

**Implementasi *Context-Aware Recommender System* Berbasis *Ontology* untuk
Merekomendasikan Tujuan Wisata di Bandung
Raya**

Tugas Akhir

**Kelompok Keahlian: Modelling Computational
Experiment**

**Rizaldy Hafid Arigi
NIM: 1103130256**



**Program Studi Sarjana Teknik Informatika
Fakultas Informatika
Universitas Telkom
Bandung
2016**

Lembar Pengesahan

Implementasi *Context-Aware Recommender System* Berbasis
Ontology untuk Merekomendasikan Tujuan Wisata di Bandung
Raya

*Context-aware Recommender System Implementation Based on
Ontology for Recommending Tourism Destination at Greater
Bandung*

Rizaldy Hafid Arigi
NIM: 1103130256

Tugas Akhir ini diterima dan disahkan untuk memenuhi sebagian dari syarat
untuk memperoleh gelar sarjana Sains Komputasi
Program Studi Sarjana Teknik Informatika
Fakultas Informatika Universitas Telkom

Bandung, 11 November 2016
Menyetujui

Pembimbing 1

Pembimbing 2

Z.K. Abdurrahman Baizal, S.Si.,M.Kom
NIP: 99750047

Anisa Herdiani, S.T.,M.T
NIP: 15850002

Mengesahkan,
Kepala Program Studi Teknik Informatika

Dr. Ketua Program Studi
NIP: 95650581-1

Abstrak

Sektor pariwisata menunjukkan tren kenaikan untuk sumbangan devisa negara. Salah satu faktor kenaikan tersebut adalah semakin mudahnya mendapatkan informasi tujuan wisata di Indonesia melalui layanan Internet yang ada seperti TripAdvisor dan Instagram. Namun hingga saat ini, masih belum ada layanan Internet komersil yang mampu memberikan rekomendasi tujuan wisata yang terpersonalisasi.

Recommender system adalah teknik dan kaskas perangkat lunak yang digunakan untuk menyediakan rekomendasi item yang sesuai dengan kebutuhan pengguna. Untuk dapat menyediakan rekomendasi item sebagai keluaran sistem, tentu dibutuhkan masukan yang sesuai. Salah satu teknik yang dapat diterapkan untuk memperoleh masukan *recommender system* adalah *context-aware*. *Recommender system* dapat bersifat *context-aware* dengan memanfaatkan informasi kontekstual pengguna seperti lokasi dan cuaca.

Tugas akhir ini disusun melalui pengembangan dan pengujian *recommender system* yang bersifat *context-aware* dengan memanfaatkan masukan berupa lokasi pengguna, cuaca di sekitar tujuan wisata, waktu lokal dan preferensi pengguna terhadap kategori wisata, kemudian diproses dengan melakukan *reasoning* pada salah satu model objek dalam *knowledge-based*, yaitu *ontology*, sehingga menghasilkan rekomendasi tujuan wisata.

Kata Kunci: tujuan wisata, *recommender system*, *context-aware*, *knowledge-based*, *ontology*

Abstract

Tourism sector shows increasing contribution for country revenue. One of key factor is convenience to get Indonesia tourism information through internet service such as TripAdvisor and Instagram. However, those commercial internet services are not yet able to provide personalized tourism destination recommendation.

Recommender systems are software tools and technique providing suggestions for item recommendation based on user's needs. To provide item recommendation as system output, indeed suitable system input is needed. One of technique that can be implemented to get recommender system input is context-aware.

This final project is written through recommender system development and testing which has context-aware properties. This recommender system is making use of user location, weather around tourism destinations, local time and user preference towards tourism destination category. Those system inputs are processed using knowledge-based ontological reasoning and recommender system will propose tourism destination recommendation.

Keywords: tourism destination, recommender system, context-aware, knowledge-based, ontology

Lembar Persembahan

Alhamdulillah, setelah melalui perjalanan yang panjang akhirnya tugas akhir ini dapat selesai. Selama pengerjaan tugas akhir ini, penyusun banyak dibantu dengan doa, motivasi dan panduan dari berbagai pihak. Pada kesempatan ini, penyusun ingin mengucapkan terima kasih kepada:

1. Allah SWT, yang telah memberikan rahmat dan karunia-Nya kepada penyusun sehingga dapat menyelesaikan tugas akhir ini.
2. Kedua orang tua yang saya cintai, mama dan ayah yang selalu memberikan semangat dan dukungan. Semoga Allah tetap melimpahkan berkah kepada mereka.
3. Pak Baizal selaku pembimbing I dan bu Annisa selaku pembimbing II yang telah mendengarkan, membimbing dan memberikan panduan dikala penyusun menemui kesulitan. Terima kasih pak Baizal dan bu Annisa, semoga sukses selalu.
4. Pak Dana Kusumo selaku dosen wali, terima kasih atas masukan dan informasi yang diberikan kepada penyusun. Penyusun tidak akan lupa jasa bapak.
5. Kawan-kawan yang senantiasa mengisi "markas" Hardware and Embedded System Studio. Zavli, Fiqih, Bayu, kak Ari, kak Dick, kak Andra fatass, kak Kris, kak Niken. Kalian luar biasa!
6. Teman-teman satu kelas IF-37-02, yang selalu bersama-sama dari awal dan tetap bersama hingga akhir. Penyusun tidak akan melupakan kalian.

Kata Pengantar

Assalamualaikum wr, wb.

Alhamduillah, puja dan puji syukur saya panjatkan ke hadirat Allah SWT yang telah memberikan rahmat dan karunia-Nya sehingga penyusun dapat menyelesaikan tugas akhir dengan judul "Implementasi *Context-Aware Recommender System* Berbasis *Ontology* untuk Merekomendasikan Tujuan Wisata di Bandung Raya".

Dalam tugas akhir ini, penyusun bertujuan untuk mengimplementasikan dan menguji *recommender system* berbasis *ontology* yang mampu merekomendasikan tujuan wisata dengan memanfaatkan lokasi pengguna, waktu lokal, cuaca di tujuan wisata dan preferensi pengguna terhadap kategori wisata tertentu sebagai masukan untuk sistem. Penyusun berharap sistem yang dikembangkan dapat bermanfaat untuk menjadi sebuah purwarupa untuk *recommender system* yang akan dikembangkan di masa mendatang.

Semoga tugas akhir ini bermanfaat bagi siapapun yang membacanya.

Bandung, 15 Juli 2017

Rizaldy Hafid Arigi

Daftar Isi

Abstrak	i
Abstract	ii
Lembar Persembahan	iii
Kata Pengantar	iv
Daftar Isi	v
Daftar Gambar	vii
Daftar Tabel	viii
I Pendahuluan	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Tujuan Penulisan	2
1.4 Batasan Masalah	2
1.5 Rencana Kegiatan	3
1.6 Jadwal Kegiatan	3
II Kajian Pustaka	4
2.1 Wisata dan Pariwisata	4
2.2 Recommender System	4
2.3 Context-aware	5
2.4 Semantic Web dan Ontology	6
2.5 Android	7
2.6 Google Maps API	7
2.7 OpenWeatherMap API	8
2.8 Jython	8
2.9 Protokol HTTP	8
2.10 Javascript Object Notation	8
2.11 SPARQL	9

III Metodologi dan Desain Sistem	10
3.1 Pengembangan Ontology	10
3.1.1 <i>Specifcation</i>	10
3.1.2 <i>Conceptualization</i>	11
3.1.3 <i>Knowledge Acquisition</i>	19
3.1.4 <i>Implementation</i>	19
3.2 Metodologi Pengembangan Sistem	19
3.3 Model Rekomendasi Daya Tarik Wisata	20
3.3.1 Mendapatkan Masukan Preferensi Pengguna	20
3.3.2 Menyaring Daya tarik Wisata Berdasarkan Masukan	22
3.4 Desain Sistem	24
3.4.1 Arsitektur Sistem	24
3.4.2 Flowchart Sistem	24
3.4.3 Contoh Penghitungan	26
Daftar Pustaka	27
Lampiran	29

Daftar Gambar

2.1	Gambaran tahap-tahap <i>methontology</i>	7
3.1	diagram relasi daya tarik wisata dengan infrastruktur pendukung cuaca	14
3.2	Gambaran <i>concept tree</i> daya tarik wisata	18
3.3	GUI pada <i>frontend</i> untuk mendapatkan masukan tingkat ketertarikan dari pengguna	21
3.4	Gambaran komponen-komponen sistem dan relasinya	24
3.5	Gambaran alur kerja sistem ketika pengguna ingin mendapat rekomendasi daya tarik wisata	25
3.6	Gambaran alur kerja sistem aktivitas yang dilakukan server secara berkala	26

