

RANCANG BANGUN APLIKASI VIRTUAL GLOBE MAP QUESTION ANSWERING SYSTEM DENGAN METODE Q-CLASS

THE DESIGN OF VIRTUAL GLOBE MAP QUESTION ANSWERING SYSTEM APPLICATION USING Q - CLASS

Rendy Halim¹, Seng Hansun²

**Program Studi Teknik Informatika
Universitas Multimedia Nusantara – Tangerang
¹rendy2307@rocketmail.com, ²hansun@umn.ac.id**

Abstrak

Paper ini menjelaskan Virtual Application Globe Peta yang diimplementasikan dengan Pertanyaan Menjawab konsep Sistem yang menggunakan metode Q-Class. Banyak aplikasi berbasis lokasi layanan menggunakan kata kunci sebagai masukan mereka, tetapi dalam makalah ini, pertanyaan yang digunakan sebagai input. Tujuan dari penelitian ini adalah untuk membangun Virtual Globe Peta Pertanyaan Menjawab aplikasi System dengan metode Q-Class, dan kemudian membandingkannya dengan aplikasi sejenis lainnya, seperti Google Map. Penelitian ini menggunakan empat variabel untuk mengukur kepuasan pengguna, yaitu Efektivitas Sistem, Efektivitas Pengguna, Usaha Pengguna, dan Karakteristik Pengguna. Sampel diambil dari 30 responden dengan 43 pertanyaan.

Kata kunci: Google Map, Q-Class, Pertanyaan Menjawab Sistem, Virtual Globe Peta

Abstract

This paper describes the Virtual Globe Map Application implemented by using Question Answering System concept and Q-Class method. Many location-based service applications use keywords as their input, but in this paper, question is used as an input. The objective of this research is to build Virtual Globe Map Question Answering System application with Q-Class method, and to compare it to another similar application, such as Google Map. This research uses four variables to measure the user satisfaction, i.e. System Effectiveness, User Effectiveness, User Effort, and User Characteristic. Samples are taken from 30 respondents with 43 questions.

Keywords: Google Map, Q-Class, Question Answering System, Virtual Globe Map

Tanggal Terima Naskah : 09 April 2015
Tanggal Persetujuan Naskah : 03 Agustus 2015

1. PENDAHULUAN

Saat ini, mobilitas sangat berperan dalam kehidupan manusia. Manusia bergerak dari satu tempat ke tempat lain untuk memenuhi hak dan kewajibannya. Tingginya mobilitas manusia ini membuat informasi tentang suatu lokasi menjadi sangat penting, misalnya seseorang yang mengadakan perjalanan ke suatu tempat asing, segala informasi

tentang lokasi tersebut akan menjadi sangat bermanfaat. Umumnya, seseorang dapat bertanya kepada orang di lokasi tersebut atau menggunakan peta untuk dapat sampai ke tempat tujuan. Namun, dengan perkembangan teknologi komunikasi dan informasi yang pesat saat ini, pencarian lokasi dapat dilakukan dengan bantuan perangkat navigasi, seperti *Global Positioning System* (GPS) atau layanan berbasis lokasi milik *Google*, yaitu *Google Maps*.

Layanan berbasis lokasi (*Location Based Services* atau LBS) merupakan salah satu topik menarik dalam perkembangan teknologi informasi [1]. Hal ini terbukti dengan diterapkannya LBS pada beberapa jejaring sosial, baik yang sudah mendunia seperti *Twitter* dan *Facebook*, ataupun yang baru-baru ini digemari masyarakat, *Path*. Untuk layanan yang secara spesifik memakai LBS sebagai fitur utamanya, terdapat *Foursquare* dan *Google Maps*.

