Implementasi *Goal Oriented Requirement Engineering* Menggunakan *Knowledge Acquisition in autOmated Spesification* Untuk Pengelolaan Administrasi Kepolisian Sindangkerta

Yusiani Kamalia[1], Sri Widowati[2], Jati H. Husen[3]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[1]yusianik@students.telkomuniversity.ac.id, [2]sriwidowati@telkomuniversity.ac.id,
[3]jatihusen@telkomuniversity.ac.id

Abstrak

Permasalahan dari sebuah kegagalan pengembangan perangkat lunak salah satunya berasal dari kesalahan yang terjadi pada proses pendefinisian kebutuhan (*requirement*). Adapun salah satu metode untuk menyelesaikan permasalahan tersebut adalah dengan menggunakan metode yang terdapat dalam GORE (*Goal Oriented Requirement Engineering*), GORE merupakan salah satu metode dalam *Requirement Engineering* yang dimana memusatkan perhatian pada tujuan (*goals*) yang ingin dicapai oleh pengguna. Didalam GORE terdapat salah satu metode yaitu KAOS (*Knowledge Acquitition in autOmated Spesification*) metode ini adalah metode yang mementingkan tujuan organisasi. Adapun hasil dari validasi *requirements* berdasarkan indikator *good requirements* pada penelitian ini yang digunakan adalah *correct* dan *clear*. Dari hasil pengujian berdasarkan masing-masing *requirements user* didapatkan nilai mean terhadap pengujian faktor *correct* adapun *requirements* yang terspesifikasi dengan benar sebesar 82,12% adapun kekurangannya adalah 17,88% dikarenakan ada beberapa *requirement* yang tidak terspesifikasi dengan detail. Adapun hasil pengujian berdasarkan indikator *clear* menunjukkan sebesar 82,72% dan dikategorikan setuju atau *requirements* tersebut *clear* terhadap *requirements* adapun kekurangannya adalah 17,28% dikarenakan ada beberapa *requirement* yang belum bisa menggambarkan fitur yang akan dibuat dengan jelas.

Kata kunci : *requirement, GORE,* KAOS, administrasi kepolisian

Abstract

One problem with software development problems is that they are excluded from errors that occur in the process of defining requirements (requirements). One method to solve this problem is to use a method that is in the GORE (Goal Oriented Requirement Engineering), GORE is one of the methods in Requirement Engineering that focuses on the goals (objectives) that the user wants to find. In GORE is one method, namely KAOS (Knowledge Actuition in Automatic Specifications) this method is a method that emphasizes organizational goals. From the results of the validation requirements based on good requirements indicators in this study, it is used correctly and clearly. From the test results based on each user requirement obtaining a meaningful value for the correct factor testing requirements as for correctly specified amounted to 82.12% while the shortcomings were 17.88% because there were several requirements that were not detailed in detail. From 82.72% and categorized as approval or clear requirements for requirements, while the drawback is 17.28%, there need to be some features that can not be imagined features that will be made clearly.

Keyword : requirement, GORE, KAOS, police administration

1. Pendahuluan

Latar Belakang

Requirement merupakan suatu keadaan yang harus dipenuhi atau dimiliki oleh suatu sistem untuk memenuhi sebuah standar atau spesifikasi dari apa yang harus diimplementasikan[1]. *Requirement* mempunyai peranan penting dalam pengembangan perangkat lunak dikarenakan apabila terjadi kesalahan dalam penentuan *requirement* maka akan menyebabkan perangkat lunak yang dihasilkan tidak berkualitas karena tidak memenuhi spesifikasi dan ekspetasi pengguna[2]. Disisi lain, apabila terdapat asumsi kebutuhan yang tidak sama maka akan menyebabkan kesalahpahaman antara pengguna dengan tim pengembang yang nantinya akan menimbulkan masalah pada saat tahap *design*, *code* dan pengujian perangkat lunak. Adapun masalah lain yang sering terjadi dalam pendefinisian kebutuhan yaitu kurangnya keterlibatan pengguna, kebutuhan yang ambigu, tidak ada prioritas, membangun fungsionalitas yang tidak sama sekali digunakan dan keinginan yang terus menerus dari pihak pengguna akan tetapi tidak sesuai dengan tujuan yang ingin dicapai (*scope creep*)[3].

Salah satu tahapan dalam proses pengembangan perangkat lunak adalah *Requirement Engineering* (RE). RE merupakan proses yang berorientasi tujuan dengan mengidentifikasi pengguna dan tujuannya termasuk pada faktor-faktor dalam menetapkan spesifikasi dari sebuah perangkat lunak. RE mempunyai tujuan yaitu mendefinisikan perangkat lunak yang akan dikembangkan. RE didasari oleh dua kegiatan utama yaitu *requirements development* dan *requirements managements*. Adapun *requirements development* terdiri dari beberapa sub kegiatan diantaranya adalah *elicitation*, *analysis*, *spesification* dan *validation*[1]. Pada penelitian yang dilakukan berfokus pada *requirements development* yang menghasilkan sebuah *output* yaitu data *requirement.*