Daftar Tabel

1.1	Tabel rencana kegiatan	3
3.1	Tabel spesifikasi kebutuhan ontology yang akan dibangun	10
3.2	istilah dan definisi dalam domain kateogori wisata	14
3.3	Tabel spesifikasi relasi memiliki dukungan infrastruktur	14
3.4	Tabel spesifikasi atribut kode cuaca	15
3.5	Tabel spesifikasi atribut kode cuaca	15
3.6	Contoh tabel <i>instance</i> daya tarik wisata	15
3.7	Tabel contoh spesifikasi atribut senin buka	16
3.8	Tabel spesifikasi atribut latitude	16
3.9	Tabel spesifikasi atribut longitude	16
3.10	Contoh tabel <i>instance</i> daya tarik wisata	17
3.11	Tabel nilai numerik tingkat ketertarikan pengguna terhadap su- atu kategori	21

Bab I

Pendahuluan

1.1 Latar Belakang

Pariwisata telah menjadi sektor ekonomi yang penting di Indonesia. Hal tersebut terbukti ketika pada tahun 2014, sektor pariwisata menyumbang penerimaan devisa negara terbesar keempat setelah komoditas minyak, batu bara dan kelapa sawit. Pada tahun 2009, sektor pariwisata bahkan menempati peringkat ketiga melebihi batu bara[4]. Meskipun memang peringkat penyumbang devisa negara turun, sektor pariwisata berhasil meningkatkan sumbangan devisa dari USD 6,297 miliar di tahun 2010, hingga menjadi USD 11,16613 miliar di tahun 2014 [5].

Pada era teknologi informasi ini, mendapatkan informasi umum apapun yang diinginkan adalah hal yang mudah. Calon wisatawan mencari tahu informasi tujuan wisata secara manual dengan *search engine* atau media sosial merupakan hal yang sudah biasa. Layanan di Internet yang sangat populer untuk mendapatkan rekomendasi tujuan wisata diantaranya TripAdvisor dan Instagram. Namun, mendapatkan informasi tujuan wisata yang sesuai dengan preferensi pengguna melalui cara ini memiliki kelemahan, diantaranya adalah menghasilkan rekomendasi yang kurang terpersonalisasi.

Rekomendasi yang kurang terpersonalisasi terjadi karena sistem terlalu menitikberatkan pada tujuan wisata yang sangat populer atau sistem tidak tanggap dengan kondisi yang dihadapi pengguna. Untuk mengatasi hal ini, maka *recommender system* yang digunakan harus bersifat *context-aware*[2] dan menggunakan teknik *knowledge-based* untuk memberikan rekomendasi yang adil. Salah satu model representasi pada *knowledge-based* adalah *ontology*. *Ontology* memiliki keunggulan yaitu dapat merepresentasikan informasi kontekstual secara semantik dan kemudahan untuk menjelaskan bagaimana sistem dapat mencapai suatu kesimpulan. Namun, *ontology* sendiri tidak dapat melakukan *reasoning* dengan sendirinya karena *ontology* hanya merepresentasikan pengetahuan. Oleh karena itu dibutuhkan metode *reasoning* untuk mengambil kesimpulan dari representasi pengetahuan yang ada.

Pada prakteknya, *context-aware* cocok untuk diterapkan pada domain pariwisata. *Recommender system* yang bersifat *context-aware* dapat mendukung

wisatawan yang sedang bergerak dengan memberikan rekomendasi tujuan wisata yang cocok berdasarkan keadaan lingkungan wisatawan berada dan preferensi wisatawan tersebut[2]. Hal itu dapat dicapai dengan memanfaatkan lokasi pengguna, cuaca dan waktu saat itu [9].

Pada tugas akhir ini akan dilakukan pengembangan dan pengujian *recommender system* dengan teknik *knowledge-based* dan bersifat *context-aware* dengan memanfaatkan variabel lokasi pengguna, cuaca dan waktu serta preferensi kategori wisata yang ingin dikunjungi. Semua variabel masukan tersebut diproses dengan metode *reasoning* berdasarkan model *ontology* yang dibangun dan menghasilkan keluaran berupa rekomendasi tujuan wisata. Pengujian sistem dilakukan dari sisi akurasi rekomendasi dan persepsi pengguna.

1.2 Perumusan Masalah

1. Bagaimana model *ontology* yang tepat untuk merepresentasikan pengetahuan sistem pada domain pariwisata?
2. Bagaimana metode *reasoning* untuk merekomendasikan tujuan wisata?
3. Bagaimana skema pengujian dari sisi akurasi dan persepsi pengguna untuk *recommender system* yang bersifat *context-aware*?

1.3 Tujuan Penulisan

1. Membangun model *ontology* yang tepat untuk merepresentasikan pengetahuan sistem pada domain pariwisata.
2. Mengimplementasikan metode *reasoning* untuk merekomendasikan tujuan wisata.
3. Menguji performansi sistem dari sisi akurasi (melibatkan *domain expert*) dan persepsi pengguna.

1.4 Batasan Masalah

1. Sistem hanya merekomendasikan tujuan wisata di Bandung Raya, yang meliputi: Kota Bandung, Kabupaten Bandung, Kabupaten Bandung Barat dan Kabupaten Sumedang.
2. Data masukan hanya berupa lokasi pengguna, cuaca dan waktu di lokasi pengguna, serta preferensi kategori tujuan wisata pengguna.
3. Sistem hanya mengeluarkan keluaran berupa rekomendasi tujuan wisata.
4. *Mobile platform* yang digunakan adalah Android.
5. Pengguna diasumsikan sedang berada di Bandung Raya ketika menggunakan sistem yang dibangun.

1.5 Rencana Kegiatan

Kegiatan-kegiatan yang akan dilakukan selama proses penyusunan tugas akhir adalah sebagai berikut:

1. Desain sistem

Tahap ini menitikberatkan pada perancangan sisi *front-end*, *back-end*, dan representasi *ontology*.

2. Pengumpulan data lokasi wisata di Bandung.

Mengumpulkan data-data wisata di Bandung dengan metode observasi dan wawancara.

3. Implementasi sistem

Mengimplementasikan desain yang telah dibuat pada tahap desain sistem.

4. Pengujian sistem.

Melakukan dua jenis pengujian, satu bertujuan untuk mencari *bug* dan mendapatkan akurasi rekomendasi tujuan wisata dari domain ahli serta sisi persepsi pengguna.

5. Pembuatan laporan.

Tahap penyusunan laporan berdasarkan kaidah penulisan tugas akhir yang berlaku di institusi.

1.6 Jadwal Kegiatan

Berikut adalah tabel kegiatan dan bulan pelaksanaan.

No	Kegiatan	Januari	Februari	Maret	April	Mei	Juni
1	Desain sistem						
2	Pengumpulan data tujuan wisata						
3	Implementasi						
4	Pengujian						
5	Pembuatan laporan						

Tabel 1.1: Tabel rencana kegiatan

Bab II

Kajian Pustaka

2.1 Wisata dan Pariwisata

Wisata adalah kegiatan perjalanan yang dilakukan oleh seseorang atau sekelompok orang dengan mengunjungi tempat tertentu untuk tujuan rekreasi, pengembangan pribadi, atau mempelajari keunikan daya tarik wisata yang dikunjungi dalam jangka waktu sementara.

Pariwisata adalah berbagai macam kegiatan wisata dan didukung berbagai fasilitas serta layanan yang disediakan oleh masyarakat, pengusaha, pemerintah dan Pemerintah Daerah[11].

2.2 Recommender System

Recommender system (RS) adalah kakas dan teknik yang digunakan untuk mendapatkan sebuah saran atau rekomendasi item yang dapat digunakan oleh pengguna [13]. Rekomendasi item merupakan sebuah keluaran dari proses pembuatan keputusan. Item dapat berupa barang apa yang sebaiknya untuk dibeli, musik apa yang cocok dengan selera pengguna, dan masih banyak lagi.

RS telah banyak digunakan pada media sosial dan layanan-layanan multimedia, contohnya pada Youtube. Youtube adalah sebuah layanan berbagi video milik Google yang sangat populer. RS pada Youtube berperan untuk memberikan rekomendasi video berdasarkan aktivitas pengguna ketika menggunakan Youtube.

Ada tiga teknik yang paling sering digunakan untuk membangun RS, diantaranya adalah *content-based*, *collaborative filtering* dan *knowledge-based*.