Tren perkembangan interaksi manusia dengan komputer di masa mendatang mengarah pada model interaksi alamiah menggunakan bahasa natural manusia [2]. Namun, model interaksi yang diterapkan aplikasi seperti *Foursquare* dan *Google Maps* masih mengacu pada konsep *search engine*. Teknik pencarian dengan konsep *search engine* ini mengharuskan pengguna mengisikan beberapa kolom pencarian untuk mencari suatu informasi. Konsep *search engine* ini terkadang tidak mempertimbangkan pengaksesan informasi dengan kalimat alamiah. *Google Maps* saat ini telah mulai mengembangkan konsep bahasa *natural* sebagai masukannya, namun belum optimal. Bagi pengguna, cara yang paling nyaman untuk berinteraksi adalah dengan bahasa *natural* [3]. Salah satu cara untuk mengalamiahkan interaksi antara manusia dengan komputer adalah dengan menerapkan *Question Answering* (QA). QA merupakan salah satu bidang Pemrosesan Bahasa Alami (*Natural Language Processing*), yang memungkinkan pengguna untuk bertanya kepada mesin seperti layaknya mereka bertanya pada manusia.

LBS sebenarnya terbagi menjadi beberapa jenis [4]. Terdapat beberapa contoh LBS, seperti *Foursquare*, *Google Maps*, *Waze*, dan *Localmind*. *Foursquare* berjenis *Local Search and Discovery*, yang artinya pengguna layanan ini dapat menemukan dan ditemukan oleh pengguna lain di sekitarnya sedangkan *Waze* dan *Localmind* memiliki layanan *GPS-based Geographical Navigation Social Media*, dimana pengguna menggunakan GPS untuk melakukan navigasi ke lokasi tertentu dengan fitur-fitur, seperti *social media*. *Google Maps* memiliki jenis tersendiri, yaitu *Virtual Globe Map*, karena memiliki kemiripan dengan fungsi *globe*.

Pada penelitian ini, dikembangkan aplikasi *Virtual Globe Map*. Untuk mengimplementasikan konsep QA System, terdapat dua metode yang dapat digunakan, yaitu dengan membangun *class* pertanyaan berdasarkan jenisnya (*Q-Class*), atau dengan membangun daftar pola pertanyaan dan mengarahkannya pada sebuah tipe jawaban yang sesuai [5]. Dalam penelitian ini digunakan metode *Q-Class*. Keunggulan dari metode ini adalah pengimplementasiannya yang tidak terlalu kompleks dan rumit.

2. QUESTION ANSWERING SYSTEM

Question Answering System (QA System) merupakan sebuah sistem yang mengizinkan pengguna menyatakan kebutuhannya dalam bentuk yang lebih spesifik dan alami, yaitu dalam bentuk *Natural Language Question* [6]. Contohnya, jika seorang pengguna ingin mengetahui nama presiden pertama Indonesia, pengguna dapat memberikan *input* pertanyaan seperti “Siapa presiden pertama Indonesia?” pada QA system, dan sistem akan memberikan keluaran “Ir. Soekarno”. Hal ini berbeda dengan *search engine*, dimana jika pengguna memasukkan pertanyaan “Siapa presiden pertama Indonesia?”, sistem akan mengembalikan dokumen yang memuat kalimat tersebut, dan untuk menemukan jawaban yang diinginkan, pengguna perlu untuk memilah-milah setiap

dokumen tersebut.

Umumnya, sebuah sistem QA terdiri atas tiga komponen utama, yaitu *Question Analyzer*, *Information Retriever*, dan *Answer Finder* [7]. Namun, pada pengaplikasiannya, khususnya pada bahasa Indonesia, dibutuhkan suatu *pre-processing* untuk pertanyaan yang diajukan. Hal ini disebabkan adanya berbagai imbuhan pada Bahasa Indonesia yang dapat ditambahkan pada suatu kata. Oleh karena itu, pada tahap *pre-processing* ini akan dilakukan *stemming* atau proses mengubah suatu kata menjadi kata akarnya (*root word*). Pada penelitian ini, digunakan algoritma Porter untuk melakukan *stemming*.

Tahapan *Question Analyzer* berfungsi untuk mengklasifikasikan pertanyaan pengguna sesuai dengan tipe jawaban yang diharapkan dan mendapatkan kata kunci dari pertanyaan tersebut. Terdapat dua buah pendekatan sederhana untuk menentukan tipe pertanyaan yang diajukan, yaitu dengan membangun *class* pertanyaan berdasarkan jenisnya (*Q-Class*) atau dengan membangun daftar pola pertanyaan dan mengarahkannya pada sebuah tipe jawaban yang sesuai. Pada penelitian ini, digunakan *Q-Class* untuk mengidentifikasi pertanyaan. *Information Retriever* berfungsi untuk menemukan dokumen yang mengandung kandidat jawaban berdasarkan kata kunci dan tipe jawaban yang dihasilkan *Question Analyzer*. *Answer Finder* berfungsi untuk menemukan kandidat jawaban dari dokumen yang telah ditemukan pada *Information Retriever*.