Pada RE tradisional hanya berfokus pada sistem hal ini mempunyai kelemahan karena cenderung mengurangi interaksi pengguna dengan proses bisnis yang terjadi. Hal ini dapat menyebabkan terjadinya *gold plating* dan *scope creep* yang nantinya akan berdampak negatif pada saat tahap pembangunan perangkat lunak. Adapun GORE yang berorientasi *goal* dan *agent* lebih menjanjikan dibandingkan dengan menggunakan RE tradisional karena dengan *goal* memungkinkan kelengkapan dan ketepatan yang sesuai dari suatu spesifikasi kebutuhan[3]. *Goals* juga memberikan alasan mengapa sebuah *requirements* harus ada ataupun tidak. Kehadiran GORE ini melengkapi kekurangcukupan pada RE tradisional yang berfokus pada sistem. Namun, GORE dilengkapi dengan memberikan sebuah alasan untuk fungsionalitas sistem dengan menjawab "mengapa" sebuah fungsionalitas tersebut diperlukan[4]. GORE juga dapat meminimalisir terjadinya *gold plating* atau pencantuman *requirement* yang sebetulnya tidak diperlukan tetapi diadakan karena melihat dari tujuan utama yang harus dipenuhi[1]. Disamping itu GORE juga mencegah terjadinya *scope creep* atau keinginan yang terus menerus dari pihak pengguna akan tetapi tidak sesuai dengan tujuan yang ingin dicapai[3].

Adapun salah satu metode dalam GORE yaitu *Knowledge Acquisition in autOmated Spesification* (KAOS)[5]. KAOS merupakan pendekatan yang mementingkan tujuan organisasi[6]. Pemodelan KAOS ini adalah menggambarkan secara hirarki dari *requirements* yang telah diidentifikasi agar lebih mudah dalam melakukan penelusuran (*high traceability*)[7].

Topik dan Batasannya

Berdasarkan latar belakang masalah diatas dapat diidentifikasi bahwa kegagalan pada pengembangan perangkat lunak salah satunya berakar dari kesalahan dalam penentuan kebutuhan sistem. Dalam hal ini salah satu metode yang terdapat dalam GORE yaitu KAOS akan membantu mendifiniskan kebutuhan dengan memusatkan pada tujuan (*goal*) organisasi administrasi kepolisian yang ingin dicapai sebagai dasar untuk menentukan sebuah *requirements* mengapa diperlukan ataupun sebaliknya.

Adapun batasan masalah dalam penelitian ini adalah sebagai berikut :

1. Penerapan metode GORE menggunakan KAOS diterapkan pada pengelolaan administrasi di POLSEK Sindangkerta
2. *Goal* yang ingin dicapai berfokus pada 1 *goal* yaitu meningkatkan pelayanan administrasi.
3. Validasi *requirements* menggunakan skala likert berdasarkan indikator *good requirements* yaitu *correct* dan *clear*.

Tujuan

Tujuan dari penelitian ini adalah mengimplementasikan metode GORE menggunakan KAOS untuk menentukan spesifikasi kebutuhan sistem pengelolaan administrasi di POLSEK Sindangkerta dan melakukan validasi *requirements* berdasarkan indikator *good requirement* yaitu *correct* dan *clear*.

Organisasi Tulisan

Kelanjutan dari penelitian ini yaitu bagian 2 studi terkait dan metodologi yang digunakan, bagian 3 menjelaskan sistem yang dibangun, bab 4 membahas evaluasi sistem yang telah dibangun, dan bab 5 memberikan kesimpulan dari penelitian yang telah dilakukan.

2. Studi Terkait

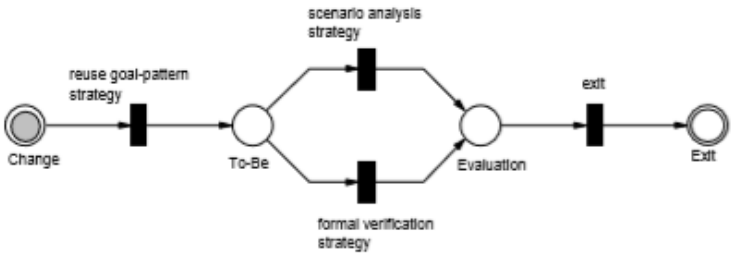
2.1 *Goal Oriented Requirement Engineering* (GORE)

Goal Oriented Requirement Engineering (GORE) ialah sebuah pendekatan didalam RE yang berorientasi pada identifikasi tujuan (*goal*). *Goals* tersebut diubah menjadi sebuah kebutuhan (*requirements*)[8]. GORE mendefinisikan *goal* sebagai tujuan yang harus dicapai melalui kerjasama antara *Agent* dan lingkungannya. Gore mempunyai beberapa kelebihan diantaranya adalah memberikan alasan untuk sebuah *requirements* dan dapat menunjukan struktur yang dapat dipahami yang membantu dalam pembuatan *Software Requirements Document* [9]. Kehadiran GORE ini dikarenakan adanya kekurangcukupan dalam pendekatan tradisional karena pada pendekatan tradisional tersebut berkonsentrasi pada sistem dan tidak mempunyai alasan untuk sebuah keadaan *requirement*. Sedangkan GORE mempunyai sebuah alasan untuk ada dan tidak adanya sebuah *requirement*.