Pada *content-based* RS belajar untuk menghasilkan rekomendasi item kepada pengguna berdasarkan apa yang pengguna sukai sebelumnya[12]. Ciri item akan dibandingkan dan dihitung untuk mengetahui asosiasi antar item. Contohnya jika pengguna memberi respon suka pada beberapa video mengenai komputer, maka RS akan memberikan rekomendasi berupa video yang berbicara mengenai pemrograman komputer.

Sedikit berbeda dengan *content-based*, *collaborative filtering* akan menghasilkan rekomendasi item kepada pengguna aktif berdasarkan apa yang pengguna lain sukai sebelumnya yang memiliki selera sama dengan pengguna ak-

tif tersebut[6]. Kemiripan selera dihitung berdasarkan kemiripan histori antar pengguna. Kedua pendekatan yang telah disebutkan memiliki kelemahan. Permasalahan teknik *content-based* adalah *cold start* dan *overspecialization*. *cold start* muncul ketika pengguna baru aktif di sistem dan RS belum memiliki data mengenai preferensi pengguna. Sedangkan *overspecialization* adalah masalah ketika RS memberi rekomendasi yang kemiripannya terlalu dekat dengan item sebelumnya yang disukai pengguna[3]. *Collaborative filtering* juga memiliki permasalahan *cold start*[13]. Permasalahan *cold start* dan *overspecialization* tidak dimiliki oleh *knowledge-based*.

Knowledge-based adalah teknik RS yang merekomendasikan suatu item berdasarkan suatu representasi pengetahuan bagaimana ciri item bisa memenuhi kebutuhan dan selera pengguna serta tingkat kegunaan item untuk pengguna. *Knowledge-based* menggunakan fungsi kesamaan yang mengestimasi seberapa cocok kebutuhan pengguna dengan rekomendasi yang akan diberikan tanpa memerlukan data histori apapun dari pengguna sehingga masalah *cold start* dan *over-specialization* tidak akan muncul. Selain tidak membutuhkan data histori, *knowledge-based* cocok untuk digunakan untuk pengembangan RS berdasarkan kasus dunia nyata.

2.3 Context-aware

Dalam bidang *ubiquitous computing*, konteks merupakan lokasi pengguna, identitas orang-orang disekitar pengguna, objek disekitarnya dan perubahan pada elemen-elemen tersebut [13]. Informasi kontekstual adalah hal yang sangat vital dalam perolehan rekomendasi yang relevan. Hal tersebut menjadikan *context-aware* merupakan salah satu pendekatan yang efektif dalam memahami kebutuhan pengguna.

Pendekatan penggunaan informasi kontekstual untuk proses rekomendasi dapat dikategorikan menjadi dua kelompok, yang pertama adalah rekomendasi yang menggunakan *context-driven querying* dan yang kedua adalah *contextual preference elicitation and estimation*[13]. Pendekatan pertama telah digunakan pada RS untuk pariwisata dan perangkat *mobile*. Sistem menggunakan pendekatan ini untuk mengekstrak informasi kontekstual langsung dari pengguna [15][1], contohnya dengan memberi pertanyaan bagaimana suasana hati pengguna saat ini, atau berdasar informasi lingkungan pengguna seperti lokasi, cuaca dan waktu untuk melakukan eksekusi *query* dan mendapatkan item yang paling cocok dengan pengguna. Sedangkan pendekatan kedua memodelkan dan mempelajari preferensi pengguna dengan *collaborative filtering*, *content-based*, atau teknik analisis data dari *machine learning*. Pada RS yang penyusun akan kembangkan, RS menggunakan pendekatan pertama karena pendekatan pertama cocok untuk RS dengan paradigma *knowledge-based* [13] dan kompleksitas komputasi yang lebih kecil dari pendketan kedua.

2.4 Semantic Web dan Ontology

World Wide Web Consortium (W3C) mendefinisikan *semantic web* sebagai kerangka kerja umum yang dapat memperkenankan data untuk bisa diberikan dan digunakan ulang di lintas aplikasi, perusahaan dan batasan komunitas. Secara khusus, teknologi *semantic web* menawarkan kemampuan menghubungkan data pada sebuah jaringan.

Ontology merupakan salah satu bagian penting dalam domain *semantic web*. *Ontology* adalah sebuah cara formal untuk merepresentasikan jaringan taksonomi dan klasifikasi serta mendefinisikan struktur pengetahuan. Pada tugas akhir ini, *ontology* akan digunakan untuk merepresentasikan pengetahuan kategori wisata. *Ontology* akan diimplementasikan dengan menggunakan bahasa khusus untuk membangun *ontology* yaitu *Web Ontology Language* (OWL).

Beberapa penelitian telah mengajukan suatu metode untuk merekonstruksi *ontology*. Salah satunya adalah *Methontology*[10]. Berikut adalah tahap pengembangan *ontology*[7]:

1. *Planning*

Merencanakan bagaimana suatu tugas (untuk langkah selanjutnya) dilakukan. Hal ini meliputi orang-orang yang terlibat, perangkat lunak dan perangkat keras.

2. *Spesification*

Mengidentifikasi tujuan *ontology*, ruang lingkup *ontology*, termasuk istilah didalamnya. keluaran dari tahap ini adalah spesifikasi *ontology* dalam bahasa manusia.

3. *Knowledge Acquisition*

Wawancara dengan ahli atau studi pustaka pada ruang lingkup terkait dapat membantu akuisisi pengetahuan secara signifikan.

4. *Conceptualization*

Data kunci pada suatu domain seperti konsep, *instance*, relasi kata kerja, sifat direpresentasikan secara informal (bahasa manusia).

5. *Formalization*

Formalisasi dari konsep yang telah dibangun dalam bahasa manusia menjadi konsep formal yang bisa diimplementasikan pada perangkat lunak.

6. *Integration*

Tahap untuk menentukan apakah *ontology* lain yang sudah ada dapat diintegrasikan dengan *ontology* yang dibangun.

7. Implementation

ontology direpresentasikan secara formal pada bahasa tertentu, contohnya OWL.

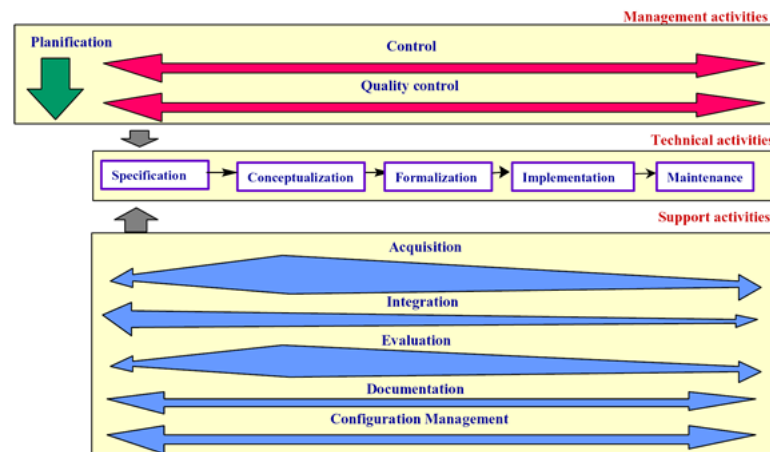
8. Evaluation

Tahap untuk mencari definisi yang salah pada *ontology* yang dibangun.

9. Documentation

10. Maintenance

Gambar 2.1 memvisualisasikan tahap-tahap yang dijabarkan diatas.



Gambar 2.1: Gambaran tahap-tahap *methontology*

2.5 Android

Android adalah sistem operasi untuk perangkat *mobile* yang dikembangkan oleh Google dengan memanfaatkan *kernel* dari sistem operasi Linux. Sistem operasi ini telah banyak digunakan pada *smartphone*, tablet pc, TV, *wearable device*, dan masih banyak lagi.

Android digunakan sebagai *platform* untuk sisi *front-end* karena kemudahan Android untuk mendapatkan data lokasi pengguna dibandingkan dengan menggunakan PC.

2.6 Google Maps API

Google Maps API (*Application Programming Interface*) adalah sebuah *interface* khusus yang dapat digunakan pengembang perangkat lunak sehingga aplikasi yang dikembangkan dapat menggunakan layanan, fitur dan basis data pada Google Maps.

Google Maps API dipilih karena kemudahan integrasi API pada pengembangan aplikasi Android.

2.7 OpenWeatherMap API

OpenWeatherMap adalah sebuah API yang menyediakan layanan untuk mendapatkan data cuaca terkini. OpenWeatherMap API memiliki dukungan untuk berbagai macam *platform* seperti Android dan iOS.

