



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KI141502

**SKEMA METADATA UNTUK DOKUMENTASI
PENINGGALAN WARISAN BUDAYA YANG
BERTAUTAN DENGAN GEO LINKED OPEN
DATA**

MOCHAMAD BAKHTIAR H
NRP. 5112 100 120

Dosen Pembimbing 1
Dr.Tech. Ir. Raden Venantius Hari Ginardi, M.Sc.

Dosen Pembimbing 2
Nurul Fajrin A.,S.Kom., M.Sc.

JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya
2016



TUGAS AKHIR - KI141502

**SKEMA METADATA UNTUK DOKUMENTASI
PENINGGALAN WARISAN BUDAYA YANG
BERTAUTAN DENGAN *GEO LINKED OPEN
DATA***

**MOCHAMAD BAKHTIAR H
NRP. 5112 100 120**

**Dosen Pembimbing 1
Dr.Tech. Ir. Raden Venatius Hari Ginardi, M.Sc.**

**Dosen Pembimbing 2
Nurul Fajrin A.,S.Kom., M.Sc.**

**JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya
2016**

(Halaman ini sengaja dikosongkan)



FINAL PROJECT - KI141502

**METADATA SCHEMA FOR THE
DOCUMENTATION OF CULTURAL HERITAGE
RELATED TO GEO LINKED OPEN DATA**

**MOCHAMAD BAKHTIAR H
NRP. 5112 100 120**

**Supervisor 1
Dr.Tech. Ir. Raden Venatius Hari Ginardi, M.Sc.**

**Supervisor 2
Nurul Fajrin A.,S.Kom., M.Sc.**

**DEPARTMENT OF INFORMATICS
Faculty of Information Technology
Sepuluh Nopember Institute of Technology
Surabaya
2016**

(Halaman ini sengaja dikosongkan)

LEMBAR PENGESAHAN

SKEMA METADATA UNTUK DOKUMENTASI PENINGGALAN WARISAN BUDAYA YANG BERTAUTAN DENGAN *GEO LINKED OPEN DATA*

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Manajemen Informasi
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

MOCHAMAD BAKHTIAR H

NRP: 5112 100 120

Disetujui oleh Dosen Pembimbing Tugas Akhir:

1. Dr. Tech. Ir. Raden Venantius Hari Ginardi, S.T., M.Sc.
NIP: 19650518 199203 1 003 (Pembimbing 1)
2. Nurul Fajrin Ariyani, S.Kom., M.
NIP. 19860722 201504 2 003 (pembimbing 2)



**SURABAYA
JUNI, 2016**

(Halaman ini sengaja dikosongkan)

**SKEMA METADATA UNTUK DOKUMENTASI
PENINGGALAN WARISAN BUDAYA YANG
BERTAUTAN DENGAN *GEO LINKED OPEN DATA***

Nama : Mochamad Bakhtiar H
NRP : 5112100120
Jurusan : Teknik Informatika
Fakultas Teknologi Informasi ITS
Dosen Pembimbing I : Dr. Tech. Ir. Raden Venantius Hari
Ginardi, M.Sc.
Dosen Pembimbing II : Nurul Fajrin A.,S.Kom., M.Sc.

ABSTRAK

Peninggalan Warisan budaya adalah peninggalan yang berupa benda maupun non benda yang merupakan jati diri suatu masyarakat yang diwariskan dari generasi sebelumnya dan yang dilestarikan untuk generasi yang akan datang. Warisan budaya dapat berupa benda seperti candi, Museum, artefak, fosil, prasasti, atau non benda seperti tradisi, ritual.

Banyak usaha untuk melestarikan peninggalan warisan budaya, salah satunya yaitu membuat dokumentasinya. Pada tugas akhir ini dibuat skema metadata untuk membantu dalam dokumentasi peninggalan warisan budaya. Hasil pembuatan skema metadata akan ditautkan dengan Geo Linked Open Data untuk menambahkan informasi lokasi secara lengkap ditemukannya peninggalan warisan budaya. Dari skema tersebut akan ditampilkan kepada user daftar peninggalan warisan budaya yang telah terdokumentasi dalam skema metadata melalui aplikasi berbasis web.

Data geospasial dari data artefak berhasil ditautkan dengan geonames dan Wikipedia melalui aplikasi. Pada pengujian yang telah dilakukan, dari 50 data artefak sebanyak 50 data geospasial provinsi berhasil ditautkan dengan geonames dan Wikipedia, 50 data geospasial kabupaten/kota berhasil ditautkan dengan geonames dan sebanyak 29 data yang berhasil ditautkan

dengan Wikipedia, Serta 29 Data Geospasial Kecamatan berhasil ditautkan dengan geonames.

Kata kunci: Geo Linked Open Data, Metadata.

**METADATA SCHEMA FOR DOCUMENTATION
CULTURAL HERITAGE RELATED TO GEO LINKED
OPEN DATA**

Name : Mochamad Bakhtiar H
NRP : 5112100120
Department : Department of Informatics
Faculty of Information Technology ITS
Supervisor I : Dr. Tech. Ir. Raden Venantius Hari
Ginardi, M.Sc.
Supervisor II : Nurul Fajrin A.,S.Kom., M.Sc.

ABSTRACT

Cultural heritage is a heritage in the form of objects or non object, the identity of a society inherited from previous generations and preserved for generations to come. Cultural heritage can be objects such as temples, Museumts, artifacts, fossils, inscription, or non objects such as tradition, ritual.

Many attempts to preserve the cultural heritage, one of which is to make its documentation. In this final project create metadata scheme to assist in the documentation of cultural heritage. Results of create metadata schemes will be linked with Geo Linked Open Data to add location information completely discovery of cultural heritage. From the scheme will be displayed to the user a list of heritage cultural heritage has been documented in metadata schemes through web-based applications.

Geospatial data from artifact data has succesfull linked to the GeoNames and Wikipedia via the application. In the testing that was done, from 50 data artifacts as much as 50 geospatial province successfully linked with GeoNames and Wikipedia, 50 of geospatial data regencies / cities have successfully linked with GeoNames and as many as 29 data link successfully with

Wikipedia, as well as 29 Geospatial Data Subdistrict successfully linked with GeoNames.

Key words: Geo Linked Open Data, Metadata

KATA PENGANTAR

Puji syukur kepada Allah Yang Maha Esa atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan tugas akhir yang berjudul:

Skema Metadata untuk Dokumentasi Peninggalan Warisan Budaya yang Bertautan dengan *Geo Linked Open Data*

Melalui lembar ini, penulis hanya ingin menyampaikan ucapan terima kasih dan penghormatan yang sebesar-besarnya kepada:

1. Allah Subhaanahu Wa Ta'ala atas segala nikmat dan rahmat yang telah diberikan kepada hambaNya ini.
2. Nabi Muhammad ﷺ atas tuntunan dan bimbinganNya.
3. Ayah, Ibu dan keluarga penulis yang tiada henti-hentinya mencurahkan kasih sayang, perhatian dan doa kepada penulis selama ini.
4. Bapak Hari Ginardi selaku dosen pembimbing yang telah memberikan bimbingan, motivasi, nasihat dan meluangkan waktu untuk membantu pengerjaan Tugas Akhir ini.
5. Ibu Nurul Fajrin Ariyani selaku dosen pembimbing yang telah memberikan nasihat, arahan, dan bantuan sehingga penulis dapat menyelesaikan Tugas Akhir ini.
6. Bapak Joko selaku dosen wali penulis yang telah memberikan perhatian dan motivasi kepada penulis selama menjadi mahasiswa di lingkungan Teknik Informatika ITS.
7. Bapak dan Ibu dosen Teknik Informatika ITS yang telah membina dan memberikan ilmu kepada penulis selama menempuh studi di Teknik Informatika ITS.
8. Sahabat-sahabat penulis, Radwa, Dita Agung, Naufal, Hanif, Basofi, Yudi yang telah memberikan dukungan ,semangat serta do'a.
9. Teman seperjuangan penulis, Wimba , Kamali , Madis, Alief yang selalu bersedia diajak diskusi mengenai Tugas Akhir ini.
10. Keluarga TC 2012 yang terus memberikan dukungan dan memberikan semangat selama kuliah.

11. Seluruh teman yang telah membantu,memberikan support kepada penulis untuk menyelesaikan Tugas Akhir ini.
12. Serta pihak-pihak lain yang namanya tidak dapat penulis sebutkan satu-persatu.

Bagaimanapun juga penulis telah berusaha sebaik-baiknya dalam menyusun tugas akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Juni 2016

Mochamad Bakhtiar H

DAFTAR ISI

HALAMAN JUDUL.....	iii
LEMBAR PENGESAHAN.....	vii
ABSTRAK.....	ix
ABSTRACT.....	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR.....	xix
DAFTAR TABEL.....	xxi
DAFTAR KODE SUMBER.....	xxiii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah.....	2
1.4. Tujuan.....	2
1.5. Metodologi.....	3
1.6. Sistematika Penulisan.....	3
BAB II DASAR TEORI.....	5
2.1. Peninggalan Warisan Budaya.....	5
2.2. Metadata.....	5
2.3. RDF.....	6
2.4. Ontology.....	7
2.5. Informasi Geospasial.....	7
2.6. Linked Open Data.....	7
2.7. Geo Linked Open Data.....	8
2.8. Geonames.....	8
2.9. Wikipedia.....	9
2.10. Geocoding.....	9
2.11. RAP Library.....	10
BAB III METODOLOGI PEMECAHAN MASALAH.....	13
3.1. Analisis Data.....	13
3.2. Perancangan Skema Metadata.....	14
3.2.1. Seleksi Kelas dan Properti dari Cidoc-CRM.....	15

3.2.2.	Seleksi Kelas dari Factmin Schema.....	17
3.2.3.	Seleksi Property dari WGS84 Geo Positioning...	18
3.2.4.	Pengembangan Skema Metadata.....	18
3.3.	Penautan dengan Geonames dan Wikipedia.....	19
3.4.	Menampilkan Informasi Spatial Peninggalan Warisan Budaya.....	20
BAB IV ANALISIS DAN PERANCANGAN SISTEM.....		21
4.1.	Analisis	21
4.1.1.	Cakupan Permasalahan.....	21
4.1.2.	Deskripsi Umum Sistem.....	21
4.1.3.	Spesifikasi Kebutuhan Perangkat Lunak.....	22
4.1.4.	Aktor	23
4.1.5.	Kasus Penggunaan.....	23
4.2.	Perancangan Sistem.....	30
4.2.1	Perancangan Data	30
4.2.2	Perancangan Arsitektur.....	31
4.2.3	Perancangan Proses Aplikasi.....	32
4.3	Perancangan Antarmuka.....	35
4.3.1	Antarmuka Halaman Input Data.....	35
4.3.2	Antarmuka Halaman Menampilkan Data Artefak.....	36
4.3.3	Antarmuka Halaman Menampilkan Detail Artefak.....	38
BAB V IMPLEMENTASI		39
5.1.	Lingkungan Implementasi	39
5.2.	Implementasi Data.....	39
5.3.	Implementasi Proses Aplikasi	41
5.3.1.	Implementasi Class FormArtefak.....	41
5.3.2.	Fungsi CekLokasiProvinsi.....	43
5.3.3.	Fungsi CekLokasiKota	45
5.3.4	Fungsi CekLokasiKec.....	47
5.3.5	Implementasi Class GenerateRDF.....	52