Terdapat beberapa metode yang dapat digunakan untuk mengimplementasikan GORE, diantaranya Deriving Tabular Event-Based Spesifications from goal oriented requirement model (DTEBS), GBRAM (Goal-Based Requirements Analysis Method)[8], AGORA (Attributed Goal-Oriented Requirements Analysis Method)[10], Visual Variability Analysis for goal models (VVA), Goal-Oriented Idea Generation Method (GOIG), Deriving Operational Software Specifications (DOSS), Agent-Based Tactics for goal-oriented requirements elaboration (A-BT), and goal oriented requirement elicitation based on General System Thingking Heuristics (GSTH) selain itu ada[8] juga Non-Functional Requirements Framework (NFR Frameworks), i*/Tropos, *Knowledge Acquisition in autOmated Spesification* (KAOS) maupun Goals-Skilss-Preferences Framework (GSP Framework)[12]. Terdapat juga[13] metode : i*, ConGolog, Albert language, Sedangkan Van Lamsweerd[10], menjelaskan bahwa gabungan NFR, i*, dan Tropos dikembangkan lagi menjadi metode GRL.

2.2 *Knowledge Acquisition in autOmated Specification* (KAOS)

KAOS adalah pendekatan yang menekankan pentingnya memodelkan tujuan organisasi[6]. Didalam Pemodelan KAOS dilakukan untuk proses *elicitation*, *specification*, menganalisis tujuan (*goals*), *requirements*, *scenario* dan tanggung jawab tugas[5]. Kelebihan dari menggunakan pemodelan KAOS ini adalah memberikan gambaran secara hirarki dari kebutuhan yang telah diidentifikasi agar lebih mudah dalam melakukan penulusuran (high traceability)[7]. Adapun cara kerja dari pemodelan KAOS adalah sebagai berikut[6] :



Gambar 2.1 Cara Kerja KAOS[6].

a. *Change* : mengubah proses yang sedang berjalan kemudian dilakukan identifikasi sesuai kebutuhan dan tujuannya.
b. To-Be : mencari karakteristik sistuasi organisasi dimasa yang akan datang.
c. Evaluation : melakukan penilaian terhadap kesesuaian kebutuhan dan tujuannya juga tentang bagaimana kebutuhan organisasi pada masa yang akan datang.

Ada beberapa elemen pada KAOS yaitu[7] :

a. *Goal* : keadaan yang harus dipenuhi atau dapat diterima oleh sistem dalam sebuah kondisi yang ditetapkan[8].
b. *Softgoal* : mendokumentasikan perilaku alternatif dari sistem.
c. *Agent* : komponen aktif dapat berupa manusia, *hardware, software* dan lainnya *yang* mempunyai peran.
d. *Hardgoal* : merupakan suatu fungsionalitas yang harus dimiliki atau disediakan oleh sistem tersebut yang nantinya akan dipecah menjadi sebuah *requirements*.

2.3 *Good Requirements*

Requirement mempunyai peranan penting dalam pengembangan perangkat lunak dikarenakan apabila terjadi kesalahan dalam penentuan *requirement* maka akan menyebabkan perangkat lunak yang dihasilkan tidak berkualitas karena tidak memenuhi spesifikasi dan ekspetasi pengguna[2]. Sebuah kebutuhan perangkat lunak dikatakan baik apabila memenuhi karakteristik seperti berikut ini[14] :

a. *Unambigous* : hanya ada satu interpretasi terhadap sebuah *requirements* document.
b. Testable : sebuah sistem dapat diimplementasikan, sebaiknya dapat dipastikan bahwa sistem tersebut memenuhi requirements awal.
c. *Correct* : spesifikasi yang ditulis adalah benar dan sesuai dengan kebutuhan pengguna.
d. *Clear* : spesifikasi harus tepat untuk menggambarkan fitur yang akan dibangun.
e. *Understandable* : semua calon pengguna sebaiknya dapat mengerti *requirements* seperti yang ditetapkan dalam dokumen.
f. *Feasible* : kebutuhan dikatakan layak jika bisa sesuai dengan teknologi yang ada, sesuai anggaran, dan sesuai dengan waktu dan jadwal.
g. *Necessary* : kebutuhan yang diperlukan adalah kebutuhan yang harus ada untuk memenuhi tujuan sistem, sangat penting untuk pengoperasian sistem, dan jika kebutuhan tidak ada akan menyebabkan kekurangan pada sistem.