Pada tugas akhir ini, OpenWeatherMap akan digunakan untuk mendapatkan data cuaca terkini di lokasi pengguna.

2.8 Jython

Jython merupakan bahasa pemrograman Python yang terintegrasi dengan Java. Berbeda dengan Python biasa yang berjalan dengan *C compiler*, Jython berjalan dengan menggunakan Java Virtual Machine (JVM).

2.9 Protokol HTTP

Hypertext Transfer Protocol (HTTP) adalah protokol aplikasi yang menjadi dasar komunikasi data pada *World Wide Web*. *Hypertext* adalah teks dengan struktur tertentu yang dikirim diantara *node*. HTTP menjadi protokol agar bisa mengirim *Hypertext*.

HTTP berfungsi untuk memfasilitasi permintaan dan respon data pada model *client-server*. Pada contoh dunia nyata, *web browser* berperan sebagai *client* dengan mengirimkan pesan permintaan HTTP ke *server* yang berperan untuk *hosting*. *Server* akan menyediakan sumber daya yang disebutkan berdasarkan permintaan HTTP yang diterima dan mengirim respon HTTP kepada *client*.

HTTP memiliki beberapa metode permintaan seperti *GET*, *POST*, *DELETE*, *CONNECT* dan *TRACE*. Metode permintaan yang paling sering digunakan adalah *GET* dan *POST*. Pesan yang dikirim dalam permintaan berisi baris permintaan, *header* dan badan pesan (opsional). Sedangkan respon yang dikirim dengan HTTP berupa kode status permintaan, *header*, dan badan pesan.

Tugas akhir ini menggunakan HTTP sebagai perantara pertukaran pesan antara ponsel pengguna dengan proses yang berjalan pada *server*. HTTP dipilih karena HTTP dapat mengirim pesan dalam berbagai tipe konten seperti teks biasa, html, atau Javascript Object Notation (JSON).

2.10 Javascript Object Notation

Javascript Object Notation atau yang biasa disebut JSON adalah sebuah format pertukaran data. JSON memiliki kelebihan dapat dipahami manusia dengan mudah dan kemudahan *parsing* oleh mesin.

Tugas akhir ini menggunakan JSON sebagai format pertukaran data antara *client* dan *web server*.

2.11 SPARQL

SPARQL adalah bahasa *query* untuk basis data *semantic*. SPARQL memiliki kapabilitas membaca dan memanipulasi data yang tersimpan dalam format *Resource Description Framework*(RDF). Data dalam format RDF direpresentasikan dalam bentuk graf. Meskipun begitu, data RDF dapat dikategorikan sebagai database relasional SQL yang menggunakan tiga kolom yaitu subjek, predikat dan objek. SPARQL akan digunakan untuk menyimpan data *ontology* yang merepresentasikan pengetahuan sistem mengenai lokasi pariwisata Bandung.

Bab III

Metodologi dan Desain Sistem

3.1 Pengembangan Ontology

Dari sepuluh tahap *Methontology*, terdapat tiga tahap utama yang menghasilkan keluaran formal yaitu *specification*, *conceptualization* dan *implementation*.

3.1.1 *Specification*

Dalam tahap *specification* pengembangan *ontology*, terdapat tiga hal utama yang harus dideskripsikan yaitu domain, tujuan dan lingkup *ontology* yang dikembangkan.

Tahap ini akan menghasilkan dokumen spesifikasi kebutuhan *ontology* pada domain pariwisata. Tabel 3.1 adalah spesifikasi dari *ontology* yang akan dibangun.

Domain	Pariwisata
Tujuan	Ontology untuk merepresentasikan pengetahuan klasifikasi tujuan wisata dan aktivitas yang dapat dilakukan
Lingkup	Daftar kategori tujuan wisata, daftar tingkat kedekatan suatu tujuan wisata dengan kategori tujuan wisata, daftar kategori cuaca, daftar tujuan wisata, daftar atribut tujuan wisata.
Sumber	Wawancara dengan pakar. buku <i>Tourism Planning, An Integrated and Sustainable Development Approach</i> karya Edward Inskeep.

Tabel 3.1: Tabel spesifikasi kebutuhan ontology yang akan dibangun

Domain dari *ontology* yang akan dikembangkan berada di domain pariwisata, meliputi daya tarik wisata dan kategorinya. Untuk bisa mengetahui bagaimana daya tarik wisata dikategorikan, maka perlu adanya konsultasi dengan pakar terkait dan studi pustaka pada buku rujukan yang direkomendasikan oleh pakar.

3.1.2 *Conceptualization*

Ketika tahap *specification* telah selesai, maka akan didapat spesifikasi komponen *ontology* (kelas, text instance, atribut, relasi) apa saja yang harus tersedia dan bagaimana teknik mengakuisisi pengetahuan terhadap komponen tersebut. Pada tahap *conceptualization*, teknik-teknik untuk mengakuisisi pengetahuan diaplikasikan untuk mendapatkan istilah-istilah pada domain kategori pariwisata. Tabel 3.2 adalah tabel istilah preferensi kategori pariwisata.

Istilah	Keterangan
Daya tarik	Sebuah tempat yang dikunjungi oleh seseorang dengan harapan kebutuhannya yang terkait dengan mengisi waktu luang akan terpenuhi [8]. Klasifikasi secara umum membagi daya tarik menjadi tiga yaitu alam, budaya dan buatan.
Aksesibilitas	Sarana dan infrastruktur penunjang suatu aspek pada daya tarik / daya tarik wisata, contohnya jalan dan listrik.
Daya tarik alam	Daya tarik wisata yang ada karena terbentuk secara alami.
Daya tarik budaya	Daya tarik wisata yang ada karena aktivitas dan kebiasaan manusia.
Daya tarik buatan	Daya tarik wisata yang ada karena dibangun oleh manusia.
Petualangan alam	Subkelas dari daya tarik alam. Fenomena alam yang tidak biasa seperti geyser, gua, pemandian air panas, aktivitas gunung merapi merupakan daya tarik wisata yang penting untuk wisatawan yang ingin melihat-lihat atau melakukan aktivitas khusus seperti mendaki dan menjelajah gua. Pada kategori ini terdapat subkategori pemandian air panas, eksplorasi, pemandangan alam dan pendakian .
Flora dan fauna	Subkelas dari daya tarik alam. Konservasi hewan dan tumbuhan yang berada di habitatnya merupakan daya tarik wisata yang sangat menarik. Pada kategori ini terdapat subkategori konservasi dan safari .
Keindahan alam	Subkelas dari daya tarik alam. Merupakan motivasi utama dalam mengunjungi daya tarik alam. Pada kategori ini terdapat sub kategori <i>camping, hiking, outbound</i> dan pemandangan alam .

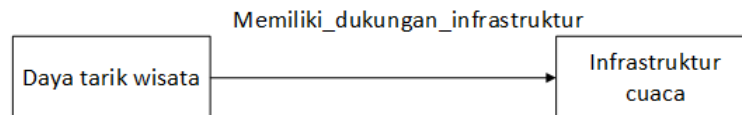
Pesisir dan perairan	Subkelas dari daya tarik alam. Sungai, danau dan area laut dapat menjadi daya tarik wisata untuk berbagai macam aktivitas dari mulai rekreasi hingga olahraga yang menantang seperti <i>diving</i> dan <i>rafting</i> . Pada kategori ini terdapat subkategori olahraga akuatik , relaksasi akuatik dan pemandangan alam .
Hiburan	Subkelas dari daya tarik buatan. Kategori hiburan diasosiasikan dengan seni pertunjukan kontemporer dan musik. Klub malam, karaoke dan kafe yang menyajikan musik masuk dalam kategori ini. Pada kategori ini terdapat subkategori hiburan malam , dan seni pertunjukan .
Event	Subkelas dari daya tarik buatan. Bangunan konferensi yang dapat digunakan untuk mengadakan acara besar seperti konser dan <i>event</i> merupakan daya tarik wisata yang semakin populer di seluruh dunia. Pada kategori ini terdapat subkategori konferensi dan stadion .
Kuliner	Subkelas dari daya tarik buatan. Pada kategori ini terdapat subkategori kuliner umum dan kuliner khas .
Olahraga	Subkelas dari daya tarik buatan. Fasilitas rekreasi dan olahraga dapat menjadi daya tarik wisata primer maupun sekunder untuk turis, seperti <i>golf course</i> , lapangan tenis, gelanggang olahraga dan lain-lain. Pada kategori ini terdapat subkategori olahraga akuatik , olahraga nonakuatik dan rekreasi .
Pusat belanja	Subkelas dari daya tarik buatan. Belanja adalah suatu aktivitas dan jenis pengeluaran pariwisata sehingga harus dianggap sebagai daya tarik wisata. Pada kategori ini terdapat subkategori mall , dan pasar modern .
Taman hiburan	Subkelas dari daya tarik buatan. <i>Theme park</i> adalah jenis taman hiburan yang memiliki suatu corak tema tertentu, seperti hewan, petualangan, fantasi dan menyajikan acara, wahana, dan pengalaman dalam suatu lingkungan terkendali [8]. <i>Theme park</i> cukup sulit dibedakan dari <i>amusement park</i> kecuali dari ada atau tidaknya tema tertentu yang digunakan. Namun keduanya tetap masih bisa disebut sebagai taman hiburan. Pada kategori ini terdapat subkategori taman hiburan .