5.3.6	Implementasi Menampilkan Data Artefak	59
5.4	Implementasi Antarmuka Pengguna.....	65
5.4.1	Antarmuka Halaman Input Data.....	65
5.4.2	Antarmuka Halaman Menampilkan Data Artefak.....	67
5.4.3	Antarmuka Halaman Menampilkan Detail Artefak.....	70
BAB VI PENGUJIAN DAN EVALUASI.....		79
6.1.	Lingkungan Pengujian.....	79
6.2.	Data Uji	79
6.3.	Skenario Pengujian.....	81
6.4.	Pengujian Fungsionalitas.....	81
6.4.1.	Pengujian Menambahkan Data Artefak.....	81
6.4.2.	Pengujian Melihat Data Artefak.....	83
6.4.3.	Pengujian Melihat Detail Artefak.....	84
6.5	Pengujian Performa	86
6.6	Evaluasi Pengujian	86
6.6.1	Evaluasi Pengujian Fungsionalitas.....	86
6.6.2	Evaluasi Pengujian Performa.....	87
BAB VII KESIMPULAN DAN SARAN		89
7.1	Kesimpulan.....	89
7.2	Saran.....	89
DAFTAR PUSTAKA.....		91
BIODATA PENULIS.....		93

[Halaman ini sengaja dikosongkan]

DAFTAR GAMBAR

Gambar 2.1 Output dari Fungsi RAP Library	11
Gambar 3.1 Data Artefak yang akan dipakai	14
Gambar 4.1 Diagram Kasus Penggunaan Sistem.....	23
Gambar 4.2 Diagram Aktivitas Menambahkan Data Artefak.....	26
Gambar 4.3 Diagram Aktivitas Melihat Data Artefak	28
Gambar 4.4 Diagram Aktivitas Melihat Detail Artefak	29
Gambar 4.5 Rancangan Arsitektur Sistem Artefak.....	31
Gambar 4.6 Flowchart Aplikasi	33
Gambar 4.7 Rancangan Antarmuka Halaman Input Data	36
Gambar 4.8 Rancangan Antarmuka Halaman Menampilkan Data Artefak.....	37
Gambar 4.9 Rancangan Antarmuka Halaman Menampilkan Data Artefak Berdasarkan Kabupaten/Kota.....	37
Gambar 4.10 Rancangan Antarmuka Halaman Menampilkan Detail Artefak.....	38
Gambar 5.1 Contoh Penggunaan Input Data Artefak.....	40
Gambar 5.2 Implementasi Antarmuka Halaman Penambahan Data	66
Gambar 5.3 Halaman Awal Daftar Data Artefak.....	67
Gambar 5.4 Halaman Daftar Data Artefak Berdasarkan Kabupaten/Kota.....	69
Gambar 5.5 Halaman Detail Informasi Artefak.....	70
Gambar 6.1 Data Uji	80
Gambar 6.2 Hasil Keluaran Pengujian Menambahkan Data Artefak.....	82
Gambar 6.3 Hasil Keluaran Pengujian Melihat Data Artefak.....	84
Gambar 6.4 Hasil Keluaran Pengujian Menambahkan Data Artefak	86

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 3.1 Seleksi Class dari Cidoc-CRM.....	16
Tabel 3.2 Seleksi Object Properties dari Cidoc-CRM.....	16
Tabel 3.3. Seleksi Class dari Factmint Schema.....	18
Tabel 3.4. Data Properties dari WGS84 Geo Positioning	18
Tabel 3.5. Kelas baru.....	19
Tabel 4.1 Daftar Kebutuhan Fungsional Perangkat Lunak.....	22
Tabel 4.2 Daftar Kode Diagram Kasus Penggunaan.....	24
Tabel 4.3 Spesifikasi Kasus Penggunaan Menambahkan Data Artefak.....	24
Tabel 4.4 Spesifikasi Kasus Penggunaan Melihat Data Artefak.	26
Tabel 4.5 Spesifikasi Kasus Penggunaan Melihat Detail Artefak	28
Tabel 5.1 Penjelasan Atribut dari Kolom Input Data Artefak....	40
Tabel 6.1 Penjelasan Kolom pada Data Uji.....	80
Tabel 6.2 Pengujian Fitur Menambahkan Data Artefak.....	81
Tabel 6.3 Pengujian Fitur Melihat Detail Artefak.....	84

[Halaman ini sengaja dikosongkan]

DAFTAR KODE SUMBER

Kode Sumber 1. 1 Fungsi dalam RDF Library.....	11
Kode Sumber 5.1 Fungsi jButton1ActionPerformed	43
Kode Sumber 5.2 Fungsi cekLokasiProvinsi pada class Geonames	45
Kode Sumber 5.3 Fungsi cekLokasiKota pada class Geonames.	47
Kode Sumber 5.4 Fungsi cekLokasiKec pada class Geonames.	50
Kode Sumber 5.5 Fungsi cekGeo pada Class Geocoding.....	52
Kode Sumber 5.6 Fungsi GenerateRDF.....	59
Kode Sumber 5.7 Fungsi mendapat seluruh object property	60
Kode Sumber 5.8 Kode Program untuk menampilkan informasi dari file rdf artefak.....	61
Kode Sumber 5.9 Kode Program untuk menampilkan informasi lokasi dengan tautannya dari file rdf artefak	62
Kode Sumber 5.10 Kode Program untuk menampilkan data artefak berdasarkan kabupaten/kota.....	65
Kode Sumber 5.11 Kode Program untuk Menampilkan Antarmuka Halaman Input Data.....	66
Kode Sumber 5.12 Kode Program untuk Menampilkan Antarmuka Halaman Awal Daftar Data Artefak.....	69
Kode Sumber 5.13 Kode Program untuk Menampilkan Antarmuka Halaman Daftar Data Artefak Berdasarkan Kabupaten/Kota.....	69
Kode Sumber 5.14 Kode Program untuk Menampilkan Antarmuka Detail Data Artefak	77

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

Pada bab ini akan dijelaskan hal-hal yang menjadi latar belakang, permasalahan yang dihadapi, batasan masalah, tujuan, metodologi dan sistematika penulisan yang digunakan dalam pembuatan buku tugas akhir ini.

1.1. Latar Belakang

Peninggalan Warisan budaya adalah peninggalan yang berupa benda maupun non benda yang merupakan jati diri suatu masyarakat yang diwariskan dari generasi sebelumnya dan yang dilestarikan untuk generasi yang akan datang. Warisan budaya dapat berupa benda seperti candi, Museum, artefak, fosil, prasasti, atau non benda seperti tradisi, ritual.

Salah satu usaha untuk melestarikan peninggalan warisan budaya adalah dengan membuat dokumentasinya.. Banyak data peninggalan warisan budaya yang belum lengkap termasuk lokasi ditemukannya, sehingga data tersebut tidak bisa memberikan informasi lengkap tentang lokasi penemuannya.

Skema metadata akan dibuat untuk membantu dalam dokumentasi peninggalan warisan budaya. Setelah pembuatan skema meta data, maka dihubungkan dengan Geo Linked Open Data yang terdiri dari CIDOC-CRM, Geonames, Wikipedia. Dengan adanya proses tersebut dalam pembuatan dokumentasi peninggalan warisan budaya maka akan dapat memberikan informasi maupun pengetahuan tentang lokasi secara lengkap ditemukannya peninggalan warisan budaya.

Perkembangan teknologi informasi membuat pengguna lebih mudah mengakses secara mobile. Implementasi sistem dalam tugas akhir ini akan menggunakan Web Application untuk pengguna. Selain itu, Web Application juga menjadi sebuah tren aplikasi pada beberapa tahun terakhir karena aksesnya yang mudah dijangkau oleh sebagian besar perangkat seperti komputer, laptop, maupun

smartphone. Oleh karena itu diperlukan sebuah sistem untuk menampilkan data peninggalan warisan budaya.

1.2. Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana cara menentukan *class dan property* yang digunakan untuk pembuatan skema metadata Peninggalan Warisan Budaya?
2. Bagaimana menautkan informasi spasial peninggalan warisan budaya dengan skema metadata, geonames, dan wikipedia?
3. Bagaimana cara menampilkan informasi spasial peninggalan warisan budaya?

1.3. Batasan Masalah

Beberapa batasan masalah yang menjadi batas pada tugas akhir ini adalah:

1. Sistem ini dioperasikan menggunakan basis desktop dan web.
2. Sistem ini menggunakan Bahasa pemrograman Java, PHP dan sistem basis data MySQL.
3. Data spasial dari geonames yang digunakan hanya data Indonesia
4. Dataset yang digunakan adalah peninggalan warisan budaya dari artefak Museum Indonesia.

1.4. Tujuan

Tujuan dari pembuatan Tugas Akhir ini adalah membuat suatu sistem yang digunakan untuk dokumentasi peninggalan warisan budaya yang bertautan dengan *Geo Linked Open Data*.

1.5. Metodologi

Ada beberapa tahap dalam proses pengerjaan tugas akhir ini, yaitu sebagai berikut:

1. Studi Literatur

Pada studi literatur ini, akan dipelajari sejumlah referensi yang diperlukan dalam pembuatan aplikasi yaitu mengenai Geonames API, Google Maps API, dan Penggunaan Geo Linked Open Data

2. Implementasi

Implementasi algoritma adalah tahapan untuk menautkan data geospasial dari artefak dengan Wikipedia dan Geonames . Implementasi algoritma ini dibangun menggunakan Bahasa pemrograman Java.

3. Uji coba dan evaluasi

Pada tahap ini, akan dilakukan uji coba menautkan data geospasial artefak dengan Wikipedia dan Geonames berdasarkan data-data artefak yang dimiliki oleh Museum Indonesia.

4. Penyusunan buku tugas akhir

Tahap ini merupakan tahap penyusunan laporan berupa buku tugas akhir sebagai dokumentasi pelaksanaan tugas akhir, yang mencakup seluruh teori, implementasi, serta hasil pengujian yang telah dikerjakan.

1.6. Sistematika Penulisan

Sistematika penulisan dibuat bertujuan untuk mendapatkan gambaran umum dari pengerjaan Tugas Akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku Tugas Akhir terdiri atas beberapa bagian seperti berikut ini.

Bab I Pendahuluan

Bab ini berisi latar belakang masalah, tujuan dan manfaat pembuatan Tugas Akhir, permasalahan, batasan masalah, metodologi yang digunakan dan sistematika penyusunan Tugas Akhir.

Bab II Dasar Teori

Bab ini membahas beberapa teori penunjang yang berhubungan dengan pokok pembahasan dan mendasari pembuatan Tugas Akhir ini.

Bab III Metode Pemecahan Masalah

Bab ini membahas mengenai metode yang digunakan untuk memecahkan masalah yang dipaparkan pada rumusan permasalahan.

Bab IV Analisis dan Perancangan Sisten

Bab ini membahas mengenai perancangan perangkat lunak. Perancangan perangkat lunak meliputi perancangan data, arsitektur, proses dan perancangan antarmuka pada perangkat lunak.

Bab V Implementasi

Bab ini berisi implementasi dari perancangan perangkat lunak dan implementasi fitur-fitur penunjang.