2.4 Skala Likert

Skala *likert* digunakan untuk mengukur sikap, persepsi dan pendapat seseorang atau berkelompok terkait kejadian sosial[13]. Dengan menggunakan skala *likert*, maka variabel yang akan diukur dijabarkan menjadi indikator variabel[13]. Indikator tersebut digunakan untuk menyusun pertanyaan-pertanyaan[13]. Pertanyaan tersebut kemudian direspon dalam bentuk skala *likert*. Untuk keperluan analisis data secara kuantitatif, sehingga setiap pilihan jawab itu dapat diberikan skor seperti pada tabel berikut :

Tabel 2.1 Skor Respon Jawaban

No	Respon Jawaban	Skor
1.	Sangat Setuju	5
2.	Setuju	4
3.	Netral	3
4.	Tidak Setuju	2
5.	Sangat Tidak Setuju	1

Pengumpulan data angket dikelola dengan menggunakan rumus skala *likert*, maka perlu diketahui terlebih dahulu jumlah responden yang terlibat dalam kuisoner angket sebesar N orang, dan jumlah skor ideal (kriterium) untuk seluruh item dengan rumus 5X N = jumlah ideal [13]. Berdasarkan data itu maka tingkat persetujuan (TP) terhadap variabel penelitian angket dengan rumus (Total skor/jumlah ideal) X 100% = TP. Dengan catatan total skor diperoleh dari jumlah penilaian masing-masing item skor dari N orang responden[13]. Atau dapat dirumuskan sebagai berikut :

$$TP = \frac{TS}{Skor\ ideal}\ X\ 100\% \quad (1)$$

Dimana :
TP = Nilai persentase
TS = Total skor responden = ∑ skor x responden

Sehingga dari hasil analisis perhitungan tingkat pertujuan yang diperoleh nantinya diukur dalam model interval untuk setiap variable penelitian dalam angket. Adapun untuk menghitung rentan menggunakan rumus sebagai berikut :

$$Nilai\ Rentan = \frac{Nilai\ Presentase\ Terbesar - Nilai\ Presentase\ terkecil}{Jumlah\ titik\ skala}\ X\ 100\% \quad (2)$$

Berdasarkan jumlah responden dan hasil perhitungan nilai rentang diatas, diperoleh interprestasi skor sebagaimana yang terdapat dalam tabel berikut :

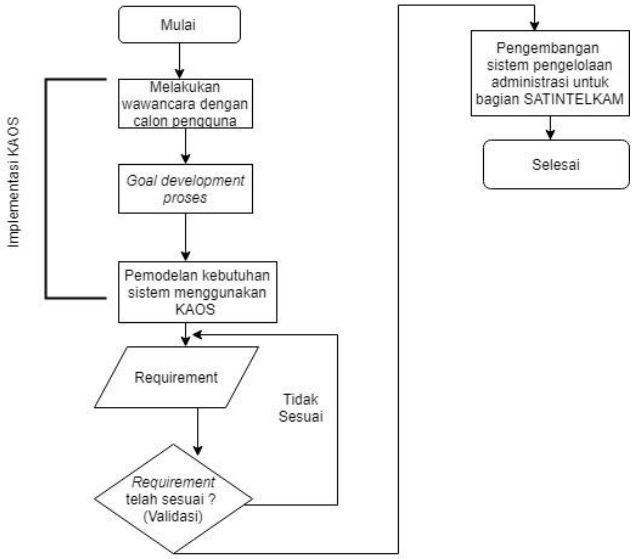
Tabel 2.2 Kategori Interpretasi Nilai

No	Presentase	Kategori Penilaian
1.	< 20% - 36 %	Sangat Tidak Setuju
2.	>36% - 52%	Tidak Setuju
3.	>52%-68%	Netral
4.	>68%-84%	Setuju
5.	>84%-100%	Sangat Setuju

3 Sistem yang Dibangun

3.1 Alur Penelitian

Berikut adalah alur penelitian yang dilakukan pada gambar 3.1 :



Gambar 3.1 Alur Penelitian

3.4 Melakukan Wawancara dengan Calon Pengguna

Langkah awal pada penelitian ini adalah wawancara dengan calon pengguna. Pada tahapan ini, peneliti melakukan observasi secara langsung ke POLSEK Sindangkerta. Observasi ini dilakukan secara singkat melalui suatu pertemuan bersama dengan Kepala Unit SATINTELKAM POLSEK Sindangkerta yaitu Aiptu Wawan Suwanda. Pada observasi ini peneliti mengumpulkan data-data yang berkaitan untuk berlangsungnya penelitian ini peneliti juga melakukan pengamatan proses bisnis yang sedang berjalan di POLSEK Sindangkerta. Setelah melakukan wawancara peneliti mendapatkan pemaparan mengenai kekurangan proses bisnis yang sedang berjalan dan tujuan yang diinginkan oleh pelayanan administrasi kepolisian Sindangkerta.Berdasarkan hasil wawancara yang telah dilakukan kepada Kepala Unit SATINTELKAM POLSEK Sindangkerta dapat simpulkan job desk masing-masing calon pengguna yaitu :

Tabel 3.1 *Jobdesk* Calon Pengguna

No	Calon Pengguna	*Job Desk*
1.	Admin	a. Memahami secara jelas prosedur pelayanan administrasi b. Bertanggung jawab atas penerbitan atau pencetakan SKCK dan Surat Laporan Kehilangan
2.	Kapolsek	Bertanggung jawab untuk menentukan penerbitan SKCK dan Surat Laporan Kehilangan
3.	Pemohon	a. Memahami prosedur permohonan pembuatan SKCK baru,pengajuan SKCK kembali perpanjang SKCK dan permohonan laporan kehilangan. b. Bertanggung jawab atas keaslian data diri pemohon.