Kegiatan ekonomi	Subkelas dari daya tarik budaya. Kegiatan ekonomi yang dilakukan penduduk lokal dapat menjadi daya tarik wisata tersendiri seperti perkebunan teh, pasar terapung diatar sungai Barito dan sebagainya. Pada kategori ini terdapat subkategori agrikultur dan pasar tradisional .
Fasilitas budaya	Subkelas dari daya tarik budaya. Berbagai fasilitas yang dibangun dengan tujuan menunjukkan ciri khusus budaya suatu daerah, contohnya seperti museum, pusat pembelajaran budaya, galeri seni, termasuk dalam kategori ini. Pada kategori ini terdapat subkategori wisata edukasi , museum sejarah dan museum budaya .
Seni	Subkelas dari daya tarik budaya. Pertunjukan seni dengan tarian, musik, drama dan pameran lukisan dapat menjadi daya tarik wisata yang penting jika bisa disajikan dengan efektif [8]. Selain pertunjukan seni, daerah tertentu yang memiliki nuansa budaya daerah pada arsitektur bangunan dan kerajinan tangan yang diproduksi juga termasuk dalam kategori ini. Pada kategori ini terdapat subkategori kerajinan , kesenian daerah dan suvenir .
Situs	Subkelas dari daya tarik budaya. Peninggalan-peninggalan budaya dan sejarah seperti bangunan, tempat ibadah, monumen, dan lokasi peristiwa sejarah merupakan daya tarik wisata yang memiliki nilai sejarah dan budaya. Situs yang memiliki nilai arkeologis juga termasuk dalam kategori ini. Pada kategori ini terdapat subkategori religi , situs budaya , dan situs sejarah .
Urban	Subkelas dari daya tarik budaya. Area urban dengan arsitektur bangunan, sudut kota, pusat publik, fasilitas belanja dan aktivitas di suatu jalan yang beragam dapat menciptakan daya tarik wisata tersendiri. Segala hal menarik di bagian area urban yang dapat menimbulkan rasa tertarik masuk ke dalam kategori ini. Pada kategori ini terdapat subkategori kuliner khas , seni pertunjukan , rute kota , dan taman kota .
Lokasi	Informasi lokasi pada umumnya menggunakan alamat, namun untuk kemudahan proses komputasi, lokasi lebih mudah direpresentasikan dengan <i>latitude</i> dan <i>longitude</i> berdasarkan koordinat Google Maps.

Waktu buka	Setiap destinasi wisata memiliki waktu bukanya tersendiri. Waktu buka bisa berbeda setiap harinya.
Infrastruktur	Cuaca disekitar destinasi wisata harus diperhitungkan agar daya tarik wisata dapat dinikmati secara penuh. Infrastruktur daya tarik wisata dapat mendukung agar daya tarik wisata dapat dikunjungi hingga kondisi cuaca tertentu. Kondisi cuaca pada tugas akhir ini dinyatakan dengan tiga tingkat yaitu buruk , kurang baik dan baik .

Tabel 3.2: istilah dan definisi dalam domain kateogori wisata

Setiap daya tarik wisata akan direpresentasikan sebagai satu *instance*. Satu *instance* daya tarik wisata pasti memiliki dukungan infrastruktur untuk tingkat cuaca tertentu. Kelas infrastruktur cuaca memiliki akan memiliki tiga *instance* yaitu **cuaca buruk**, **cuaca kurang baik** dan **cuaca baik**. *Instance* daya tarik wisata pasti akan memiliki satu properti relasi ke *instance* dari infrastruktur cuaca. Hubungan tersebut divisualisasikan pada gambar 3.1.



Gambar 3.1: diagram relasi daya tarik wisata dengan infrastruktur pendukung cuaca

Spesifikasi relasi **memiliki dukungan infrastruktur** dapat dilihat pada tabel 3.3.

Spesifikasi	Nilai
Nama relasi	memiliki dukungan infrastruktur
Ekuivalen	cuaca cocok
Domain konsep	Daya tarik wisata
Kardinalitas	(1,1)
Target konsep	Infrastruktur cuaca

Tabel 3.3: Tabel spesifikasi relasi memiliki dukungan infrastruktur

Setiap *instance* dari kelas infrastruktur cuaca harus dapat menjabarkan kondisi cuaca apa saja yang direpresentasikan, misalnya cuaca cerah, berawan

dsb. Kondisi cuaca dapat direpresentasikan dengan kode cuaca dari OpenWeatherMap. Kode cuaca dapat dilihat di .Berikut adalah tabel detail dari atribut kode cuaca:

Spesifikasi	Keterangan
Nama atribut	kode cuaca
Tipe nilai	integer
Satuan	–
Rentang	[500, 1000]

Tabel 3.4: Tabel spesifikasi atribut kode cuaca

Selain kode cuaca, *instance* dari kelas infrastruktur cuaca juga memiliki atribut untuk menyatakan kondisi cuaca menjadi nilai numerik yang dapat digunakan pada proses komputasi. Hal ini penting pada proses pemberian nilai *utility* daya tarik wisata. Berikut adalah tabel detail dari atribut nilai cuaca:

Spesifikasi	Keterangan
Nama atribut	nilai cuaca
Tipe nilai	integer
Satuan	–
Rentang	[1, 3, 5]

Tabel 3.5: Tabel spesifikasi atribut kode cuaca

Sebagai contoh berikut adalah detail dari salah satu *instance* infrastruktur cuaca, yaitu cuaca bagus.

Spesifikasi	Keterangan
Nama instance	cuaca bagus
kode cuaca	801, 802, 803, 804, 952, 953
nilai cuaca	5

Tabel 3.6: Contoh tabel *instance* daya tarik wisata

Semua daya tarik wisata pasti memiliki keterangan waktu buka. Setiap hari terdapat perubahan status daya tarik wisata, apakah sedang buka atau sedang tutup. Karena setiap hari terdapat dua status tersebut dan ada tujuh hari dalam seminggu, maka paling tidak ada empat belas data waktu perubahan

Spesifikasi	Keterangan
Nama atribut	Senin buka
Tipe nilai	float
Satuan	–
Rentang	[0, 1]

Tabel 3.7: Tabel contoh spesifikasi atribut senin buka

status daya tarik wisata. Tabel 3.7 adalah contoh spesifikasi atribut waktu perubahan status daya tarik wisata.

Nilai yang disimpan pada atribut waktu perubahan status tidak menggunakan format waktu HH:mm, namun dengan nilai *float* antara 0 hingga 1. Hal itu dilakukan untuk kemudahan proses komputasi dalam pemilihan daya tarik wisata yang sedang buka. Untuk mengkonversi data waktu perubahan status dalam format HH:mm ke nilai antara nol hingga satu, digunakan persamaan 3.1:

$$tn = \frac{60H + m}{1440} \quad (3.1)$$

Dengan tn adalah nilai hasil transformasi ke rentang nol dan satu, H adalah jam dan m adalah menit.

Setiap *instance* daya tarik wisata pasti memiliki atribut *latitude* dan *longitude*, berikut adalah spesifikasi atribut *latitude* dan *longitude*.