3. ALGORITMA PORTER

Untuk mengubah suatu kalimat menjadi bentuk dasarnya, terdapat beberapa cara, salah satunya adalah dengan *stemming*. Algoritma Porter adalah salah satu algoritma untuk melakukan *stemming*. Algoritma ini dibuat oleh Martin Porter dan dipublikasikan tahun 1980. Berikut tahapan algoritma Porter:

1. Hapus *Particle*
2. Hapus *Possesive Pronoun*
3. Hapus *First Order Derivational Prefix*. Jika tidak ada, lanjutkan ke langkah 4a. Jika ada, lanjutkan ke langkah 4b.
- 4a. Hapus *Second Order Derivational Prefix*. Lanjutkan ke langkah 5a.
- 4b. Hapus *Derivational Suffix*. Jika tidak ada, maka kata tersebut diasumsikan sebagai *root word*. Jika ada, lanjutkan ke langkah 5b.
- 5a. Hapus *Derivational Suffix*. Kata akhir diasumsikan sebagai *root word*.
- 5b. Hapus *Second Order Derivational Prefix*. Kata akhir diasumsikan sebagai *root word*.

4. Q-CLASS

Suatu pertanyaan memiliki banyak bentuk dan beragam jawaban, oleh karena itu diperlukan suatu metode untuk mengidentifikasinya. *Q-Class* adalah metode untuk mengidentifikasi sebuah pertanyaan tergantung dari jenisnya, serta menentukan jawaban apa yang sesuai. Metode ini digunakan D. Moldovan [8] untuk merancang *Lasso QA System*.

Jenis jawaban dari sebuah pertanyaan biasanya dapat ditentukan dari kata tanya pertanyaan itu. Oleh karena itu, tahap pertama dari pengklasifikasian *Q-Class* adalah mengelompokkan pertanyaan berdasarkan kata tanya, seperti apa, kenapa, siapa, bagaimana, kemana, dan lain-lain.

Namun, pengelompokkan ini tidak cukup untuk menemukan jawaban yang diinginkan. Untuk pertanyaan seperti “Siapa presiden Indonesia pertama?”, jawabannya sangat jelas, yaitu nama orang. Tapi hal ini tidak berlaku untuk jenis pertanyaan “apa”. Karena “apa” dapat memiliki beberapa jenis jawaban. Masih banyak jenis

pertanyaan lain yang memiliki banyak jenis jawaban. Solusi dari hal ini adalah adanya konsep “fokus” pada *Q-Class*.

Fokus adalah satu atau banyak kata yang mendefinisikan jenis jawaban apa yang diinginkan dari suatu pertanyaan, misalnya, “Apa ibukota Indonesia?”, fokus dari pertanyaan ini adalah “ibukota”. Dengan mengetahui jenis pertanyaan dan fokusnya, jenis jawaban lebih mudah didapatkan.

Setelah mendapatkan jenis pertanyaan dan fokusnya, bagian terakhir yang perlu didapat adalah *keyword*. *Keyword* didapat dengan menggunakan delapan heuristik yang berurutan:

- a. *Keyword-Heuristic 1*: Kata atau kalimat dengan tanda kutip menjadi *keyword*
- b. *Keyword-Heuristic 2*: Entitas bernama menjadi *keyword*
- c. *Keyword-Heuristic 3*: Nominal kompleks dengan kata sifat menjadi *keyword*
- d. *Keyword-Heuristic 4*: Nominal kompleks lainnya menjadi *keyword*
- e. *Keyword-Heuristic 5*: Semua kata benda dengan kata sifat menjadi *keyword*
- f. *Keyword-Heuristic 6*: Semua kata benda lainnya menjadi *keyword*
- g. *Keyword-Heuristic 7*: Semua kata kerja menjadi *keyword*
- h. *Keyword-Heuristic 8*: Fokus pertanyaan menjadi *keyword*

5. VARIABEL PENELITIAN

Kepuasan seseorang terhadap suatu aplikasi tidak dapat diukur oleh satu variabel saja, melainkan harus melalui beberapa variabel [9]. Dalam penelitian ini, peneliti menggunakan empat variabel untuk menentukan kepuasan pengguna *Question Answering System* yang dibangun, yaitu *System Effectiveness*, *User Effectiveness*, *User Effort*, dan *User Characteristics*.