3.5 Goal Development Process

Pada tahapan ini dilakukan identifikasi Goals yang ingin dicapai oleh Pelayanan Administrasi POLSEK Sindangkerta melalui proses *Goal Development Process*. Pada proses ini peneliti melakukan pemetaan *softgoal* yaitu kondisi yang harus dipenuhi untuk mencapai *goal* kemudian langkah selanjutnya adalah memetakan *softgoal* menjadi sebuah *hardgoal*. *Hardgoal* merupakan suatu fungsionalitas yang harus dimiliki atau disediakan oleh sistem tersebut yang nantinya akan dipecah menjadi sebuah *requirements*.

Tabel 3.2 Identifikasi *Goal*, *Softgoal* dan *Hardgoal*

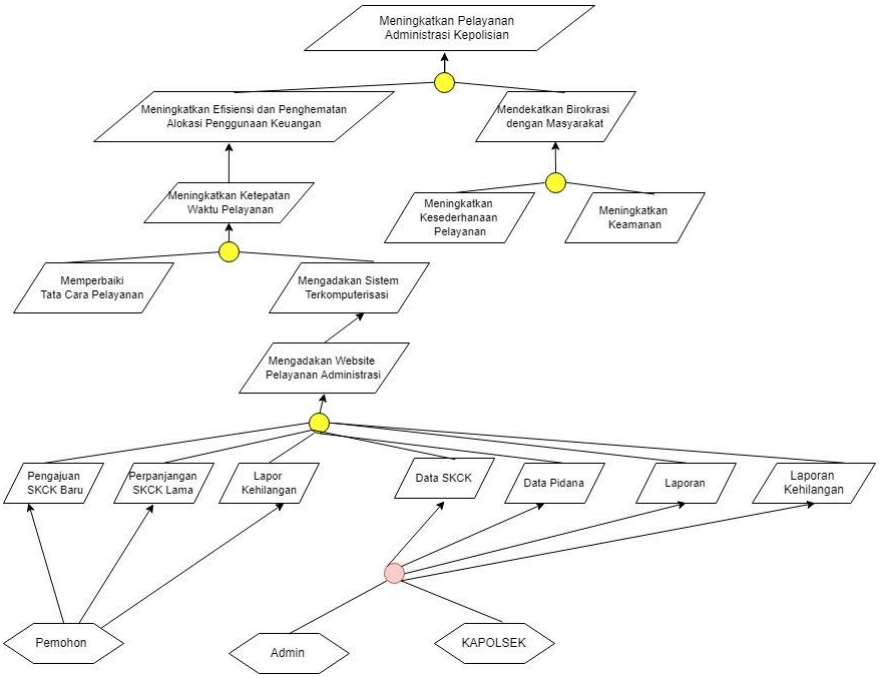
Goal	*Softgoal*	*Hardgoal*
Meningkatkan Pelayanan Administrasi	Meningkatkan efisiensi dan penghematan alokasi penggunaan keuangan	
	Meningkatkan waktu pelayanan	
	Memperbaiki tata cara pelayanan	
	Mengadakan sistem terkomputerisasi	
	Mengadakan website pelayanan administrasi	Pengajuan SKCK Baru
		Perpanjang SKCK Lama
		Lapor Kehilangan
		Data SKCK
		Pidana
		Laporan Kehilangan
		Laporan
	Meningkatkan birokrasi dengan masyarakat	
	Meningkatkan kesederhanaan pelayanan	
	Meningkatkan Keamanan	

3.6 Pemodelan kebutuhan sistem menggunakan KAOS

Pada Penelitian ini, KAOS model diterapkan pada pengelolaan administrasi di Kepolisian Sektor Sidangkerta. Dalam hal ini KAOS digunakan untuk memodelkan kebutuhan sistem pada pengelolaan administrasi. Untuk mendapatkan kebutuhan dari sistem yang akan dikembangkan maka akan dijabarkan terlebih dahulu tujuan yang ingin dicapai oleh Kepolisian Sektor tersebut adalah sebagai berikut :

1. Meningkatkan pelayanan administrasi.
2. Peningkatan kerja dan modernisasi
3. Meningkatkan pemberdayaan media.

Dari tujuan No.1, maka Kepolisian Sektor Sidangkerta memerlukan sebuah perangkat lunak guna untuk mencapai *goal* tersebut yakni meningkatkan pelayanan administrasi tersebut. Untuk menerapkan metode KAOS dalam pengembangannya, langkah pertama yang dilakukan adalah dengan *elicitation* atau proses penggalian kebutuhan dari pengguna dengan melakukan wawancara. Dari informasi yang didapat dan analisis yang telah dilakukan maka berikut adalah pemodelan kebutuhan sistem menggunakan KAOS :



Gambar 3.2 KAOS Dalam Memodelkan Kebutuhan Sistem Pengelolaan Administrasi

3.7 Requirement

Dari hasil pemodelan menggunakan KAOS didapatkan *hardgoal* yang dijadikan sebagai modul-modul yang akan dikembangkan pada sistem pengelolaan administrasi kepolisian. Sehingga diperoleh modul-modul yang akan dikembangkan sebagai berikut :