Bab VI Pengujian dan Evaluasi

Bab ini membahas pengujian dengan metode pengujian subjektif untuk mengetahui penilaian aspek kegunaan (*usability*) dari perangkat lunak dan pengujian fungsionalitas yang dibuat dengan memperhatikan keluaran yang dihasilkan serta evaluasi terhadap fitur-fitur perangkat lunak.

Bab VII Kesimpulan

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan. Bab ini membahas saran-saran untuk pengembangan sistem lebih lanjut.

Daftar Pustaka

Merupakan daftar referensi yang digunakan untuk mengembangkan Tugas Akhir.

BAB II DASAR TEORI

Bab ini akan membahas mengenai dasar teori dan literatur yang menjadi dasar pengerjaan tugas akhir ini.

2.1. Peninggalan Warisan Budaya

Cultural heritage dalam bahasa Inggris memiliki arti peninggalan warisan budaya. Peninggalan warisan budaya merupakan bagian dari kebudayaan yang diwariskan oleh generasi sebelumnya dan dilestarikan oleh generasi selanjutnya. Peninggalan warisan budaya adalah aset yang berharga bagi suatu negara, karena merupakan jati diri masyarakat dari suatu negara.

Peninggalan warisan budaya memiliki 4 macam bentuk yaitu benda fisik, pola perilaku, pandangan hidup, dan lingkungan. Seringkali yang kita tahu sebagai peninggalan warisan budaya adalah yang berupa benda fisik. Benda fisik yang dimaksud dalam peninggalan warisan budaya adalah benda yang dibuat oleh manusia. Contoh beberapa peninggalan warisan budaya yang merupakan benda fisik adalah artefak genta pendeta, Candi Borobudur, Keris.

Kita sebagai generasi penerus dari generasi sebelumnya wajib untuk memiliki ide, gagasan, maupun tindakan yang mampu menjaga peninggalan warisan budaya tetap ada hingga generasi penerus kita. Banyak cara dalam melakukan pelestarian peninggalan warisan budaya. Cara untuk melestarikan peninggalan warisan budaya yang digunakan dalam tugas akhir ini adalah dokumentasi dengan menggunakan skema metadata [1].

2.2. Metadata

Metada dapat diartikan sebagai data yang menjelaskan tentang suatu data. Data dalam metadata harus terstruktur. Metadata bermanfaat untuk memudahkan dalam bertukar data maupun informasi antar system yang berbeda. Dalam metadata berisi tanda kode(*tag*) yang digunakan dalam penyimpanan suatu

data. Metadata memiliki 3 jenis, yaitu metadata deskriptif, metadata administratif, metadata struktural. Metadata yang digunakan dalam tugas akhir ini adalah metadata structural. Metadata structural adalah data yang dapat terhubung dengan data lain yang saling terkait. [2].

2.3. RDF

RDF adalah spesifikasi kerangka kerja yang dikeluarkan oleh World Wide Web Consortium (W3C) sebagai model metadata untuk mempresentasikan resource yang terdapat di web. Ide dasar dari RDF adalah mendeskripsikan *resource* yang ada dengan subyek, predikat, obyek. RDF memiliki tiga bagian, yaitu resources, properties, dan statement. Resources adalah obyek yang ingin direpresentasikan dalam Semantic Web. Setiap Resources memiliki URL sebagai penanda yang unik. Properties juga merupakan *resource* yang digunakan dalam mendeskripsikan hubungan antar *resource*. Statement adalah representasi property dari sebuah resources yang dinyatakan dalam bentuk triple object-attribute-value. RDF memiliki beberapa format dalam merepresentasikan knowledge antara lain : Notation 3(N3), Turtle, N-Triples, graph.

Skema RDF dapat mendeskripsikan sebuah hubungan antar *resource*. Setiap skema metadata memiliki URI yang digunakan untuk mendefinisikan *class* dan *property*. Dengan adanya URI, dapat memudahkan dalam bertukar data maupun informasi meskipun berbeda system.

Berikut ini adalah contoh dari RDF :

```
<?xml version="1.0"?>
<RDF>
<Description about="http://www.han.com/artifact">
<physical_object>GentaPendeta</physical_object>
<condition_state>Baik dan utuh</condition_state>
</Description>
</RDF> [3].
```

2.4. Ontology

Definisi Ontology merupakan suatu teori yang menjelaskan makna dari suatu objek, serta relasi antar object tersebut. Ontology juga menjelaskan tentang sebuah konsep dalam sebuah domain tertentu (Classes), object properties, dan data properties. Secara umum, ontology digunakan pada Artificial Intelligence (AI) dan rekayasa pengetahuan. Metode ontologi dapat diterapkan dalam berbagai bidang agar dapat terhubung dan saling bertukar informasi dengan sistem yang berbeda [4].

2.5. Informasi Geospasial

Spasial adalah aspek keruangan yang mencakup suatu lokasi, letak dan posisinya. Seiring adanya system referensi koordinat peta secara umum, berkembang menjadi geospasial. Geospasial merupakan aspek keruangan yang menjelaskan letak, lokasi, maupun posisi suatu obyek di bumi. Geospasial saat ini telah sering dipakai dalam mengembangkan sistem khususnya sistem informasi geografis

Informasi geospasial di Indonesia telah diatur dalam UU, yaitu UU Nomor 4 Tahun 2011 tentang informasi geospasial Semua yang berkaitan dengan geospasial harus berdasarkan UU tersebut. Menurut UU tersebut, data geospasial merupakan data yang berkaitan dengan lokasi geografis, dimensi, atau karakteristik obyek yang ada di bumi. Sedangkan informasi geospasial merupakan hasil pengolahan dari data geospasial yang dapat digunakan dalam mengambil keputusan maupun kebijakan yang berhubungan dengan ruang kebumian [5].

2.6. Linked Open Data

Linked Open Data memiliki arti data yang terbuka dan tersambung. Makna dari data tersambung adalah datanya dibuat dengan memakai ulang data dari milik intitusi lain. Kita dapat menghubungkan data kita dengan data orang lain dengan

menggunakan teknik *linked open data*, salah satunya owl:sameAs. Data terbuka memiliki makna data yang dibuat dengan lisensi terbuka, sehingga setiap orang diperbolehkan untuk menggunakan maupun memanfaatkan data dari institusi lain.

Menurut Tim Berners-Lee *linked data* memiliki 4 komponen, yaitu : URI, HTTP, Data terstruktur, linked data platform. Dari *linked data* menjadi *linked open data* maka ditambahkan komponen kelima yaitu konten terbuka [6].

2.7. Geo Linked Open Data

Geo linked open data adalah data geografis yang terbuka dan saling terhubung. Geo linked open data juga merupakan upaya untuk menambah dimensi spasial ke Semantic Web. Geo Linked Open Data menggunakan informasi yang dikumpulkan oleh proyek OpenStreetMap dan membuatnya sebagai acuan dasar dalam menghasilkan pengetahuan RDF menurut teknik Linked data. OpenStreetMap merupakan kumpulan data spasial yang dapat menciptakan pengetahuan spasial. Data tersebut juga sudah sesuai dengan prinsip *Linked Data*, yaitu data yang terhubung sehingga memudahkan dalam bertukar data maupun informasi

Dalam rangka untuk menggunakan web sebagai media untuk integrasi data dan juga informasi, maka dataset dan kosakata komprehensi diperlukan karena memungkinkan disambiguasi dan penyelarasan data dan informasi lainnya. Banyak integrasi informasi kehidupan nyata dan agregasi tugas tidak mungkin tanpa latar belakang pengetahuan yang komprehensif [7].

2.8. Geonames

Geonames memiliki database yang dapat didownload secara gratis. Geonames memuat lebih dari 10 juta nama geografis dari seluruh dunia dan terdiri dari 9 juta unik dan memiliki 5,5 juta nama alternatif. Data dari geonames dapat kita pakai dengan memanfaatkan database dari geonames maupun web service yang disediakan oleh geonames.

Geonames telah melayani lebih dari 150 juta permintaan *web service* per hari.

Geonames mengintegrasikan data geografi seperti nama-nama tempat di berbagai bahasa, elevasi, penduduk dan lain-lain dari berbagai sumber. Semua koordinat lat / panjang mengacu pada WGS84 (World Geodetic System 1984) [8].

2.9. Wikipedia

Wikipedia merupakan situs yang berisi tentang berbagai artikel ensiklopedia dan juga merupakan salah satu referensi online yang sering digunakan oleh orang-orang di seluruh dunia. Ketika anda menuliskan suatu keyword melalui situs google, maka artikel yang ada dalam Wikipedia seringkali muncul pada halaman pertama hasil pencarian. Wikipedia saat ini merupakan situs yang paling banyak memiliki konten artikel. Wikipedia juga bersifat *open*, sehingga setiap orang dapat membuat artikel dalam situs tersebut.

Wikipedia adalah sebuah situs yang didirikan oleh Larry Sanger dan Jim Wales pada bulan Januari 2001. Wikipedia dikelola oleh Wikimedia Foundation sebuah perusahaan nirlaba yang bermarkas di Amerika Serikat. Yang paling istimewa dari Wikipedia adalah sebuah situs besar yang bebas iklan sehingga orang-orang yang mencari artikel merasa nyaman jika mengunjungi situs wikipedia [9].

2.10. Geocoding

Geocoding adalah proses mengkonversi alamat suatu tempat menjadi koordinat geografis yang dapat digunakan untuk menempatkan sebuah *marker* dalam *google maps*. Reverse Geocoding adalah proses mengkonversi koordinat geografis

menjadi alamat suatu tempat. Dalam google maps terdapat Geocoding API yang disediakan untuk menemukan alamat suatu tempat dengan memasukkan koordinat geografis [10].

2.11. RAP Library

RAP adalah RDF API yang digunakan untuk Bahasa pemrograman PHP. RAP merupakan library yang digunakan untuk *parsing, query, manipulasi* RDF. Library ini memiliki fitur antara lain : *statement-centric methods, resource-centric methods, ontology-centric methods, quad and named graph-centric methods*, terintegrasi dengan RDF/XML, N3, N-TRIPLE, TriX, GRDDL, RSS parser. *Statement-centric methods* yang digunakan untuk manipulasi sebuah model RDF sebagai satu set triple RDF. *Resource-centric methods* yang digunakan untuk manipulasi sebuah model RDF sebagai satu set sumber daya. *Ontology-centric methods* untuk manipulasi model RDF melalui *vocabulary specific method*. *Quad and named graph-centric methods* yang digunakan untuk manipulasi dataset RDF [11].