Spesifikasi	Keterangan
Nama atribut	latitude
Tipe nilai	float
Satuan	–
Rentang	[-90, 90]

Tabel 3.8: Tabel spesifikasi atribut latitude

Spesifikasi	Keterangan
Nama atribut	longitude
Tipe nilai	float
Satuan	–
Rentang	[-180, 180]

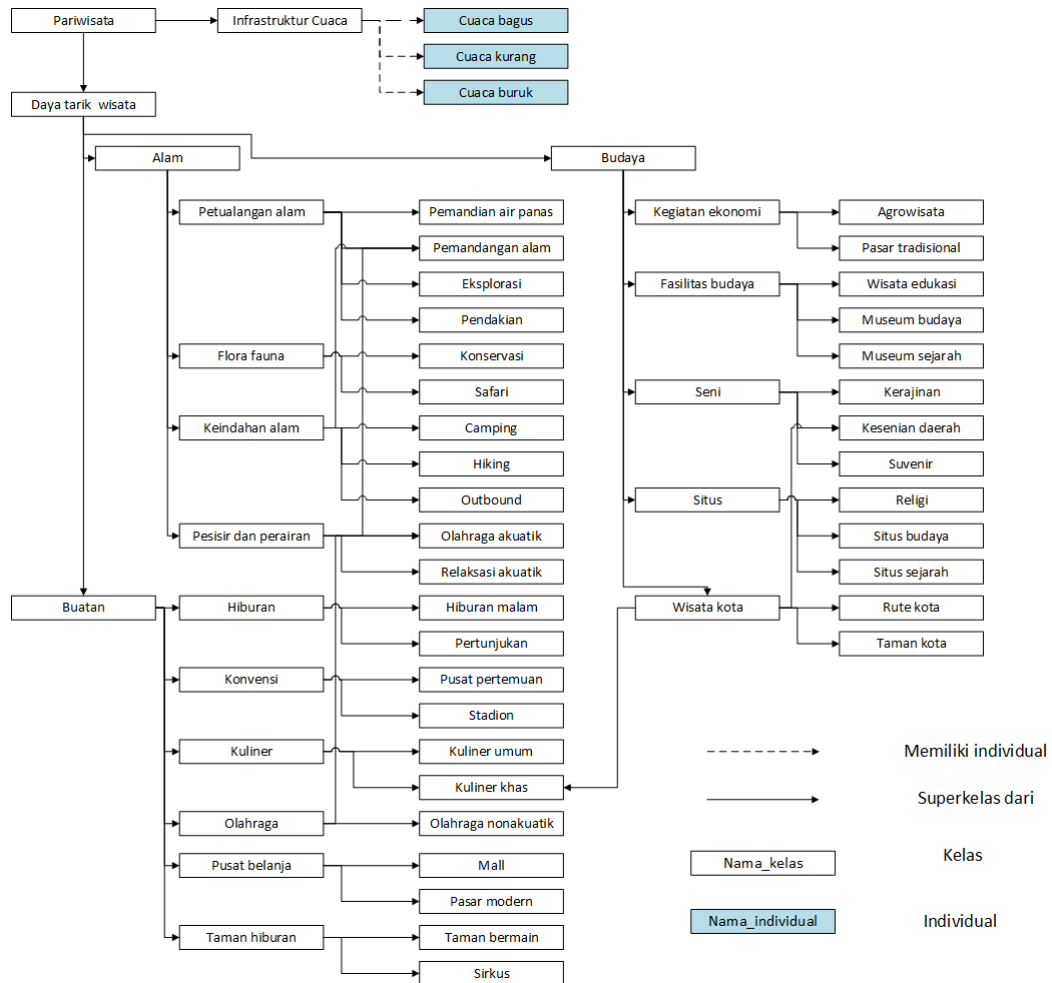
Tabel 3.9: Tabel spesifikasi atribut longitude

Spesifikasi relasi dan atribut-atribut yang telah dijabarkan sebelumnya berfungsi untuk validasi ketika menetapkan nilai-nilai atribut dan pembuatan relasi pada suatu *instance* daya tarik wisata. Tabel 3.10 merupakan contoh detail *instance* daya tarik wisata.

Spesifikasi	Keterangan
Nama instance	alam wisata cimahi
Latitude	-6.841073f
Longitude	107.54805f
Memiliki dukungan infrastruktur	cuaca kurang baik
Senin buka	0.3334f
Senin tutup	0.875f
—	—
Minggu tutup	0.875f

Tabel 3.10: Contoh tabel *instance* daya tarik wisata

Untuk mendapatkan gambaran jelas dari hubungan antara semantik konsep istilah yang didapatkan, gambar 3.2 memvisualisasikan hubungan tersebut.



Gambar 3.2: Gambaran *concept tree* daya tarik wisata

3.1.3 *Knowledge Acquisition*

Tahap ini merupakan tahap independen dalam pengembangan *ontology*, namun tetap dilaksanakan bersamaan dengan tahap lain. Untuk domain pariwisata, sebagian besar pengetahuan diperoleh pada awal pengembangan *ontology*. Ada empat aktivitas yang dilakukan ketika berada di tahap *knowledge acquisition*, yaitu:

- Mengadakan wawancara dengan pakar. Wawancara awal ini diadakan dengan tujuan dapat memperoleh ikhtisar dari domain pariwisata.
- Mempelajari literatur yang berhubungan dengan domain pariwisata. Hal ini dapat mengurangi waktu yang diperlukan dari pakar untuk memberikan instruksi ketika memperoleh konsep domain pariwisata.
- Setelah memperoleh pengetahuan mengenai domain pariwisata, pencarian konsep bergerak dari umum hingga mengerucut terutama mengenai kategori daya tarik pariwisata.
- Konfirmasi konsep domain pariwisata oleh pakar.

3.1.4 *Implementation*

Desain *ontology* diimplementasikan dengan menggunakan *framework* khusus untuk membangun aplikasi *knowledge-based* yaitu Protege. Protege memiliki beberapa kelebihan, diantaranya:

- *Ontology* yang dibangun dengan Protege memenuhi standar W3C,
- UI sederhana
- dukungan visualisasi

3.2 Metodologi Pengembangan Sistem

Tugas akhir ini akan disusun dengan melalui tahap-tahap berikut:

1. Pengumpulan Data Lokasi Wisata Bandung

Tujuan tahap ini adalah mendapatkan data lokasi wisata. Metode yang dapat dilakukan adalah:

- (a) Pencarian data di Google, TripAdvisor dan Foursquare
- (b) Pengajuan permintaan data ke Dinas Kebudayaan dan Pariwisata Jawa Barat

2. Analisis Sistem

Beberapa kebutuhan perangkat lunak yang harus dipenuhi dalam membangun *recommender system* ini adalah:

- (a) Gambaran sistem secara umum
- (b) *Graphical user interface*
- (c) *Software interface*
- (d) Basis data

Pada tahap ini juga dilakukan penerapan pengembangan *ontology* dengan *Methontology*.

3. Implementasi Sistem

Tahap implementasi meliputi:

- (a) Pengembangan aplikasi Android sebagai sisi *client*.
- (b) Pengembangan *backend*.
- (c) Implementasi komunikasi antara *client* dan *server*.
- (d) Implementasi *ontology* dengan Protege.

4. Pengujian

Tahap pengujian memiliki dua jenis pengujian:

- (a) Pengujian untuk menemukan tingkat kepuasan dari sisi pengguna
- (b) Pengujian akurasi sistem dengan melibatkan pakar.

5. Pembuatan Laporan

Laporan akhir akan ditulis sesuai dengan kaidah dan ketentuan yang berlaku di institusi.

3.3 Model Rekomendasi Daya Tarik Wisata

Sistem yang dikembangkan memiliki masukan berupa preferensi pengguna terhadap kategori daya tarik wisata tertentu dan *contextual factor* seperti lokasi pengguna, cuaca disekitar daya tarik wisata serta waktu lokal. Sistem akan memberikan sepuluh rekomendasi tujuan wisata paling sesuai berdasarkan masukan yang diterima.

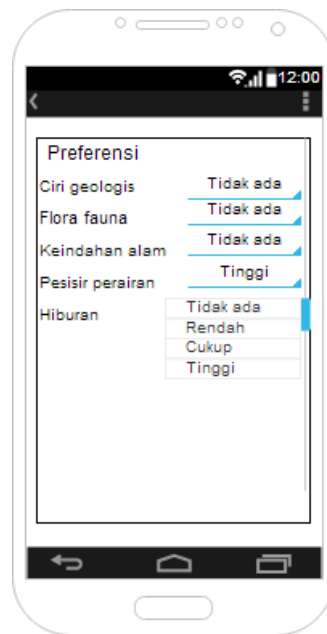
3.3.1 Mendapatkan Masukan Preferensi Pengguna

Untuk mendapatkan preferensi pengguna terhadap suatu kategori daya tarik wisata, pengguna dapat menyatakan secara eksplisit ketertarikan terhadap suatu kategori daya tarik wisata. Ada empat tingkat ketertarikan pengguna, yaitu tidak ada, rendah, cukup tinggi dan tinggi. Setiap tingkat ketertarikan memiliki nilai bobot tersendiri. Berikut adalah nilai bobot dari setiap tingkat ketertarikan.