Tabel 3.3 Modul-modul yang akan dikembangkan

No	User	Nama Modul	Sub Modul
1.	Pemohon (Masyarakat)	Pengajuan SKCK Baru	
2.	Pemohon (Masyarakat)	Perpanjang SKCK Lama	
3.	Pemohon(Masyarakat)	Lapor Kehilangan	
4.	Admin dan KAPOLSEK	Data SKCK	Data Pemohon
	Admin		Data Pengajuan

	Admin		Data Perpanjang SKCK
5.	Admin	Pidana	Pasal Pidana
	Admin		Tindakan Pidana
	Admin dan KAPOLSEK		Data Narapidana
6.	Admin dan KAPOLSEK	Laporan Kehilangan	Data Kehilangan
7.	Admin dan KAPOLSEK	Laporan	Laporan SKCK
			Laporan Narapidana
			Laporan Kehilangan

Bedasarkan modul-modul diatas kemudian akan dipecah kembali menjadi *functional requirements*. *functional requirements* ini akan digunakan sebagai dasar dalam pengembangan sistem. Detail *functional requirements* dari sistem yang dibangun terdapat pada lampiran dokumen ini.

3.8 Validasi *Requirements*

Validasi *requirements* dilaksanakan untuk memastikan bahwa *requirements* yang telah dianalisis ditahap sebelumnya telah sesuai dengan apa yang dibutuhkan oleh calon pengguna. Proses ini memberikan gambaran secara jelas terhadap *requirements* yang telah ditentukan agar dapat dirancang ke dalam model sistem. Proses validasi ini dilakukan dengan cara memberikan data *functional requirements* kepada calon pengguna. Dalam proses validasi *requirements* ini dibagi menjadi 3 bagian berdasarkan kebutuhan masing-masing pengguna yaitu pemohon (masyarakat), admin, dan kapolsek dengan jumlah keseluruhan responden yaitu 23 responden yang terdiri dari 16 orang pemohon (masyarakat), 5 orang bagian administrasi dipolsek dan 2 orang yaitu KAPOLSEK dan Wakil kapolsek. Proses validasi ini dilakukan dengan cara memberikan kuisoner yang berisi data *functional requirements* dengan dua indikator dalam *good requirements* yaitu *correct* dan *clear*. *Correct* bertujuan untuk menguji apakah spesifikasi yang ditulis adalah benar dan sesuai kebutuhan pengguna. Sedangkan *Clear* adalah untuk memastikan apakah spesifikasi yang dibuah sudah tepat dan dapat menggambarkan fitur yang akan dibangun. Indikator ini dipilih karena untuk mengetahui kebenaran spesifikasi kebutuhan yang telah dibuat adalah tepat dan dapat menggambarkan fitur yang akan dikembangkan kepada pengguna.

3.9 Pengembangan Sistem

Dalam tahapan ini dilakukan pengembangan website dengan menggunakan bahasa pemograman PHP dan MySql sebagai penyimpanan database. Proses pengembangan website berdasar pada spesifikasi kebutuhan pengguna yang telah dianalisis dan dirancang dalam bentuk pemodelan seperti yang telah disebutkan sebelumnya.

4 Evaluasi

4.1 Hasil Pengujian Indikator Correct

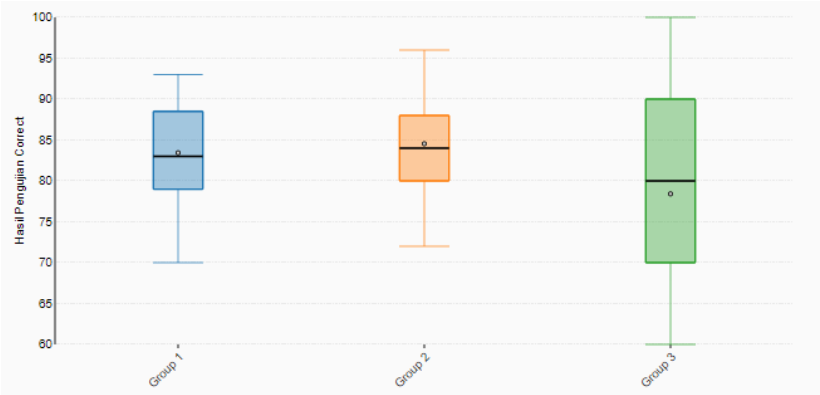
Berikut adalah hasil pengujian dengan indikator correct pada ke tiga jenis user :

Tabel 4.1 Hasil Pengujian Indikator *Correct*

Data Summary							
Groups	N	Min	Q_1	Median	Q_3	Max	Mean
Group 1	23	70	79	83	88.5	93	83.4348
Group 2	59	72	80	84	88	96	84.5424
Group 3	44	60	70	80	90	100	78.4091

Keterangan
Groups 1 : Hasil Pengujian Faktor *Correct* Pemohon
Groups 2 : Hasil Pengujian Faktor *Correct* Admin