A. *System Effectiveness*

System Effectiveness menunjukkan bagaimana suatu sistem dapat memberikan hasil yang sesuai dengan masukan pengguna. Dalam penelitian ini, *System Effectiveness* dinilai berdasarkan kesesuaian hasil keluaran aplikasi dengan masukan pengguna.

B. *User Effectiveness*

User Effectiveness menunjukkan bagaimana pengguna dapat mencapai suatu tujuan dengan menggunakan sistem yang ada, artinya, jika *User Effectiveness* rendah, maka pengguna akan kesulitan untuk mencapai suatu tujuan (*goal*). Seseorang akan menyerah mencari suatu informasi pada mesin pencari setelah melakukan delapan kali pencarian [10]. Dalam penelitian ini, *User Effectiveness* dinilai berdasarkan keberhasilan pengguna mendapatkan hasil yang diinginkan menurut skenario yang diberikan. Pengguna dinyatakan gagal mendapatkan jawaban jika pengguna telah memasukkan pertanyaan sebanyak delapan kali.

C. *User Effort*

User Effort menunjukkan besarnya usaha yang diperlukan pengguna untuk mencapai suatu tujuan. Dalam penelitian ini, *User Effort* dinilai berdasarkan jumlah pertanyaan yang dimasukkan pengguna untuk mencapai suatu tujuan.

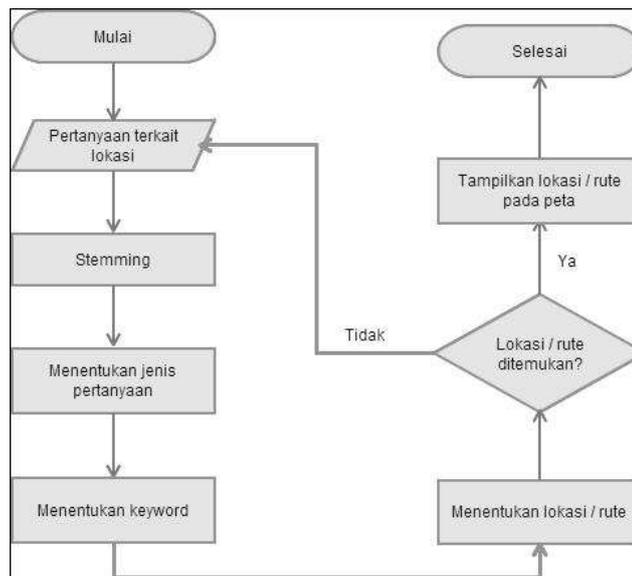
D. *User Characteristics*

Faktor manusia tentunya merupakan salah satu kunci kepuasan pengguna suatu aplikasi. *User Characteristics* menunjukkan kepuasan pengguna berdasarkan persepsinya masing-masing. Hasil dari variabel ini dapat terbagi menjadi lima kriteria objektif, yaitu Sangat Baik, Baik, Cukup Baik, Kurang Baik, dan Sangat Kurang Baik. Untuk menghitung variabel ini, digunakan pendekatan dengan skala Likert [11].

6. RANCANGAN DAN TAMPILAN ANTARMUKA APLIKASI

6.1 Rancangan Aplikasi

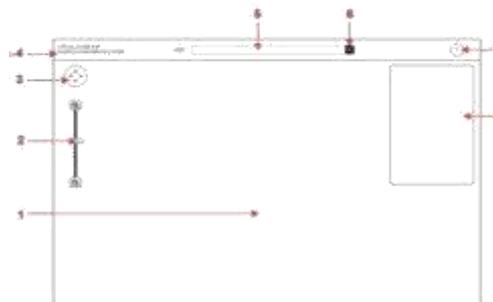
Gambar 1 menunjukkan diagram alir aplikasi yang menjelaskan alur proses aplikasi secara garis besar. Alur aplikasi dimulai dari *input* pengguna berupa pertanyaan terkait lokasi. Pertanyaan tersebut akan melalui proses *stemming* agar menjadi bentuk dasarnya. Setelah itu, dengan menggunakan metode *Q-Class*, pertanyaan yang diberikan akan diidentifikasi jenis dan *keyword*-nya. Setelah jenis dan *keyword*-nya didapat, jika lokasi ditemukan maka lokasi dan jawaban dari pertanyaan tersebut akan ditampilkan pada peta.