Pengguna memilih tingkat ketertarikan yang sesuai melalui tampilan GUI berikut:

Tingkat	Nilai bobot
Tidak ada	0
Rendah	3
Cukup tinggi	6
Tinggi	9

Tabel 3.11: Tabel nilai numerik tingkat ketertarikan pengguna terhadap suatu kategori



Gambar 3.3: GUI pada *frontend* untuk mendapatkan masukan tingkat ketertarikan dari pengguna

Semua kategori daya tarik wisata merupakan subkelas langsung dari kelas alam, budaya dan buatan. Pengguna dapat mengetahui keterangan kategori daya tarik wisata dengan cara *tap* teks kategori.

Aplikasi *frontend* akan mengirimkan lokasi dan tingkat ketertarikan pengguna melalui protokol HTTP. Format data yang dikirimkan sebagai HTTP request adalah sebagai berikut:

```
{
  "location":{
    "latitude":float ,
    "longitude":float ,
  }
  "preferences":{
    "Ciri_geologis":int ,
```

```

    "Flora_fauna":int,
    "Keindahan_alam":int,
    "Pesisir_perairan":int,
    "Hiburan":9,
    "Konvensi":int,
    "Olahraga":int,
    "Pusat_belanja":int,
    "Taman_hiburan":int,
    "Ekonomi":int,
    "Fasilitas_budaya":int,
    "Seni":int,
    "Situs":int,
    "Urban":int
  }
}

```

Nilai pada setiap *field* di *preferences* merupakan nilai numerik dari tingkat ketertarikan pengguna terhadap suatu kategori.

3.3.2 Menyaring Daya tarik Wisata Berdasarkan Masukan

Terdapat tiga tahap dalam memproses masukan *recommender system* hingga pengguna mendapat rekomendasi daya tarik wisata. Kedua tahap itu adalah:

1. Memilih daya tarik wisata yang buka

Untuk memilih daya tarik wisata yang sedang buka, *Context Aware API Server* mengeksekusi SPARQL *query* terhadap objek *ontology*. Sebelum mengeksekusi *query*, *Context Aware API Server* harus mengetahui jam dan menit saat itu. Kemudian ilai jam dan menit dikonversikan dengan persamaan 3.1.

Seleksi daya tarik wisata dilakukan berdasar atribut waktu perubahan status. Pemilihan atribut dilakukan berdasar hari eksekusi *query*. Berikut adalah contoh *query* untuk memilih daya tarik wisata yang sedang buka pada hari senin pukul dua belas siang

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX ns: <http://www.semanticweb.org/dell/ontologies/2017/4/tourism#>