Berikut ini contoh penggunaan dari RAP Library.

```
// Include RAP
define("RDFAPI_INCLUDE_DIR", "../api/");
include(RDFAPI_INCLUDE_DIR . "RDFAPI.php");

// Filename of an RDF document
$base="example1.rdf";

// Create a new MemModel
$model = ModelFactory::getDefaultModel();

// Load and parse document
$model->load($base);
```

```
// Output model as HTML table
$model->writeAsHtmlTable();
```

Kode Sumber 1. 2 Fungsi dalam RDF Library

Dan untuk output dari kode sumber di atas adalah pada Gambar 2.1 berikut :

Output:

No.	Subject	Predicate	Object
1.	Resource: http://www.w3.org/Home/Lassila	Resource: http://description.org/schema/Creator	Blank Node: b85740
2.	Blank Node: b85740	RDF Node: http://www.w3.org/1999/02/22-rdf-syntax-ns#type	Resource: http://description.org/schema/Person
3.	Blank Node: b85740	Resource: http://example.org/stuff/1.0/Name	Literal: Ora Lassila (rdf:datatype="http://www.w3.org/TR/xmlschema-2#string")
4.	Blank Node: b85740	Resource: http://example.org/stuff/1.0/Email	Literal: lassila@w3.org

Gambar 2.1 Output dari Fungsi RAP Library

[Halaman ini sengaja dikosongkan]

BAB III

METODOLOGI PEMECAHAN MASALAH

Pada bab ini dijelaskan mengenai langkah-langkah yang dilakukan untuk menentukan pembuatan skema metadata, menautkan informasi spasial peninggalan warisan budaya dengan geonames dan Wikipedia, dan kemudian menampilkan informasi spasial peninggalan warisan budaya. Mulai dari metode yang dilakukan untuk menggabungkan beberapa *linked open data* hingga pengembangan menjadi satu skema metadata yang telah digabungkan. Penggabungan dan pengembangan tersebut bertujuan untuk menghasilkan suatu skema metadata yang akan digunakan dalam dokumentasi peninggalan warisan budaya. Proses selanjutnya yaitu menautkan informasi spasial peninggalan warisan budaya dengan Geonames dan Wikipedia. Setelah itu informasi spasial peninggalan warisan budaya dapat diperoleh.

3.1. Analisis Data

Untuk memecahkan masalah dalam menentukan *class* dan *property* yang digunakan dalam pembuatan skema metadata, langkah yang pertama kali dilakukan adalah menganalisis dataset yang akan digunakan. Dari dataset yang ada dipilih atribut yang sesuai dengan tugas akhir ini.

Data yang digunakan dalam perancangan skema metadata ini adalah data artefak dari Museum Indonesia. Objek utama untuk dijadikan data adalah artefak dan lokasi penemuan artefak. Data dari Museum Indonesia berupa file excel. Dalam file tersebut ada beberapa atribut antara lain : No.Urut, No.InvB, No_FotoGab, Nama_Benda, Tempat_Asal_Des_Kec, Des_Gabungan1, Kondisi,Lokasi_Gedung1, Lokasi Ruang1, Lokasi Lemari_Laci1, Lokasi_Lain1, Tempat_Asal_Kab, Tempat_Asal_Prop, Tempat Pembuatan, Tempat Perolehan, Tahun Pembuatan. Dari beberapa atribut tersebut akan dipilah untuk menentukan yang akan digunakan dalam Tugas Akhir ini. Berikut beberapa atribut yang akan digunakan, antara lain : No.Urut, Nama Benda,

Tempat_Asal_Kec, Tempat_Asal_Kab, Tempat_Asal_Prop, Tahun_Pembuatan, Bahan, Ukuran(panjang, lebar, tinggi, tebal, diameter), Kondisi. No.Urut merupakan sebagai id setiap artefak yang unik. Nama Benda digunakan untuk mendeskripsikan nama artefak. Untuk lokasi penemuan artefak dimasukkan dalam kolom Tempat_Asal_Kec, Tempat_Asal_Kab, Tempat_Asal_Prop. Dalam mendeskripsikan ukuran benda(artefak) ada dalam beberapa kolom, yaitu panjang, lebar, tinggi, tebal, dan diameter. Kemudian untuk mendeskripsikan tahun pembuatan artefak ada pada kolom Tahun Pembuatan. Kolom Bahan dan Kondisi masing-masing digunakan untuk mendeskripsikan Bahan Pembuatan dari artefak dan kondisi artefak pada saat ditemukan. Sebagai Contoh,Gambar 3.1 berikut adalah hasil akhir Data yang akan dipakai.

Tempat_Asal_Kec	Tempat_Asal_Kab	Tempat_Asal_Prop	Tahun_Pembuatan	Bahan	Panjang	Lebar	Tinggi	Tebal	Diameter	No.Urut
...
...

Gambar 3.1 Data Artefak yang akan dipakai

3.2. Perancangan Skema Metadata

Pada tahap ini, skema metadata dibuat dengan menggabungkan beberapa *linked open data* yang sudah ada. *Linked Open Data* yang digunakan adalah Cidoc-CRM Ontologi, factmint schema,dan WGS84 Geo Positioning. Cidoc-CRM *Ontology* adalah ontologi yang mendeskripsikan peninggalan warisan budaya. Factmint schema adalah skema dari lokasi geografis. Sedangkan WGS84 Geo Positioning adalah skema untuk koordinat lokasi. Setelah menggabungkannya, skema metadata dikembangkan agar dapat menyimpan informasi yang telah didapat. Cidoc-CRM, factmint schema, dan WGS84 Geo Positioning memiliki komponen *class* dan *property* yang sesuai

untuk menyelesaikan permasalahan dari topik tugas akhir ini. Berdasarkan ketiga ontologi yang akan digunakan, *Cidoc-CRM* digunakan sebagai landasan pembangunan skema metadata yang dikerjakan pada tugas akhir ini. Factmint schema memiliki komponen paling sesuai untuk lokasi penemuan artefak. Sedangkan WGS84 Geo Positioning sesuai dengan koordinat lokasi penemuan artefak. [5]

3.2.1. Seleksi Kelas dan Properti dari Cidoc-CRM

Cidoc-CRM merupakan Ontologi standard yang berkaitan dengan museum termasuk artefak. Ada banyak *class* dan *property* yang ada pada ontology ini, namun hanya beberapa yang akan digunakan dalam Tugas Akhir Ini. Pertama *class* E19.Physical-Object dalam tugas akhir ini digunakan oleh suatu benda artefak, misalnya Tangkai Genta Pendeta maka menggunakan *class* E19.Physical-Object. Kemudian untuk mendefinisikan waktu pembuatan artefak maka menggunakan *class* E52_Time-Span. E57_Material digunakan untuk mendefinisikan bahan pembuatan suatu artefak. Selain itu juga dalam tugas akhir ini memakai *class* E54_Dimension yang digunakan untuk mendefinisikan ukuran suatu artefak yang meliputi panjang, lebar, tinggi, tebal, diameter dan E3_Condition_State digunakan untuk mendefinisikan kondisi/keadaan artefak pada saat ditemukan. Untuk lebih jelasnya *Classes* beserta fungsi secara umum yang digunakan terurai pada Tabel 3.1. Skema metadata yang akan dibuat juga akan memakai *object property* yang telah ada pada Ontologi *Cidoc-CRM*. Ada 5 *object property* yang digunakan. Pertama, P53_has_former_or_current_location digunakan untuk menghubungkan antara artefak dengan tempat penemuan artefak. Kedua, P43_has_dimension digunakan untuk menghubungkan antara artefak dengan ukuran artefak. Ketiga, P4_has_time_span digunakan untuk menghubungkan antara artefak dengan waktu pembuatan artefak. Keempat, P44_has_condition digunakan untuk menghubungkan antara artefak dengan kondisi artefak. Kelima,

P45_has_consist_of digunakan untuk menghubungkan artefak dengan material/bahan pembuatan artefak. Agar lebih jelas, Object Properties yang digunakan terurai pada Tabel 3.2

Tabel 3.1 Seleksi Class dari Cidoc-CRM

Classes		
E19.Physical-Object	URI	http://www.cidoc-crm.org/cidoc-crm#
	Fungsi	Mendekripsikan <i>instance</i> sebagai objek fisik
E52_Time-Span	URI	http://www.cidoc-crm.org/cidoc-crm#
	Fungsi	Mendekripsikan <i>instance</i> sebagai waktu kejadian
E57_Material	URI	http://www.cidoc-crm.org/cidoc-crm#
	Fungsi	Mendekripsikan <i>instance</i> sebagai bahan pembuatan
E54_Dimension	URI	http://www.cidoc-crm.org/cidoc-crm#
	Fungsi	Mendekripsikan <i>instance</i> sebagai dimensi/ukuran
E3_Condition_State	URI	http://www.cidoc-crm.org/cidoc-crm#
	Fungsi	Mendekripsikan <i>instance</i> sebagai kondisi

Tabel 3.2 Seleksi Object Properties dari Cidoc-CRM

Object Properties		
P53_has_former_or_current_location	URI	http://www.cidoc-crm.org/cidoc-crm#
	Fungsi	Mendeskripsikan relasi Physical Object dengan tempat ditemukannya.
P43_has_dimension	URI	http://www.cidoc-crm.org/cidoc-crm#

Object Properties		
	Fungsi	Mendeskripsikan relasi antara Physical Object dengan Dimension.
P4_has_time-span	URI	http://www.cidoc-crm.org/cidoc-crm#
	Fungsi	Mendeskripsikan relasi antara Physical Object dengan Time Span.
P44_has_condition	URI	http://www.cidoc-crm.org/cidoc-crm#
	Fungsi	Mendeskripsikan relasi antara Physical Object dengan Condition
P45_consist_of	URI	http://www.cidoc-crm.org/cidoc-crm#
	Fungsi	Mendeskripsikan relasi antara Physical Object dengan Material

3.2.2. Seleksi Kelas dari Factmin Schema

Factmin Schema adalah gabungan dari beberapa skema ontologi. Salah satu skema yang disediakan oleh Ontologi ini dan akan dipakai dalam tugas akhir ini yaitu skema untuk mendeskripsikan informasi letak geografis. Dalam Skema tersebut ada beberapa *Classes* dan 2 dari *class* yang ada cocok digunakan untuk tugas akhir ini. Pertama Province merupakan *class* yang digunakan untuk lokasi provinsi tempat ditemukannya artefak. Kemudian yang kedua *class* District digunakan untuk mendeskripsikan lokasi kabupaten/kota tempat ditemukannya

artefak. Untuk lebih jelas, *Classes* yang digunakan terurai pada tabel 3.3 .

Tabel 3.3. Seleksi Class dari Factmint Schema

Classes		
Province	URI	http://rdf.fm/geographic-location#
	Fungsi	Mendekripsikan <i>instance</i> sebagai lokasi geografis provinsi
District	URI	http://rdf.fm/geographic-location#
	Fungsi	Mendekripsikan <i>instance</i> sebagai lokasi geografis kabupaten/kota

3.2.3. Seleksi Property dari WGS84 Geo Positioning

Ontologi ini menyediakan *data properties* untuk mendeskripsikan koordinat letak geografis. Geonames juga menggunakan data properties dari WGS84 Geo Positioning yaitu latitude dan longitude. Dalam tugas akhir ini, data properties yang digunakan adalah data properties latitude dan longitude untuk mendeskripsikan koordinat lokasi ditemukannya artefak. Dari berbagai *properties* yang terdapat pada Ontologi ini, diambil *data properties* yang diuraikan pada Tabel 3.4.

Tabel 3.4. Data Properties dari WGS84 Geo Positioning

Data Properties		
latitude	URI	http://www.w3.org/2003/01/geo/wgs84_pos#
	Fungsi	Mendeskripsikan koordinat <i>latitude</i> letak geografis
longitude	URI	http://www.w3.org/2003/01/geo/wgs84_pos#
	Fungsi	Mendeskripsikan koordinat <i>longiude</i> letak geografis

3.2.4. Pengembangan Skema Metadata

Dari Ontologi yang sudah ada masih kurang 1 *classes* yang mendeskripsikan lokasi penemuan kecamatan, maka dibutuhkan kelas baru. Kelas baru ditambahkan agar menjadi satu skema

metadata yang dibutuhkan. Kelas baru yang ditambahkan terurai pada Tabel 3.5.

