Asri (01:23)	Ais (01:56)	Ayu (01:44)	
301	4	On Hook	On Hook	On Hook	Ais
301	4	OnHook	OffHook	On Hook	Ayu
301	4	On Hook	OffHook	Off Hook	Asri
301	4	On Hook	On Hook	On Hook	Ais
301	4	On Hook	On Hook	Off Hook	Ais
301	4	On Hook	Off Hook	On Hook	Ayu
301	4	OnHook	Off Hook	OffHook	Asri
301	4	OffHook	Off Hook	On Hook	Ayu

Pengujian Waktu Eksekusi Metode *Least Recent* Kondisi Pertama, Kedua dan Ketiga

Pengujian kondisi pertama dilakukan pada saat satu kelompok operator yang terdiri dari semua *agent* sedang on hook. Dari 10 kali percobaan diperoleh rata-rata waktu akses 1.665 detik. Pengujian kondisi kedua diperoleh rata-rata waktu eksekusi 1.855 detik. Pengujian kondisi ketiga diperoleh rata-rata waktu eksekusi 2.03 detik

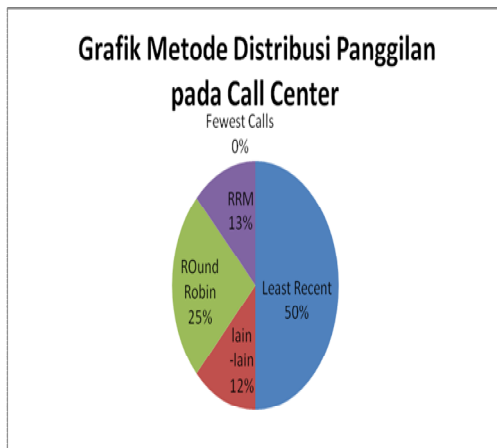
Berikut grafik waktu eksekusi untuk kondisi *agent* prioritas pertama, kedua dan ketiga:



Gambar 5. Grafik Waktu Eksekusi Metode *Least Recent*

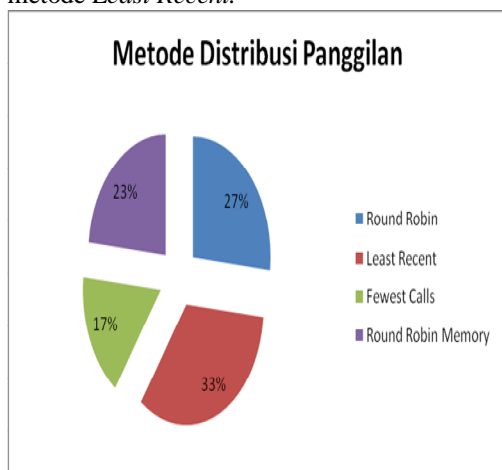
- **Pengujian Kualitatif Sistem**

Untuk menunjang ketepatan sistem call center yang telah dibuat, maka perlu diadakan survey ke beberapa call center untuk lebih mengetahui lebih detail tentang sistem yang dikembangkan oleh Call Center tersebut. Nantinya akan dibandingkan dengan sistem yang telah dibuat. Apakah sistem yang telah dibuat sudah sesuai dengan kondisi nyata yang terjadi pada dunia Call Center. Berikut merupakan grafik prosentasi hasil survey dari Call Center diatas:



Gambar 6. Grafik Prosentase Penggunaan Metode ACD

Berdasarkan hasil survey dari beberapa perusahaan dan hasil kuesioner, maka dapat ditarik sebuah analisa bahwa dari masing masing metode ACD yaitu *Round Robin*, *Least Recent*, *Fewest Calls* dan *Round Robin Memory*, yang paling banyak dipilih dan dipakai oleh perusahaan *Call Center*, adalah metode *Least Recent*.



Gambar 7. Metode Distribusi Panggilan

Metode ini dirasa paling cocok untuk diterapkan dalam *Call Center* skala besar. Pada saat pengujian sistem yang telah dibuat, waktu eksekusi untuk metode *Least Recent* tercepat daripada ketiga metode yang lain yaitu hanya 2 detik untuk setiap kondisi.

Dari hasil survey, pengujian responden dan pengujian sistem diperoleh bahwa metode *Least Recent* yang paling cocok digunakan dalam dunia *Call Center*. Ini ditinjau dari segi waktu eksekusi pengujian sistem, hasil survey lapangan dan hasil pengujian responden.

5. Kesimpulan

Setelah dilakukan pengujian dan analisa pada sistem IVR server dan *Automatic Call*

Distribution untuk layanan *Call centre* dapat diperoleh kesimpulan sebagai berikut:

- Sistem IVR telah berhasil dijalankan, dimana rata – rata waktu eksekusi dalam mengakses sebuah pilihan menu yaitu 4-5 detik. Untuk mengakses informasi layanan yang disediakan membutuhkan waktu 7- 8 detik. Waktu eksekusi ini diperoleh saat user menekan digit yang diinginkan sampai dengan informasi diberikan.
- Implementasi empat buah metode ACD yang telah dibuat, telah berjalan dan diperoleh waktu eksekusi rata – rata untuk masing – masing metode yaitu metode *Round Robin* sebesar 7.45 detik, metode *Fewest Calls* sebesar 2.2 detik, metode *Round Robin Memory* sebesar 6.8 detik dan yang terakhir metode *Least Recent* sebesar 2 detik. Dari empat metode ACD tersebut, metode *Least Recent* mempunyai waktu eksekusi tercepat dibanding ketiga metode lainnya.
- Dari hasil survey dan pengujian responden, sebesar 50 % dari enam buah perusahaan menggunakan metode *Least Recent* sebagai metode distribusi panggilannya. Hasil pengujian responden didapat sebanyak 60% memberi penilaian bagus untuk sistem yang telah dibuat dan sebanyak 30% dari responden memilih metode *Least Recent* sebagai metode yang paling bagus untuk diterapkan dalam *Call Center*.

6. DAFTAR PUSTAKA

- Jim Van Meggelen, dkk, “*Asterisk The Future of Telephony*, Second Edition”, 2007.
- Goncalves E. Flavio: “*Asterisk PBX Configuration Guide*”, Digium Inc, 2006.
- Library API – Reference at <http://asterisk-java.org/>
- Yuliana, Mike “*Interactive Voice Response*”, Buku Ajar Teori Jaringan Telephony, PENS-ITS, Surabaya, 2009.
- Anugrah, Andi, “*Call center Memilih ACD (Automatic Call Distributions)*”, Volume III no. 22, e-Business Magazine, 2004.
- <http://asterisk-java.org>