Gambar 1. Diagram alir aplikasi secara garis besar

6.2 Rancangan Tampilan Antarmuka Aplikasi

Aplikasi ini hanya memiliki satu tampilan utama dan satu tampilan pendukung untuk petunjuk pemakaian aplikasi. Rancangan tampilan utama ditunjukkan pada Gambar 2.



Gambar 2. Rancangan tampilan utama aplikasi

1. *Map Canvas*. *Canvas* ini berisi peta yang akan ditampilkan sesuai dengan masukan pengguna.
2. *Zoom Control*. *Control* ini berfungsi untuk melihat peta lebih dekat (*zoom in*) atau lebih jauh (*zoom out*).

3. *Pan Control*. *Control* ini berfungsi untuk menggeser peta ke arah atas, kanan, bawah, dan kiri.
4. Baris judul. Berisikan judul aplikasi ini.
5. *Textbox input*. Pengguna dapat memasukkan pertanyaan seputar lokasi yang ingin diajukan ke dalam *textbox* ini.
6. Tombol *search*. Tombol ini ditekan setelah pengguna memasukkan pertanyaan pada *textbox input*. Aplikasi akan menunjukkan lokasi sesuai dengan pertanyaan pengguna.
7. Tombol *help*. Apabila tombol ini ditekan, aplikasi akan menampilkan petunjuk pemakaian aplikasi.
8. Panel navigasi. Panel ini akan menunjukkan rute yang harus dilalui dari lokasi asal ke lokasi tujuan.

7. IMPLEMENTASI

Tampilan antarmuka utama aplikasi yang telah dibangun diperlihatkan pada gambar berikut.

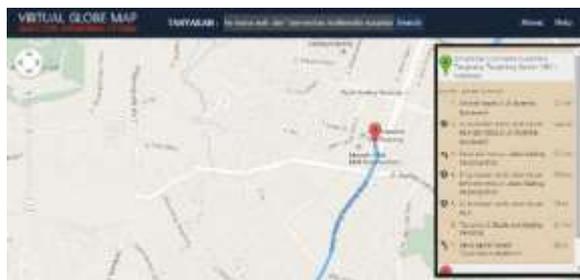


Gambar 3. Tampilan utama aplikasi

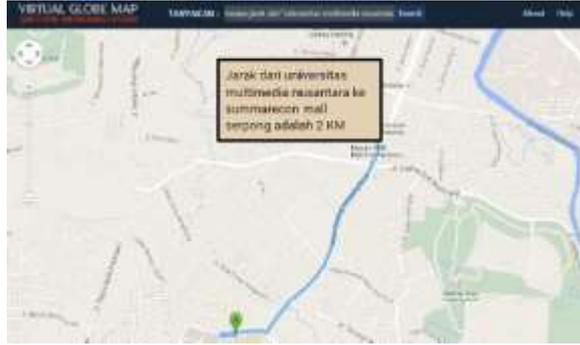
Hasil penerapan *QA System* dengan menggunakan *Q-Class* diperlihatkan pada gambar 4 - 6 berikut.



Gambar 4. Tampilan saat pengguna memasukkan pertanyaan 'Dimana lokasi Universitas Multimedia Nusantara?'



Gambar 5. Tampilan saat pengguna memasukkan pertanyaan 'Kemana arah dari Universitas Multimedia Nusantara ke Summarecon Mall Serpong?'



Gambar 6. Tampilan saat pengguna memasukkan pertanyaan ‘Berapa jarak dari Universitas Multimedia Nusantara ke Summarecon Mall Serpong?’