SELECT ?destination
WHERE {?destination ns:senin_buka ?open; ns:senin_tutup ?closed
FILTER( (?open < 0.5 && 0.5 < ?closed)

```


|| ((?closed > 0.5 || ?open < 0.5) && (?closed < ?open)))}

Hasil dari *query* diatas adalah daftar nama *instance* daya tarik wisata yang buka hari senin pukul dua belas siang.

2. **Memberi nilai *utility*** Nilai *utility* setiap daya tarik wisata ditentukan dengan persamaan berikut:

$$util_i = e^{-(d_i p_i + w_i)} \quad (3.2)$$

Utility ditentukan dengan fungsi eksponen negatif dari d_i sebagai jarak antara pengguna dengan daya tarik wisata ke i , p_i sebagai nilai *dissimilarity* antara preferensi pengguna dengan kategori daya tarik wisata ke i dan w_i sebagai selisih nilai cuaca aktual di daya tarik wisata ke i dengan nilai cuaca yang cocok untuk mengunjungi daya tarik wisata tersebut. Nilai d_i didapatkan dengan persamaan *euclidean*:

$$d_i = \sqrt{(l_1 - x_i)^2 + (l_2 - y_i)^2} \quad (3.3)$$

dengan l_1 adalah *latitude* dan l_2 adalah *longitude* pengguna. Sedangkan x_i adalah *latitude* dan y_i adalah *longitude* dari daya tarik wisata ke i .

Sedangkan untuk menghitung nilai *dissimilarity* antara preferensi pengguna terhadap kategori daya tarik wisata ke i dapat menggunakan persamaan 3.4:

$$p_i = - \sum_{j=1}^{|m|} m_j n_{ij} \quad (3.4)$$

Sementara m merupakan vektor tingkat ketertarikan pengguna terhadap kategori daya tarik wisata dan n_{ij} adalah hubungan antara daya tarik wisata ke i dengan kategori ke j . Nilai hubungan tersebut akan selalu bernilai lebih dari sama dengan 0.

Karena setiap daya tarik wisata tidak menjadi instance langsung dari kategori, melainkan menjadi instance dari subkategori yang diagregasi oleh suatu kategori, maka untuk menghitung n_{ij} digunakan persamaan 3.5:

$$n_{ij} = \sum_{k=1}^{|o|} o_{ik} \quad (3.5)$$

dengan o_{ik} adalah nilai hubungan antara daya tarik wisata ke i dengan subkategori ke k yang menjadi subkelas dari kategori ke j . Nilai hubungan menjadi satu jika *instance* daya tarik wisata memang *instance* dari subkategori tersebut dan nol jika bukan.

Sedangkan untuk menilai seberapa layak daya tarik wisata dikunjungi berdasarkan cuaca menggunakan persamaan 3.6:

$$w_i = \max(0, r_i - s_i) \quad (3.6)$$

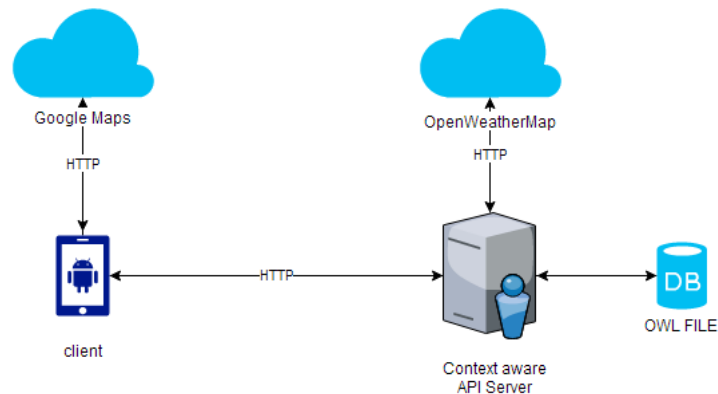
dengan r_i adalah nilai numerik dari kondisi cuaca yang membuat daya tarik wisata layak dikunjungi dan s_i adalah nilai numerik dari kondisi cuaca terkini di daya tarik wisata. Nilai s_i didapatkan dari lokasi acuan cuaca yang paling terdekat dengan daya tarik wisata tersebut.

3.4 Desain Sistem

Sebelum masuk pada fase pengembangan sistem, menentukan desain kasar sistem merupakan hal yang krusial. Hal tersebut disebabkan karena memahami komponen apa saja yang ada pada sistem dan relasi antar komponen dapat membantu menentukan arah pengembangan sistem.

3.4.1 Arsitektur Sistem

Gambar 3.4 menjelaskan desain arsitektur sistem dan relasi-relasi antar komponen.

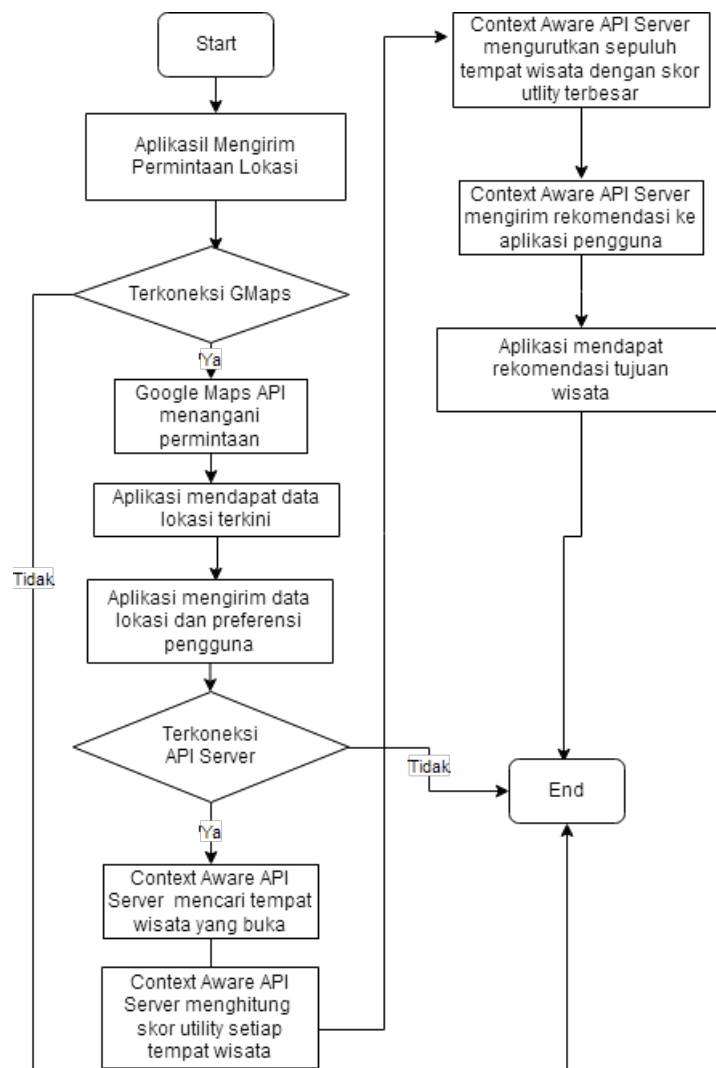


Gambar 3.4: Gambaran komponen-komponen sistem dan relasinya

Terdapat empat lima komponen penting dari sistem yang dikembangkan, yaitu *client*, *server*, *Google Maps API* dan *Open Weather Map API*.

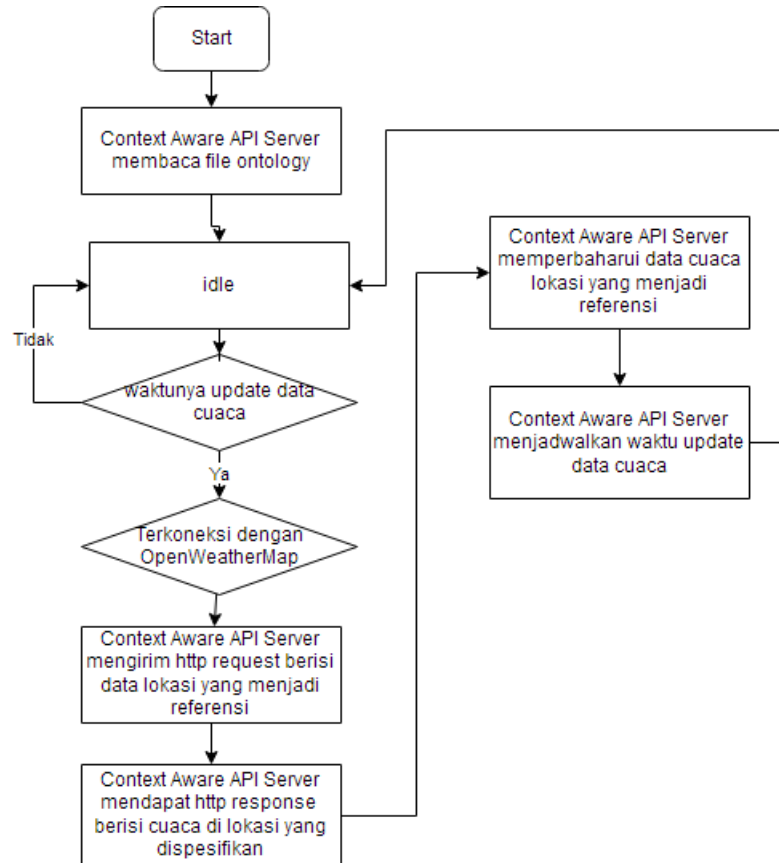
3.4.2 Flowchart Sistem

Secara umum, aliran kerja sistem dijelaskan pada gambar 3.5 dibawah ini: *Flowchart* sistem pada gambar 3.5 dapat dijelaskan secara umum sebagai tiga kegiatan utama. Tiga kegiatan tersebut adalah mendapatkan alur mendapatkan informasi kontekstual dan preferensi tujuan wisata pengguna, melakukan *reasoning* di *web server* berdasar data yang masuk, dan mendapatkan rekomendasi tujuan wisata.



Gambar 3.5: Gambaran alur kerja sistem ketika pengguna ingin mendapat rekomendasi daya tarik wisata

Context Aware API Server memiliki dua *thread* yang dijalankan bersamaan. *Thread* pertama untuk mendengarkan permintaan rekomendasi tujuan wisata dan yang kedua untuk memperbaharui data cuaca di lokasi yang menjadi acuan kondisi cuaca.



Gambar 3.6: Gambaran alur kerja sistem aktivitas yang dilakukan server secara berkala

3.5 Contoh Penghitungan

Daftar Pustaka

- [1] ABOWD, G. D., ATKESON, C. G., HONG, J., LONG, S., KOOPER, R., AND PINKERTON, M. Cyberguide: A mobile context-aware tour guide. *Wireless networks* 3, 5 (1997), 421–433.
- [2] ALHAZBI, S., LOTFI, L., ALI, R., AND SUWAILIH, R. Ontology-based model in tourism context-aware systems. In *2013 International Conference on ICT Convergence (ICTC)* (2013), IEEE, pp. 775–779.
- [3] BLANCO-FERNÁNDEZ, Y., PAZOS-ARIAS, J. J., GIL-SOLLA, A., RAMOS-CABRER, M., LÓPEZ-NORES, M., GARCÍA-DUQUE, J., FERNÁNDEZ-VILAS, A., DÍAZ-REDONDO, R. P., AND BERMEJO-MUÑOZ, J. A flexible semantic inference methodology to reason about user preferences in knowledge-based recommender systems. *Knowledge-Based Systems* 21, 4 (2008), 305–320.
- [4] BPS. Ranking devisa pariwisata terhadap komoditas ekspor lainnya tahun 2004-2009. http://www.budpar.go.id/filedata/5436_1695-Rankingdevisa.pdf, 2009. accessed 5 November 2016.
- [5] BPS. Jumlah wisatawan asing tahun 1974 - 2014, devisa wisman 1998-2014 dan tingkat penghunian kamar (tpk) hotel bintang dan non bintang tahun 1985 - 2014. <https://www.bps.go.id/linkTabelStatis/view/id/1393>, 2016. accessed 5 November 2016.
- [6] CASTILLO, L., ARMENGOL, E., ONAINDÍA, E., SEBASTIÁ, L., GONZÁLEZ-BOTICARIO, J., RODRÍGUEZ, A., FERNÁNDEZ, S., ARIAS, J. D., AND BORRAJO, D. Samap: An user-oriented adaptive system for planning tourist visits. *Expert Systems with Applications* 34, 2 (2008), 1318–1332.
- [7] FERNÁNDEZ-LÓPEZ, M., GÓMEZ-PÉREZ, A., AND JURISTO, N. Methontology: from ontological art towards ontological engineering.
- [8] INSKEEP, E., ET AL. *Tourism planning: an integrated and sustainable development approach*. Van Nostrand Reinhold, 1991.

- [9] JAKKHUPAN, W., RATTANOTHAI, S., AND KAEWONJING, J. A context-aware personalized venue recommender system on smart device. In *Wireless and Mobile (APWiMob), 2015 IEEE Asia Pacific Conference on* (2015), IEEE, pp. 202–207.
- [10] JONES, D., BENCH-CAPON, T., AND VISSER, P. Methodologies for ontology development.
- [11] KEMENTERIAN PARIWISATA REPUBLIK INDONESIA. Undang-undang ri no 10 tahun 2009 tentang kepariwisataan, 2009.
http://www.kemenpar.go.id/userfiles/file/4636_1364-UUTentangKepariwisataaannet1.pdf.
- [12] MORENO, A., VALLS, A., ISERN, D., MARIN, L., AND BORRÀS, J. Sigtur/e-destination: ontology-based personalized recommendation of tourism and leisure activities. *Engineering Applications of Artificial Intelligence* 26, 1 (2013), 633–651.
- [13] RICCI, F., ROKACH, L., AND SHAPIRA, B. *Introduction to recommender systems handbook*. Springer, 2011.
- [14] SUI, L. Strategy analytics: Android captures record 88 percent share of global smartphone shipments in q3 2016. <http://qz.com/826672/android-goog-just-hit-a-record-88-market-share-of-all-smartphones/>, 2016. accessed 5 November 2016.
- [15] VAN SETTEN, M., POKRAEV, S., AND KOOLWAAIJ, J. Context-aware recommendations in the mobile tourist application compass. In *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems* (2004), Springer, pp. 235–244.

Lampiran