Tabel 3.5. Kelas baru

Classes		
Subdistrict	URI	<i>http://www.han.com/artifact#</i>
	Fungsi	Mendeskripsikan bahwa <i>instance</i> adalah lokasi penemuan kecamatan

Dari beberapa ontology yang telah dijelaskan di atas, maka dari beberapa *class*, *object property*, *data property* yang telah diseleksi digabungkan menjadi satu yang akan membentuk suatu skema metadata yang akan digunakan dalam tugas akhir ini.

3.3. Penautan dengan Geonames dan Wikipedia

Analisis data dan juga penentuan class dan property yang dipakai dalam skema metadata telah dilakukan, maka selanjutnya penjelasan proses menautkan data artefak dengan geonames dan Wikipedia sehingga didapatkan informasi spasial peninggalan warisan budaya. Data artefak yang digunakan untuk penautan dengan geonames dan Wikipedia adalah data lokasi penemuan yang meliputi kecamatan, kabupaten/kota, provinsi. Geonames menyediakan web services yang dalam tugas akhir ini dimanfaatkan untuk proses menautkan dengan Geonames dan Wikipedia.

Cara untuk menautkan dengan geonames dan wikipedia adalah input data lokasi penemuan artefak yang dimasukkan dibagi menjadi 3 wilayah administrasi, yaitu kecamatan, kabupaten/kota, provinsi. Selain itu data yang dimasukkan juga harus secara spesifik untuk lokasi kecamatan dan kabupaten/kota, misalnya Mojokerto ada Kabupaten Mojokerto dan Kota Mojokerto, maka jika ingin mencari Kabupaten Mojokerto input data harus sesuai yaitu Kabupaten Mojokerto dan jika ingin mencari yang Kota Mojokerto maka input data yang dilakukan harus sama, yaitu Kota Mojokerto. Kemudian untuk lokasi provinsi input hanya berupa nama provinsi. Misalnya, ingin mencari Jawa Timur maka input

yang dilakukan adalah Jawa Timur. Dari input yang dilakukan akan keluar beberapa data. Jika data lebih dari 1, untuk mendapatkan hasil akurat yang diambil adalah data pertama.

3.4. Menampilkan Informasi Spasial Peninggalan Warisan Budaya

Informasi spasial peninggalan warisan budaya diperoleh setelah proses penautan dengan Geonames dan Wikipedia. Informasi spasial akan ditampilkan oleh perangkat lunak berbasis web. Informasi spasial yang diperoleh meliputi informasi tentang lokasi penemuan artefak setiap wilayah administrasi yaitu kecamatan, kabupaten/kota, provinsi . Informasi yang diperoleh diantara lain : jumlah penduduk di wilayah itu, letak lokasi yang digambarkan melalui peta, luas geografis wilayah tersebut, dsb. Selain itu juga menampilkan informasi tentang sejarah daerah misalnya sejarah Kabupaten Brebes.

Dari informasi spasial yang diperoleh, dapat menambah wawasan maupun pengetahuan tentang lokasi penemuan artefak. Kita dapat memperoleh informasi detail artefak dan juga sekaligus mengetahui kondisi geografis suatu daerah, kondisi perekonomian, sektor pariwisata, daftar pemimpin suatu daerah, dan juga sarana prasarana yang ada di suatu daerah.

BAB IV

ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini dijelaskan tentang analisis permasalahan dan perancangan Tugas Akhir. Analisis permasalahan membahas tentang permasalahan yang diangkat dalam Tugas Akhir ini beserta solusi yang ditawarkan. Selanjutnya dibahas juga tentang perancangan sistem yang dibuat.

4.1. Analisis

Tahap analisis dibagi menjadi beberapa bagian antara lain cakupan permasalahan, deskripsi umum sistem, kasus penggunaan sistem dan kebutuhan perangkat lunak.

4.1.1. Cakupan Permasalahan

Indonesia adalah negara yang memiliki warisan budaya beraneka ragam. Warisan budaya merupakan peninggalan yang wajib dijaga keberadaannya. Selama ini warisan budaya berupa benda seperti artefak disimpan di dalam museum. Namun seringkali data artefak yang ada masih terdokumentasi secara manual, belum ada sistem yang mendokumentasikan data artefak.

Dengan adanya permasalahan tersebut, dalam Tugas Akhir ini dibuat sebuah sistem informasi sebagai sarana untuk dokumentasi data artefak serta untuk menarik minat orang-orang untuk mengetahui warisan budaya di Indonesia yang telah ditemukan sehingga dapat menumbuhkan cinta tanah air.

4.1.2. Deskripsi Umum Sistem

Sistem yang dibangun dalam Tugas Akhir ini terdiri dari dua bagian utama, yaitu proses menautkan data geospasial dengan Geonames dan Wikipedia yang merupakan bagian dari proses menambahkan data artefak dan Menampilkan daftar data artefak beserta detail data artefak. Data yang diperlukan adalah data artefak dengan atribut nama artefak, dimensi artefak, tempat

ditemukannya artefak(kecamatan,kabupaten.kota,provinsi), bahan pembuatan artefak, tahun pembuatan artefak, kondisi artefak.

Sistem ini dibangun menggunakan bahasa pemrograman Java yang digunakan sebagai proses menautkan data geospasial dan menambahkan data artefak. Selain itu juga dibangun system berbasis Web yang digunakan untuk menampilkan daftar data artefak dan detail data artefak dengan menggunakan bahasa *PHP Hypertext Preprocessor* (PHP) dan JavaScript. Sistem ini menggunakan library geonames dan geocoding API sebagai proses penautan data. Library lain yang digunakan ialah RAP sebagai penampil detail dari data artefak.

4.1.3. Spesifikasi Kebutuhan Perangkat Lunak

Subbab ini menjelaskan kebutuhan perangkat lunak dalam bentuk diagram kasus dan diagram aktivitas. Masing-masing diagram menjelaskan perilaku atau sifat dari sistem ini.

4.1.3.1. Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan pokok yang harus dipenuhi agar sistem dapat berjalan dengan baik. Daftar kebutuhan fungsional dapat dilihat pada Tabel 4.1.

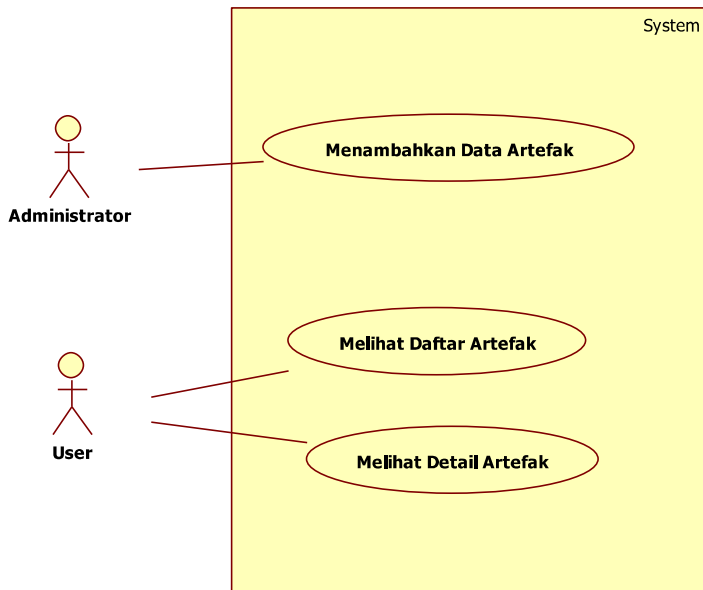
Tabel 4.1 Daftar Kebutuhan Fungsional Perangkat Lunak

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
TA-F01	Menambahkan Data Artefak	Melakukan penambahan data artefak dan membuat skema metadata berupa file rdf
TA-F02	Melihat Data Artefak	Melihat daftar data artefak yang telah ditambahkan
TA-F03	Melihat Detail Artefak	Melihat Detail Informasi artefak

4.1.4. Aktor

Aktor merupakan entitas-entitas yang terlibat dan berinteraksi langsung dengan sistem. Entitas yang dimaksud dapat berupa manusia, system, atau perangkat lunak yang lain. Aktor yang berinteraksi dengan Tugas Akhir ini yaitu administrator yang bertugas untuk menambahkan data artefak baru dan pengguna yang diasumsikan tidak memahami bahasa pemrograman. Pengguna dapat melihat data artefak dan dapat melihat detail informasi artefak dengan memilih salah satu data artefak.

4.1.5. Kasus Penggunaan



Gambar 4.1 Diagram Kasus Penggunaan Sistem

Kasus penggunaan dalam Subbab ini akan dijelaskan secara rinci. Kasus penggunaan dijabarkan dalam bentuk

spesifikasi kasus penggunaan dan diagram aktivitas. Diagram kasus penggunaan dapat dilihat pada Gambar 4.1. Daftar kode diagram kasus penggunaan sistem dapat dilihat pada Tabel 4.2.

Tabel 4.2 Daftar Kode Diagram Kasus Penggunaan

Kode Kasus Penggunaan	Nama Kasus Penggunaan
UC01	Menambah Data Artefak
UC02	Melihat Daftar Artefak
UC03	Melihat Detail Artefak

4.1.5.1. Kasus Penggunaan Menambahkan Data Artefak

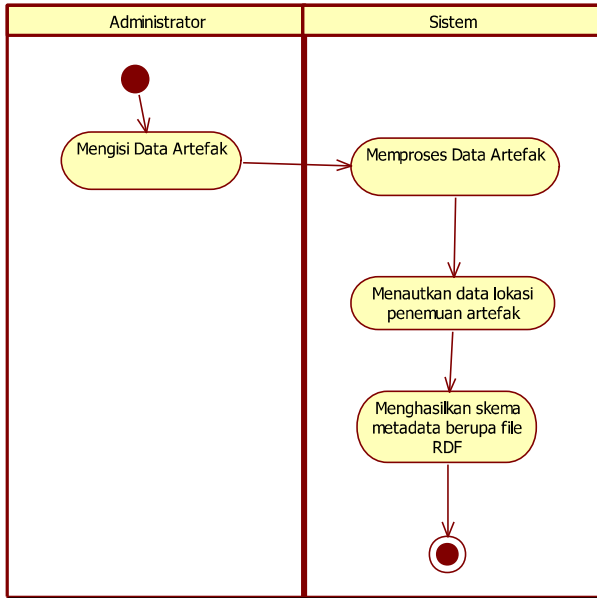
Pada kasus penggunaan menambahkan data artefak, administrator memasukkan data artefak yang berasal dari data Museum Indonesia dengan atribut berupa nama artefak, tempat penemuan(kecamatan, kota/kabupaten, provinsi), ukuran artefak, tahun penemuan, Kondisi artefak, Bahan artefak dalam sebuah form yang disediakan oleh system berbasis desktop yang nantinya akan diproses menggunakan library geonames maupun Geocoding API sehingga menghasilkan sebuah skema metadata yang berupa file rdf yang berisikan data artefak tersebut. Library Geonames dan Geocoding API digunakan untuk menautkan peninggalan warisan budaya dengan Geonames dan Wikipedia. Data yang ditautkan adalah data lokasi penemuan artefak antara lain kecamatan, kabupaten/kota, provinsi. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 4.3.