Groups 3 : Hasil Pengujian Faktor *Correct* Kapolsek



Gambar 4.1 Box Plot Hasil Pengujian Indikator *Correct*

Berdasarkan hasil pengujian dengan menggunakan perhitungan skala likert dan penggambaran menggunakan box plot pada groups 1 (pengujian faktor *correct* pemohon) dapat dilihat bahwa nilai terhadap *requirements* yang ada pada *requirements* pemohon cenderung tidak berbentuk simetris. Hal ini dapat dilihat dari jarak Q1 ke median dan median ke Q3 yang tidak sama panjangnya. Berdasakan hasil pengujian nilai mean yang di dapat pada pengujian faktor *correct* pemohon adalah 83,43% atau dapat dikategorikan pemohon setuju dengan *requirements* tersebut adapun kekurangannya sebesar 16,57%.

Hasil pengujian pada groups 2 (pengujian faktor *correct* admin) dapat dilihat nilai terhadap *requirements* yang ada pada *requirements* admin berbentuk simetris. Hal ini dapat dilihat dari jarak Q1 ke median dan median ke Q3 yang sama panjangnya. Berdasarkan hasil pengujian nilai mean yang didapat pada pengujian faktor *correct* admin adalah 84,54% atau dapat dikategorikan admin sangat setuju dengan *requirements* tersebut adapun kekurangannya sebesar 14,46%.

Hasil pengujian pada groups 3 (pengujian faktor *correct* kapolsek) dapat dilihat dari nilai terhadap *requirements* yang ada pada *requirements* kapolsek berbentuk simetris. Hal ini dapat dilihat dari jarak Q1 ke median dan median ke Q3 yang sama panjangnya. Berdasarkan hasil pengujian nilai mean yang didapat pada pengujian faktor *correct* kapolsek adalah 78,4% atau dapat dikategorikan setuju dengan *requirements* tersebut adapun kekurangannya sebesar 21,6%

Dari ketiga hasil pengujian diatas didapatkan nilai mean terhadap pengujian faktor *correct* adalah sebesar 82,12% dan dikategorikan setuju berdasarkan skala likert adapun kekurangannya adalah 17,88% dikarenakan ada beberapa *requirement* yang tidak terspesifikasi dengan detail.

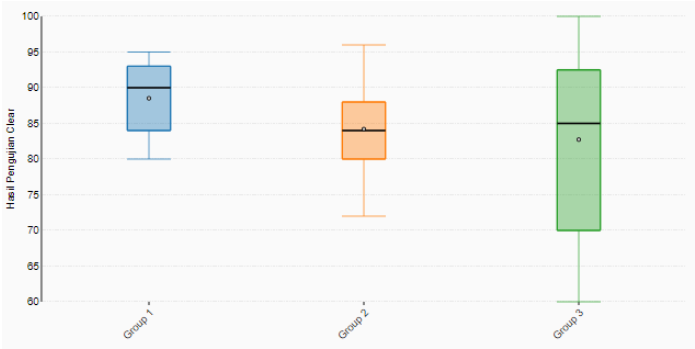
4.2 Hasil Pengujian Indikator Clear

Berikut adalah hasil pengujian dengan indikator *clear* pada ke tiga jenis user :

Tabel 4.2 Hasil Pengujian Indikator *Clear*

Data Summary							
Groups	N	Min	Q_1	Median	Q_3	Max	Mean
Group 1	23	80	84	90	93	95	88.5217
Group 2	59	72	80	84	88	96	84.2034
Group 3	44	60	70	85	92.5	100	82.7273

Keterangan
Groups 1 : Hasil Pengujian Faktor *Clear* Pemohon
Groups 2 : Hasil Pengujian Faktor *Clear* Admin
Groups 3 : Hasil Pengujian Faktor *Clear* Kapolsek



Gambar 4.2 Box Plot Hasil Pengujian Indikator *Clear*

Berdasarkan hasil pengujian dengan menggunakan perhitungan skala likert dan penggambaran menggunakan box plot pada groups 1 (pengujian faktor *clear* pemohon) dapat dilihat bahwa nilai terhadap *requirements* yang ada pada *requirements* pemohon cenderung tidak berbentuk simetris. Hal ini dapat dilihat dari jarak Q1 ke median dan median ke Q3 yang tidak sama panjangnya. Berdasakan hasil pengujian nilai mean yang di dapat pada pengujian faktor *clear* pemohon adalah 88,52% atau dapat dikategorikan pemohon sangat setuju dengan *requirements* tesrebut adapun kekurangannya sebesar 11,48%.

Hasil pengujian pada groups 2 (pengujian faktor *clear* admin) dapat dilihat bahwa nilai terhadap *requirements* yang ada pada *requirements* admin berbentuk simetris. Hal ini dapat dilihat dari jarak Q1 ke median dan median ke Q3 yang sama panjangnya. Berdasarkan hasil pengujian nilai mean yang didapat pada pengujian faktor *clear* admin adalah 84,2% atau dapat dikategorikan admin sangat setuju dengan *requirements* tersebut adapun kekurangannya sebesar 15,8%.