Pada aplikasi yang dibangun, pengguna dapat memasukkan sembarang pertanyaan yang diinginkan dan akan diproses oleh sistem untuk menentukan jenis pertanyaan dan fokusnya menggunakan metode *Q-Class*. Terdapat tiga kelompok pertanyaan utama yang dicakup oleh aplikasi, yakni ‘dimana’ untuk menentukan lokasi suatu objek yang dicari dan hasilnya menampilkan titik lokasi objek pada *virtual globe map*, ‘kemana’ untuk menentukan arah dari satu lokasi ke lokasi lainnya dan hasilnya menampilkan arah atau jalan yang dilalui antara dua objek yang ditentukan, dan ‘berapa’ untuk menentukan perkiraan jarak antara dua lokasi dan hasilnya menampilkan jalur antara kedua obyek yang ditentukan dalam *virtual globe map* dan *pop-up box* yang menunjukkan perkiraan jarak antara kedua objek tersebut.

7.1 *Sample Data*

Pengumpulan data untuk analisis lebih lanjut dilakukan dengan bantuan beberapa sukarelawan untuk menggunakan aplikasi, yang selanjutnya akan disebut responden. Responden akan diminta untuk memasukkan sembarang pertanyaan untuk mencapai suatu jawaban dari skenario yang diberikan secara acak. Jumlah maksimal pertanyaan yang diajukan seorang responden adalah sebanyak delapan kali (lebih dari itu, jawaban dianggap tidak ditemukan), dengan minimal satu kali (jika jawaban langsung ditemukan). Setelah menggunakan aplikasi, responden diminta untuk mengisi kuisisioner tentang penilaian responden terhadap aplikasi tersebut.

Jumlah minimal *sample* yang digunakan untuk populasi tak hingga adalah 30 [12]. Oleh karena itu, pengujian dilakukan terhadap 30 responden. Dengan demikian terdapat 30 lembar kuisisioner. Jumlah pertanyaan yang diajukan 30 responden tersebut adalah 43 pertanyaan karena terdapat beberapa responden yang mengajukan pertanyaan lebih dari satu kali untuk menemukan jawaban yang diinginkan. Sebaran jenis pertanyaan yang diujikan kepada responden diperlihatkan pada Tabel 1.

Tabel 1. Sebaran responden

Jenis Pertanyaan	Jumlah responden	Jumlah pertanyaan
Jarak	7	9
Lokasi	7	7
Rute	12	19
Waktu	4	8

7.2 Analisis Sample Data

Tabel 2 menunjukkan rekapitulasi hasil pengujian. Setiap pertanyaan yang dimasukkan oleh pengguna akan menghasilkan sebuah jawaban dari aplikasi. Terdapat tiga jenis jawaban, yaitu jawaban benar, jawaban salah, dan jawaban tidak ditemukan. Jawaban benar muncul jika jawaban yang dihasilkan sesuai dengan pertanyaan yang diajukan dan memiliki informasi yang tepat. Jawaban salah muncul jika jawaban yang ditampilkan aplikasi memiliki informasi yang tidak tepat atau jenis jawaban tidak sesuai dengan yang diinginkan oleh pengguna. Jawaban tidak ditemukan muncul jika pertanyaan yang dimasukkan tidak menggunakan struktur yang benar atau informasi yang dicari tidak tersedia dalam aplikasi yang dibangun.

Tabel 2. Rekapitulasi hasil pengujian

Jenis Pertanyaan	Jawaban benar	Jawaban salah	Jawaban tidak ditemukan
Jarak	7	0	2
Lokasi	7	0	0
Rute	12	0	7
Waktu	4	0	4

Terdapat empat variabel yang diukur pada penelitian ini, yaitu *System Effectiveness*, *User Effectiveness*, *User Effort*, dan *User Characteristics*.

7.2.1 System Effectiveness

Variabel *System Effectiveness* dihitung berdasarkan rasio jumlah jawaban benar terhadap jumlah total pertanyaan. Dari hasil pengujian, terdapat 30 jawaban benar dari 43 pertanyaan, artinya variabel *System Effectiveness* dari aplikasi ini memiliki persentase sebesar 69,76%. *System Effectiveness* yang dapat diterima oleh seseorang dalam melakukan pencarian informasi adalah 70% [13]. Dengan demikian, aplikasi ini memiliki *System Effectiveness* yang sedikit di bawah standar (Kurang Baik).