Tabel 4.3 Spesifikasi Kasus Penggunaan Menambahkan Data Artefak

Nama Kasus Penggunaan	Menambahkan data artefak
Nomor	UC01

Aktor	Administrator
Kondisi Awal	Administrator telah memiliki data artefak berupa file excel
Kondisi Akhir	Sistem menghasilkan sebuah file rdf yang berisi data artefak
Alur Normal	<ol style="list-style-type: none"> 1. Administrator mengisi form data artefak. 2. Sistem akan memproses data masukan dari administrator dan menautkan data dengan geonames dan wikipedia. 3. Sistem menghasilkan skema metadata berupa file rdf yang berisi data artefak
Alur Alternatif	-

Berdasarkan skenario kasus penggunaan pada Tabel 3.3 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan Menambah Data Artefak dapat dilihat pada Gambar 4.2.



Gambar 4.2 Diagram Aktivitas Menambahkan Data Artefak

4.1.5.2. Kasus Penggunaan Melihat Data Artefak

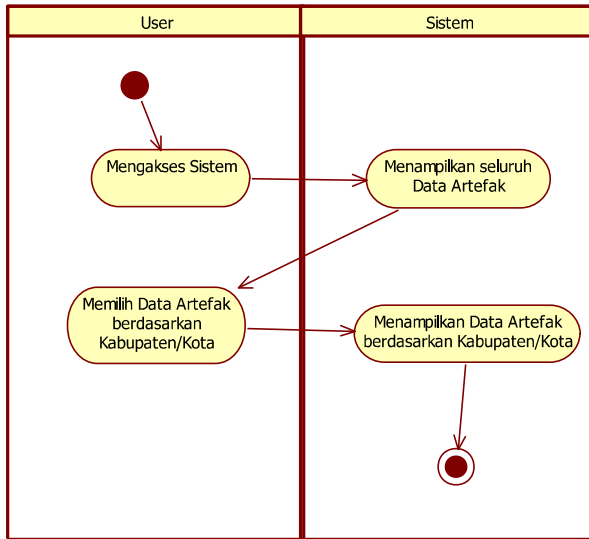
Pada kasus penggunaan melihat daftar data artefak, user dapat melihat hasil daftar data artefak. Hasil akhir dari proses ini, sistem akan menampilkan sebuah daftar data artefak. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel.

Tabel 4.4 Spesifikasi Kasus Penggunaan Melihat Data Artefak

Nama Kasus Penggunaan	Melihat data artefak
Nomor	UC02

Aktor	User
Kondisi Awal	User belum membuka <i>url</i> menuju halaman daftar data artefak
Kondisi Akhir	Sistem menampilkan daftar data artefak
Alur Normal	<ol style="list-style-type: none"> 1. User membuka <i>url</i> untuk menampilkan daftar data artefak melalui <i>Web Browser</i>. 2. Sistem akan menampilkan daftar data artefak Seluruh Indonesia. 3. User memilih data artefak berdasarkan kabupaten/kota. 4. Sistem menampilkan data artefak berdasarkan kabupaten/kota yang dipilih
Alur Alternatif	-

Berdasarkan skenario kasus penggunaan pada Tabel 3.4 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan Melihat daftar data artefak dapat dilihat pada Gambar 4.3.



Gambar 4.3 Diagram Aktivitas Melihat Data Artefak

4.1.5.3. Kasus Penggunaan Melihat Detail Artefak

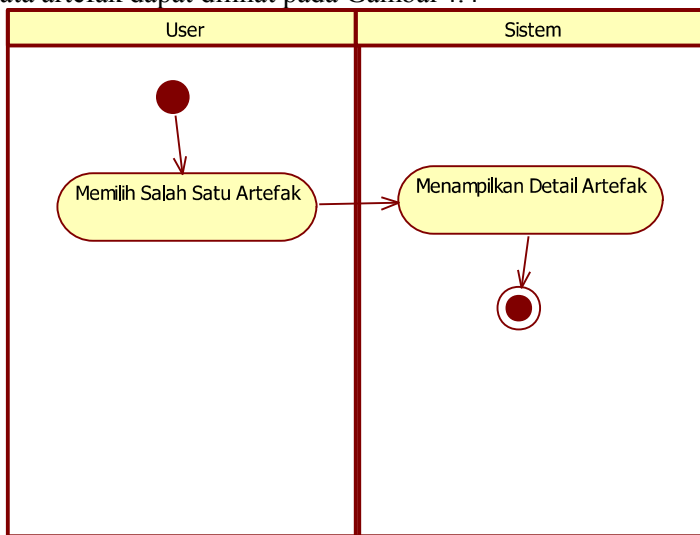
Pada kasus penggunaan melihat detail artefak, user dapat melihat detail artefak. Hasil akhir dari proses ini, sistem akan menampilkan sebuah detail informasi artefak dan informasi spatial peninggalan warisan budaya. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 4.5.

Tabel 4.5 Spesifikasi Kasus Penggunaan Melihat Detail Artefak

Nama Kasus Penggunaan	Melihat Detail artefak
Nomor	UC03
Aktor	User
Kondisi Awal	Sistem menampilkan Data Artefak

Kondisi Akhir	Sistem menampilkan detail informasi data artefak
Alur Normal	<ol style="list-style-type: none"> 1. User memilih salah satu data artefak. 2. Sistem akan menampilkan detail informasi data artefak. 3. User mendapatkan detail informasi artefak yang telah dipilih dan informasi spatial peninggalan warisan budaya.
Alur Alternatif	-

Berdasarkan skenario kasus penggunaan pada Tabel 3.4 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan Melihat daftar data artefak dapat dilihat pada Gambar 4.4



Gambar 4.4 Diagram Aktivitas Melihat Detail Artefak

4.2. Perancangan Sistem

Pada subbab ini dijelaskan mengenai tahapan perancangan sistem. Perancangan sistem ini dibagi menjadi beberapa bagian yang meliputi perancangan data, perancangan arsitektur sistem, perancangan proses aplikasi, dan perancangan antarmuka pengguna.

4.2.1 Perancangan Data

Dalam tugas akhir ini data-data yang digunakan adalah sesuai dengan analisis kebutuhan. Adapun perancangan data ini digunakan sebagai acuan dalam pemrosesan menautkan data geospasial dengan geonames memanfaatkan Library Geonames dan Google Maps Geocoding API. Oleh karena itu, kebutuhan data yang dibutuhkan oleh sistem ini adalah:

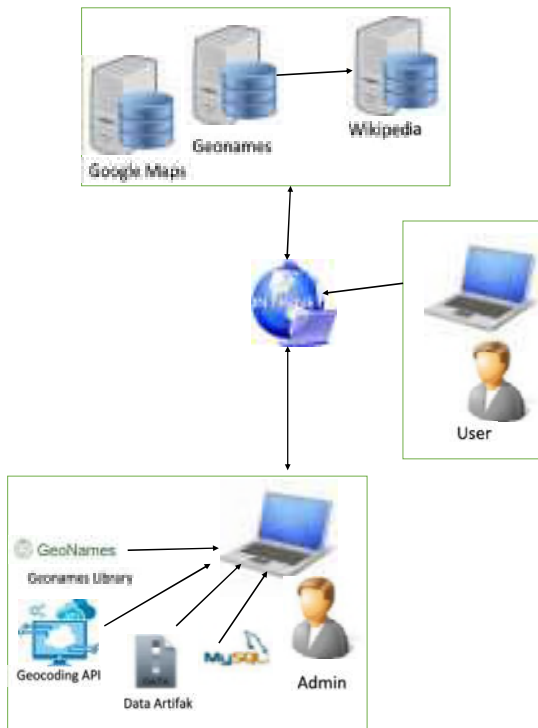
1. Data Artefak dari Museum Indonesia. Data ini nantinya akan dijadikan sebagai metadata yang terdiri dari beberapa *linked open data*.
2. Data spasial Negara Indonesia yang terdiri dari Provinsi, Kabupaten/Kota, Kecamatan. Data spasial ini dibutuhkan dalam proses menautkan dari data artefak yang diinputkan. Data spasial yang dibutuhkan berupa xml yang cara mendapatkan data tersebut memanfaatkan library geonames.
3. Data artefak lengkap dari Museum Indonesia yang berisi nama artefak, lokasi ditemukannya artefak, kondisi artefak pada saat ditemukan, tahun ditemukannya artefak, bahan pembuatan artefak, dan ukuran artefak. Data tersebut dipilah terlebih dahulu untuk mendapatkan data yang secara lengkap lokasi ditemukannya artefak.

Dalam tugas akhir ini menggunakan basis data MySQL. Basis data digunakan untuk menyimpan data artefak yang berisi nama artefak dan juga lokasi tempat menyimpan file rdf yang berisi skema metadata artefak. Dalam basis data yang digunakan hanya

memiliki 1 tabel yang bernama tabel artefak dengan atribut `id_artefak`, `nama_artefak`, `link_rdf`.

4.2.2 Perancangan Arsitektur

Dalam tugas akhir ini akan dirancang dan diaplikasikan sebuah aplikasi dekstop untuk input data artefak dan menautkan data geospasial serta sistem informasi berbasis web menggunakan PHP dan MySQL. Secara garis besar aplikasi ini memiliki rancangan arsitektur sistem yang dapat dilihat pada Gambar 4.5.

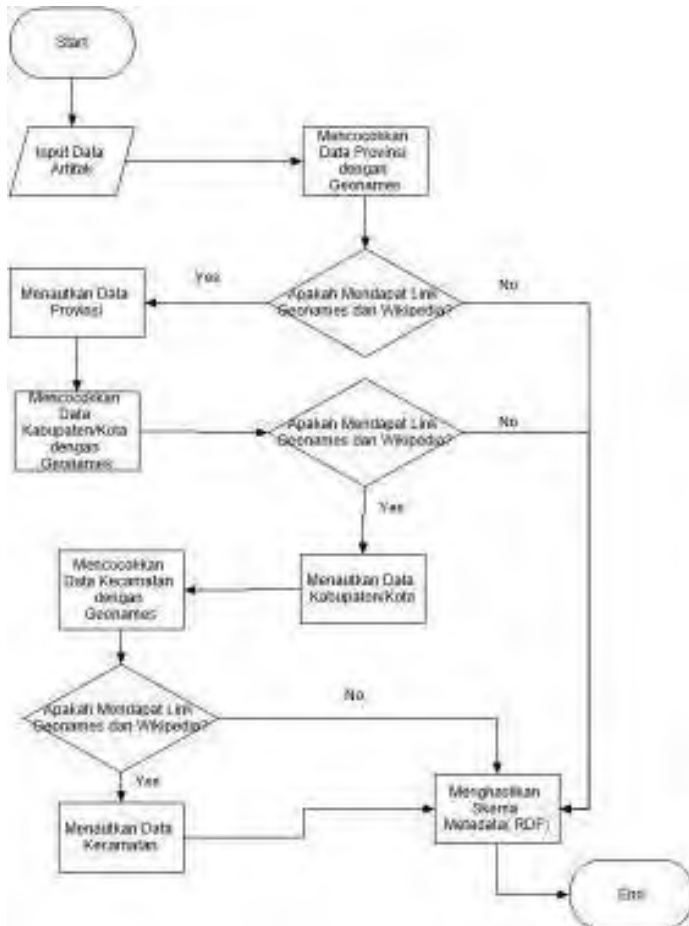


Gambar 4.5 Rancangan Arsitektur Sistem Artefak

Dari gambar 4.5, Administrator memasukkan data artefak dengan atribut Nama, Provinsi, Kabupaten/Kota, Kecamatan, Tahun, Bahan, Ukuran, Kondisi . Data artefak ini kemudian akan dikelola oleh sistem sesuai masukan oleh administrator. Sistem akan membutuhkan koneksi internet untuk menggunakan library Geonames yang digunakan untuk menautkan lokasi ditemukannya artefak. Sistem juga akan membutuhkan layanan dari Google Maps Geocoding Application Programming Interface (API) untuk mendapatkan lokasi kecamatan artefak dengan tepat. Setelah proses memasukkan data artefak berhasil, maka data artefak yang telah ditambahkan dapat dilihat oleh user.