Hasil pengujian pada groups 3 (pengujian faktor *clear* kapolsek) dapat dilihat bahwa nilai terhadap *requirements* yang ada pada requirements kapolsek cenderung tidak simetris. Hal ini dapat dilihat dari jarak Q1 ke median dan median ke Q3 tidak sama panjang. Berdasarkan hasil pengujian nilai mean yang didapat pada pengujian faktor *clear* kapolsek adalah 82,72%. Atau dapat dikategorikan kapolsek setuju dengan requirements tersebut adapun kekurangannya sebesar 17,28%.

Dari ketiga hasil pengujian diatas didapatkan nilai mean terhadap pengujian faktor *clear* adalah sebesar 85,14% dan dikategorikan sangat setuju berdasarkan skala likert adapun kekurangannya adalah 14,86% dikarenakan ada beberapa *requirement* yang belum bisa menggambarkan fitur yang akan dibuat dengan jelas.

5 Kesimpulan

Berdasarkan penelitian yang telah dilakukan mengenai implementasi GORE menggunakan KAOS untuk pengelolaan administrasi kepolisian Sindangkerta dalam menentukan spesifikasi kebutuhan adapun kesimpulan yang didapat merupakan hasil dari validasi *requirements* berdasarkan indikator *good requirements* pada penelitian ini yang digunakan adalah *correct* dan *clear*. Dari hasil pengujian berdasarkan masing-masing *requirements user* didapatkan nilai mean terhadap pengujian faktor *correct* adapun *requirements* yang terspesifikasi dengan benar sebesar 82,12% adapun kekurangannya adalah 17,88% dikarenakan ada beberapa *requirement* yang tidak terspesifikasi dengan detail. Adapun hasil pengujian berdasarkan indikator *clear* menunjukkan sebesar 82,72% dan dikategorikan setuju atau *requirements* tersebut *clear* terhadap *requirements* adapun kekurangannya adalah 17,28% dikarenakan ada beberapa *requirement* yang belum bisa menggambarkan fitur yang akan dibuat dengan jelas.

Adapun saran perbaikan untuk penelitian selanjutnya dapat dilakukan dengan melakukan penggabungan metode-metode lain kedalam *framework* KAOS guna untuk meningkatkan presisi dan hasil akurasi dari *requirements*. Selain itu penurunan *goals* yang ditarik menjadi sebuah *requirements* perlu disempurnakan dan bukan hanya memilih mana *requirements* yang dapat diselesaikan oleh sistem. Namun, harus diperkuat kembali dengan perhitungan kuantitatif yang lebih nyata.

Daftar Pustaka

[1] K. Wiegers and J. Beatty, Software Requirements Third Edition, Washington: Microsoft Press, 2013.
[2] IEEE Standards Collection : Software Engineering, IEEE Standard 610.12-1990, IEEE, 1993.
[3] Siahaan, D. O, Analisa Kebutuhan Dalam Rekayasa Perangkat Lunak, Yogyakarta: Andi, 2012.
[4] A. Sultan, B. Jameela, H. Nisar. "Goal Oriented requirements Engineering – A Review", College of Computers and Information Technology,Saudi Arabia. 2007.
[5] A. Van Lamsweerde and E.Letier, "From object orientation to goal orientation: A paradigm shift for requirements engineering, "Radical Innovations of Software and Systems Engineering in the Future, vol. 2941, no. 1, pp.325-340,2004
[6] Kavkali, E. "Goal-Oriented Requirement Engineering: A Unifying Framework." Requirement Engineering, January 2002, Vol.6 No.4. pp.237-251.
[7] M. Teruel, E. Navarro, and V. López-Jaquero, "Comparing Goal-Oriented Approaches to Model Requirements for CSCW," Evaluation of Novel Approaches to Software Engineering, pp.169-184, 2012.
[8] Lapouchian, A., "Goal-Oriented Requirements Engineering: An Overview of the Current Research", Departement of Computer Science University Of Toronto,2005.
[9] A. van Lamsweerde. Goal-Oriented Requirements Engineering: A Guided Tour. Proc.5th IEEE International Symposium on Requirements Engineering (RE'01), Toronto, Canada, August 2001.
[10] Haruhiko Kaiya, Hisayuki Horai, Motoshi Saeki, "AGORA: Attributed Goal-Oriented Requirements Analysis Method". Proceedings of the IEEE Joint International Conference on Requirements Engineering (RE'02), University of Essen, Germany, 2002.
[11] Hui, B., Liaskos, S., Mylopoulos, J., "Requirements Analysis for Customizable Software: A Goals-SkillsPreferences Framework". Proc. International Conference on Requirements Engineering (RE'03), Monterey, USA, September 2003.

[12] Singh, Y., Gosain, A., Kumar, M., "Evaluation of Agent Oriented Requirements Engineering Frameworks", International Conference on Computer Science and Software Engineering, 2008.

[13] Sugiyono, Metode Penelitian Kuantitatif, Kualitatif, dan R&D, Bandung: ALFABETA, 2016.

[14] Peter Zielczynski, *Requirements Management Using IBM Rational RequisitePro*, IBM Press, 2008