7.2.2 User Effectiveness

User Effectiveness dihitung berdasarkan rasio jumlah skenario yang dapat diselesaikan terhadap jumlah skenario keseluruhan. Hasilnya, seluruh skenario dapat diselesaikan oleh responden, dimana persentase dari *User Effectiveness* adalah 100%. Responden menilai *User Effectiveness* aplikasi pada penelitian ini Sangat Baik.

7.2.3 User Effort

User Effort merepresentasikan usaha yang diperlukan seseorang untuk mencari sebuah informasi, artinya semakin kecil usaha yang dikeluarkan, semakin baik pula aplikasi tersebut. Dari hasil pengujian seluruh responden, rata-rata responden membutuhkan 1,43 kali pencarian untuk menemukan informasi. Nilai ini dinilai Baik, karena rata-rata pengguna mesin pencari membutuhkan setidaknya tiga kali pencarian untuk menemukan suatu informasi [10].

7.2.4 User Characteristics

Dari hasil kuisisioner yang disebar, dengan menggunakan skala Likert, didapatkan hasil variabel *User Characteristics* dinilai Baik.

7.2.5 Kepuasan Pengguna

Dengan *System Effectiveness* dinilai Kurang Baik, *User Effectiveness* dinilai Sangat Baik, *User Effort* dan *User Characteristics* dinilai Baik, nilai kepuasan pengguna adalah 80%. Hasilnya, pengguna Puas dengan aplikasi *Virtual Globe Map* dengan konsep *Question Answering System*.

8. KESIMPULAN

Aplikasi *Virtual Globe Map* dengan konsep *Question Answering System* telah berhasil dibangun dengan menggunakan metode *Q-Class*. Berdasarkan hasil pengujian dengan menggunakan empat variabel penelitian, pengguna puas terhadap aplikasi yang dibangun.

REFERENSI

- [1]. Y. Dwi. 2013. Peluang Mengintegrasikan Location Based Service dengan Question Answering System untuk Tugas Pencarian Lokasi. Sekolah Tinggi Manajemen Informatika dan Komputer Bina Patria Magelang, Magelang.
- [2]. I. Lanin. 2014. Pemberdayaan TIK untuk Pengembangan Bahasa Indonesia dan Pengajaran BIPA. ASILE 2014, Bali.
- [3]. W. Zadrozny, M. Budzikowska, J. Chai, dan N. Kambhatla. 2000. Natural Language Dialogue for Personalized Interaction. *Communications of the ACM*, 43, hal. 116-120.
- [4]. A. Hartanto. 2003. Mengenal Aspek Teknik dan Bisnis – Location Based Service. Elex Media Komputindo, Jakarta.
- [5]. Gunawan dan G. Lovina. 2006. Question Answering System dan Penerapannya Pada Alkitab. Sekolah Tinggi Teknik Surabaya, Surabaya.
- [6]. C. Monz. 2003. From Document Retrieval to Question Answering. ILLC Dissertation Series, Amsterdam.
- [7]. S.M. Harabagiu, M. A. Pasca, dan S. J. Maiorano. 2000. Experiments with Open-Domain Textual Question Answering. Southern Methodist University, Dallas.
- [8]. D. Moldovan. 2000. LASSO: A Tool for Surfing the Answer Net. Dallas.
- [9]. A. Al-Maskari dan M. Sanderson. 2010. A Review of Factors Influencing User Satisfaction in Information Retrieval. Royal Melbourne Institute of Technology, Australia.
- [10]. S. Bain. 2013. Survey by SearchYourCloud: Document Search, in and out of the Office. Dalam: <http://searchyourcloud.com>.; diakses 20 November 2014.
- [11]. Sugiono. 2012. Metode Penelitian Kuantitatif, Kualitatif, dan R & D. Alfabeta, Bandung.
- [12]. Sugiono. 2010. Metode Penelitian Pendidikan. Alfabeta, Bandung.
- [13]. A. Agrahri, D. Anand, dan J. Riedl. 2008. Can People Collaborate to Improve the Relevance of Search Results?. Institute of Technology University of Minnesota, Minnesota.