4.2.3 Perancangan Proses Aplikasi

Proses-proses utama yang terdapat Sistem Daftar artefak terdiri dari beberapa proses yang meliputi input data artefak, menautkan data geospasial, generate menjadi file rdf, menampilkan data artefak. Proses aplikasi akan digambarkan dalam bentuk flowchart pada gambar 4.6 berikut :



Gambar 4.6 Flowchart Aplikasi

Berikut ini merupakan penjelasan dari proses-proses utama yang ada pada sistem.

1. Input Data Artefak

Proses pertama yang diperlukan untuk melakukan penautan data geospasial adalah input data artefak oleh Administrator. Data artefak yang perlu diproses adalah data artefak yang memiliki beberapa atribut sesuai pada tahapan perancangan

data. Atribut yang diperlukan untuk dilakukan proses penautan data geospasial dan generate menjadi file RDF diantaranya Nama,Provinsi,Kabupaten/Kota,Kecamatan,Ukuran,Tahun,Bahan ,Kondisi.

Data artefak yang diinputkan dalam proses ini adalah data artefak yang telah siap untuk dilakukan penautan data geospasial. Data yang telah siap tadi perlu dimasukkan ke dalam Sistem. Setelah data diinputkan oleh pengguna melalui Aplikasi Desktop.

2. Penautan Data Geospasial

Data yang telah dimasukkan oleh administrator digunakan untuk menautkan data geospasial di dalam aplikasi. Data geospasial yang dimaksud adalah data lokasi ditemukannya artefak seperti provinsi,kabupaten/kota,kecamatan. Dari data geospasial tersebut ditautkan dengan geonames dan Wikipedia dengan memanfaatkan library Geonames dan Geocoding API. Dari beberapa atribut setelah proses penautan akan menghasilkan tautan yang menghubungkan dengan geonames dan Wikipedia.

3. Generate File RDF

Sebelum diproses dalam aplikasi,dibuat terlebih dahulu skema metadata sesuai atribut dari data artefak yang dibutuhkan. Setelah itu aplikasi melalui kelas GenerateRDF akan melakukan generate berdasarkan data yang diinputkan dan data geospasial yang berhasil ditautkan. Dari hasil generate tersebut dihasilkan sebuah metadata dari data artefak yang telah diinputkan dan telah ditautkan dengan Geonames dan Wikipedia.

4. Menampilkan Data Artefak Hasil Menautkan

Setelah semua data yang dibutuhkan tersedia, sistem kini dapat menampilkan sebuah daftar data artefak. Pada proses ini pengguna dapat melihat daftar artefak dalam satu tabel yang ketika di klik setiap artefak akan menampilkan informasi tentang artefak

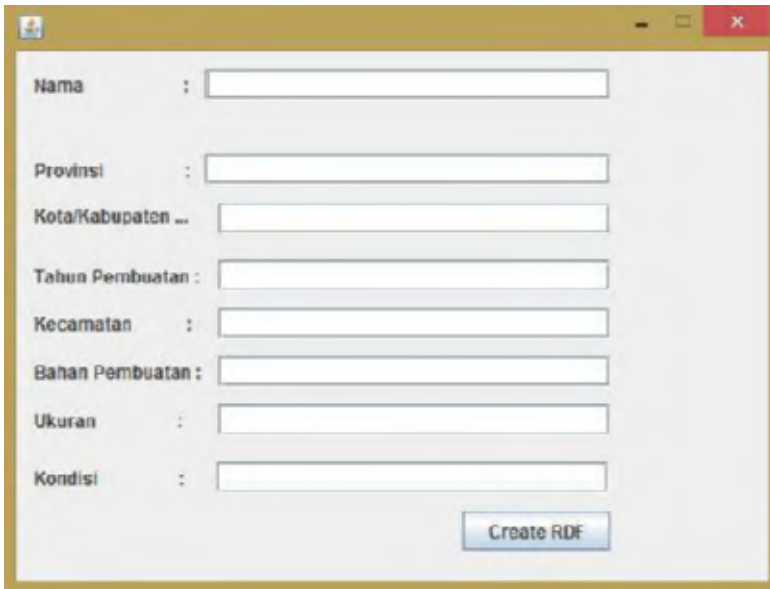
tersebut. Selain itu juga menampilkan informasi tempat ditemukannya artefak tersebut disertai link peta yang menuju geonames dan link artikel menuju Wikipedia.

4.3 Perancangan Antarmuka

Pada bagian ini dijelaskan mengenai rancangan tampilan antarmuka pengguna dari sistem. Antarmuka pengguna dalam sistem ini terdiri dari dua bagian yaitu halaman input data artefak dan menampilkan hasil menautkan data geospasial artefak. Berikut akan dijelaskan masing-masing rancangan antarmuka. Berikut ini penjelasan tiga rancangan antarmuka yang dibangun dalam Tugas Akhir ini.

4.3.1 Antarmuka Halaman Input Data

Halaman pengelompokan data digunakan oleh administrator untuk memasukkan data artefak dan menautkan data geospasial dari data artefak tersebut. Semua runtutan proses dilakukan di dalam aplikasi dekstop. Sehingga administrator tidak perlu mengetahui seluruh proses yang terjadi ketika menautkan data geospasial. Rancangan ini dapat dilihat pada Gambar 4.7.



The image shows a web form with the following fields:

- Nama :
- Provinsi :
- Kota/Kabupaten ... :
- Tahun Pembuatan :
- Kecamatan :
- Bahan Pembuatan :
- Ukuran :
- Kondisi :

At the bottom right, there is a button labeled "Create RDF".

Gambar 4.7 Rancangan Antarmuka Halaman Input Data

Pada rancangan antarmuka di atas terdapat sebuah label dan form untuk menginputkan data artefak. Tombol yang berada di kanan bawah digunakan untuk melakukan penautan data geospasial dan generate rdf.

4.3.2 Antarmuka Halaman Menampilkan Data Artefak

Halaman ini digunakan oleh user untuk melihat Data Artefak yang ada Indonesia. Di dalam halaman ini terdapat sebuah tabel yang berisi data artefak Indonesia. Rancangan antarmuka halaman ini dapat dilihat pada gambar 4.8 dan gambar 4.9.



Gambar 4.8 Rancangan Antarmuka Halaman Menampilkan Data Artefak

Pada gambar 4.8 terdapat tabel yang berisikan data artefak yang telah diinputkan oleh Administrator dan sebuah dropdown untuk melihat data artefak berdasarkan kabupaten/kota.



Gambar 4.9 Rancangan Antarmuka Halaman Menampilkan Data Artefak Berdasarkan Kabupaten/Kota

Pada gambar 4.9 terdapat daftar artefak berdasarkan kabupaten/kota dalam bentuk *list* ke bawah.

4.3.3 Antarmuka Halaman Menampilkan Detail Artefak

Halaman ini digunakan oleh user untuk melihat Detail Informasi tentang Artefak yang dipilih. Di dalam halaman ini terdapat sebuah paragraph yang berisi detail informasi tentang artefak dan terdapat sebuah mpas dengan sebuah marker yang menandakan tempat ditemukannya artefak. Rancangan antarmuka halaman ini dapat dilihat pada gambar 4.9.



Gambar 4.10 Rancangan Antarmuka Halaman Menampilkan Detail Artefak

Pada gambar 4.10 terdapat paragraph yang berisikan detail informasi mengenai artefak dan maps dengan sebuah marker yang menandakan tempat ditemukan artefak. Selain itu dalam detail informasi artefak terdapat link untuk melihat informasi spasial peninggalan warisan budaya.

BAB V IMPLEMENTASI

Pada bab ini akan dibahas implementasi dari rancangan sistem yang telah dianalisis dan dirancang dalam Bab IV. Implementasi ini meliputi implementasi kelas pada lapisan data, kontrol, dan antarmuka serta implementasi tampilan antarmuka pengguna.

5.1. Lingkungan Implementasi

Lingkungan pengembangan sistem yang digunakan untuk mengembangkan Tugas Akhir ini dilakukan pada lingkungan perangkat keras dan lingkungan perangkat lunak. Perangkat keras yang digunakan dalam pengembangan sistem berupa laptop dengan spesifikasi AMD® APU A8-6410 Processor (4CPUs) ~2.00 GHz dan memori sebesar 8GB. Spesifikasi perangkat lunak yang digunakan untuk pengembangan sistem adalah sebagai berikut:

- Sistem Operasi Windows 8.1 Professional 64 bit
- Web Server berupa XAMPP v3.2.1.
- Netbeans dan Sublime Text 3 sebagai IDE untuk implementasi aplikasi.
- Library RDF API untuk menampilkan isi dari file RDF ke dalam browser.
- Geonames API.
- Google Maps Geocoding API.
- Star UML 5 untuk merancang diagram kasus penggunaan

5.2. Implementasi Data

Pada subbab ini akan dibahas implementasi dari rancangan data yang telah dibahas pada Bab IV. Terdapat satu data yang

digunakan untuk melakukan penautan data geospasial, yaitu data artefak yang digunakan untuk input.

Data yang digunakan adalah berdasarkan data dari Museum Indonesia. Data input ini berisi kumpulan data artefak dengan informasi-informasi yang dibutuhkan meliputi Nama,Provinsi,Kabupaten/Kota,Kecamatan,Ukuran,Bahan,Tahun ,Kondisi.

Nama	Kecamatan	Kabupaten/Kota	Provinsi	Tahun Pembuatan	Bahan	Panjang	Lebar	Tinggi
Keramik	Kecamatan	Kabupaten/Kota	Provinsi	Tahun	Bahan	Panjang	Lebar	Tinggi
Keramik	Kecamatan	Kabupaten/Kota	Provinsi	Tahun	Bahan	Panjang	Lebar	Tinggi
Keramik	Kecamatan	Kabupaten/Kota	Provinsi	Tahun	Bahan	Panjang	Lebar	Tinggi

Gambar 5.1 Contoh Penggunaan Input Data Artefak

Gambar 5.1 adalah contoh input data artefak. Atribut dari kolom C-N akan dijelaskan pada tabel 4.1

Tabel 5.1 Penjelasan Atribut dari Kolom Input Data Artefak

Nama Kolom	Keterangan
Kolom C	Nama Artefak
Kolom D	Tempat Ditemukannya Artefak(Kecamatan)
Kolom E	Tempat Ditemukannya Artefak(Kabupaten/Kota)
Kolom F	Tempat Ditemukannya Artefak(Provinsi)
Kolom G	Tahun Pembuatan Artefak
Kolom H	Bahan Pembuatan Artefak
Kolom I	Panjang Artefak(Ukuran)
Kolom J	Lebar Artefak(Ukuran)
Kolom K	Tinggi Artefak(Ukuran)

Kolom L	Tebal Artefak(Ukuran)
Kolom M	Diameter Artefak(Ukuran)

5.3. Implementasi Proses Aplikasi

Pada subbab ini akan dibahas berbagai fungsi atau proses yang berjalan pada sistem data artefak. Ada dua proses utama yang dilakukan oleh sistem pada tugas akhir ini, yaitu proses menjadikan data input menjadi sebuah file rdf yang isinya data geospasial telah ditautkan dan proses menampilkan data. Runtutan proses yang berlangsung pengelolaan data input berlangsung secara otomatis yang dilakukan oleh Aplikasi berbasis desktop.

5.3.1. Implementasi Class FormArtefak

Pada class FormArtefak ini memproses data yang telah diinputkan oleh Administrator. Fungsi jButton1ActionPerformed merupakan fungsi yang digunakan untuk memanggil fungsi GenerateRDF yang terdapat pada class GenerateRDF. Kode Sumber 5.1 merupakan kode untuk Fungsi jButton1ActionPerformed.

```

Private void
jButton1ActionPerformed(java.awt.event.Action
Event evt) {
    try {
        String nama = Nm.getText();
        String provinsi = prov.getText();
        String kota = kab.getText();
        String desa = des.getText();
        String kondisi = kondisi.getText();
        String ukuran1 = ukuran.getText();
        String bahan1 = bahan.getText();
        String tahun = thn.getText();
        String
link_provinsi,link_kota,link_kec,link_wikiped
ial,link_wikipedia2,link_wikipedia3;

```

```

        nama = nama.replaceAll(" ", "_");
        kondisil = kondisil.replaceAll("
", "_");
        ukuran1 = ukuran1.replaceAll("
", "_");
        tahun = tahun.replaceAll(" ", "_");
        bahan1 = bahan1.replaceAll("
", "_");
        String provinsil =
provinsi.replaceAll(" ", "_");
        String kotal = kota.replaceAll("
", "_");
        String desal = desa.replaceAll("
", "_");
        Geonames link_geonames = new
Geonames();
        link_provinsi =
link_geonames.cekLokasiProvinsi(provinsi);
        link_kota =
link_geonames.cekLokasiKota(kota);
        link_kec =
link_geonames.cekLokasiDesa(desa, kota);

        if(link_kec=="http://geonames.org/0")
        {
            keca="";
        }
        else
        {
            keca=link_kec;
        }
        Geonames link_wikipedia = new
Geonames();
        link_wikipedia1 =
link_wikipedia.cekWikipedia(provinsi);
        link_wikipedia2 =
link_wikipedia.cekWikipedia(kota);

```

```

        String filename =
JOptionPane.showInputDialog(null, "Masukkan
Nama File", "Nama File", 1);

        String filename1=
(filename+".rdf");

        GenerateRDF grdf = new
GenerateRDF();

        grdf.GenerateRDF(filename1, nama,
provinsil, kotal,
desal, bahan1, ukuran1, kondisil, tahun, link_prov
insi,
link_kota, keca, link_wikipedial, link_wikipedia
2);

    }

    catch (Exception ex) {

        Logger.getLogger (FormArtefak.class.getName ())
        .log (Level.SEVERE, null, ex);

    }

}

```

Kode Sumber 5.1 Fungsi jButton1ActionPerformed

5.3.2. Fungsi CekLokasiProvinsi

Fungsi ini digunakan untuk menautkan data provinsi lokasi ditemukannya artefak. Proses dari fungsi ini adalah melakukan pencarian data geospasial yang sesuai dengan masukan dan melakukan set *FeatureCode* ADM1 yang menandakan administratif (provinsi). Kemudian akan didapat beberapa data geospasial yang dimasukkan ke dalam array dan kemudian diambil ID Geonames saja pada array pertama. Setelah itu dari ID Geonames tersebut digabungkan dengan string www.geonames.org/ sehingga menjadi www.geonames.org/IDGeonames. Selain itu juga diambil data koordinat. Kemudian dari semua hasil tersebut ditautkan ke data lokasi provinsi ditemukannya artefak dalam skema metadata yang akan dijadikan file RDF. Kode sumber 5.2 adalah kode untuk fungsi cekLokasiProvinsi.

```

public String cekLokasiProvinsi (String
provinsi) throws Exception
{
    Webservice.setUser ("hanafi");
    ToponymSearchCriteria searchCriteria =
new ToponymSearchCriteria ();
    searchCriteria.setQ (provinsi);

searchCriteria.setFeatureCode ("ADM1");
    searchCriteria.setMaxRows (10);
    ToponymSearchResult searchResult =
WebService.search (searchCriteria);
    b=1;
    int
bk=searchResult.getTotalResultsCount ();
    if (bk>0)
    {

searchResult.getToponyms ().stream ().forEach (n
ew Consumer<Toponym> () {
        @Override
        public void accept (Toponym
toponym) {

//System.out.println (toponym.getGeoNameId ()+
toponym.getName ()+" ");

link_kota [b]=toponym.getGeoNameId ();

lat=toponym.getLatitude ();

longi=toponym.getLongitude ();
        b++;
    }
});
    link_kotak
    ="http://geonames.org/"+link_kota [1]+"_"+lat+"
"+longi;

```

```

JOptionPane.showMessageDialog(null,"Data
Provinsi ada di Geonames");
    }
    else {

link_kotak="http://geonames.org/0_0_0_0";
JOptionPane.showMessageDialog(null,"Data
Provinsi tidak ada di Geonames");
    }

    return link_kotak;
}

```

Kode Sumber 5.2 Fungsi cekLokasiProvinsi pada class Geonames

5.3.3. Fungsi CekLokasiKota

Fungsi cekLokasiKota digunakan untuk mendapatkan geonames ID dari kabupaten/kota tempat ditemukan artefak yang selanjutnya akan digunakan untuk menautkan. Proses dari fungsi ini adalah melakukan pencarian data geospasial yang sesuai dengan masukan dan melakukan set *FeatureCode* ADM2 yang menandakan administrative2(kabupaten/kota). Kemudian akan didapat beberapa data geospasial yang dimasukkan ke dalam array dan kemudian diambil ID Geonames saja pada array pertama. Setelah itu dari ID Geonames tersebut digabungkan dengan string www.geonames.org/ sehingga menjadi www.geonames.org/IDGeonames. Kemudian dari hasil tersebut ditautkan ke data lokasi kabupaten/kota ditemukannya artefak dalam skema metadata yang akan dijadikan file RDF. Kode sumber 5.3 adalah kode untuk fungsi cekLokasiKota.

```

public String cekLokasiKota(String kota)
throws Exception
{

```

```

        Webservice.setUsername("hanafi");
        ToponymSearchCriteria searchCriteria =
new ToponymSearchCriteria();
        searchCriteria.setQ(kota);

searchCriteria.setFeatureCode("ADM2");
        searchCriteria.setMaxRows(10);
        ToponymSearchResult searchResult =
WebService.search(searchCriteria);
        b=1;
        int
bk=searchResult.getTotalResultsCount();
        if(bk>0)
        {

searchResult.getToponyms().stream().forEach(n
ew Consumer<Toponym>() {
            @Override
                public void accept(Toponym
toponym) {
                    //
System.out.println(toponym.getGeoNameId()+
toponym.getName()+" ");

link_kota[b]=toponym.getGeoNameId();

lat=toponym.getLatitude();

longi=toponym.getLongitude();
                    b++;
                }
            });
            link_kotak
            =
"http://geonames.org/"+link_kota[1]+"_"+lat+"
_"+longi;

JOptionPane.showMessageDialog(null,"Data
Kabupaten/Kota ada di Geonames");
        }
        else{

```

```

link_kotak="http://geonames.org/0_0_0";

JOptionPane.showMessageDialog(null,"Data
Kabupaten/Kota tidak ada di Geonames");
    }

    return link_kotak;
}

```

Kode Sumber 5.3 Fungsi cekLokasiKota pada class Geonames

5.3.4 Fungsi CekLokasiKec

Fungsi cekLokasiKec digunakan untuk mendapatkan geonames ID dari Kecamatan tempat ditemukan artefak yang selanjutnya akan digunakan untuk menautkan. Proses dari fungsi ini adalah melakukan pencarian data geospasial yang sesuai dengan masukan dan melakukan set *FeatureCode* ADM3 yang menandakan administrative3(kecamatan). Kemudian akan didapat beberapa data geospasial yang dimasukkan ke dalam array dan kemudian diambil koordinatnya. Setelah itu koordinat tersebut akan digunakan dalam fungsi CekGeo untuk mendapatkan data kecamatan yang tepat. Jika sudah mendapat data kecamatan yang tepat diambil ID Geonames dari data tersebut. Setelah itu dari ID Geonames tersebut digabungkan dengan string www.geonames.org/ sehingga menjadi www.geonames.org/IDGeonames. Kemudian dari hasil tersebut ditautkan ke data lokasi kecamatan ditemukannya artefak dalam skema metadata yang akan dijadikan file RDF. Kode sumber 5.4 adalah kode untuk fungsi cekLokasiKec. Kode Sumber 5.5 adalah kode untuk fungsi cekGeo.

```

public String cekLokasiKec(String kec,String
kota) throws Exception
{

```

```

        Webservice.setUsername("hanafi");
        ToponymSearchCriteria searchCriteria =
new ToponymSearchCriteria();
            searchCriteria.setQ(kec);

searchCriteria.setFeatureCode("ADM3");
        ToponymSearchResult searchResult =
WebService.search(searchCriteria);
            b=1;
            Geocoding                link_kec=new
Geocoding();
            int
bk=searchResult.getTotalResultsCount();
            if(bk>0)
            {

searchResult.getToponyms().stream().forEach(n
ew Consumer<Toponym>() {
                @Override
                public void accept(Toponym
toponym) {

//System.out.println(toponym.getGeoNameId()+
toponym.getName()+" ");

lat=toponym.getLatitude();

longi=toponym.getLongitude();

pecahkota=kota.split(" ");
                gabung="";
                int
b=pecahkota.length-1;
                for(int
a=1;a<pecahkota.length;a++)
                {
                    if(a==b)
                    {

```



```

                                gabungan=gabung
+ pecahkota[a];
                                }
                                else
                                gabungan=gabung
+ pecahkota[a]+ " ";
                                }

hasil=link_kec.cekGeo(lat,longi);

if(hasil.compareTo(kota)==0)
    {

link_desa=toponym.getGeoNameId();
    }
    else if
(hasil.compareTo(gabung)==0)
    {

link_desa=toponym.getGeoNameId();
    }
    else
    link_desa = 0;
    }
    });
    link_kotak =
"http://geonames.org/"+link_desa+"_"+lat+"_"+
longi;
JOptionPane.showMessageDialog(null,"Data
Kecamatan ada di Geonames");
    }
    else
    {

    link_kotak="http://geonames.org/0_0_0";
JOptionPane.showMessageDialog(null,"Data
Kecamatan tidak ada di Geonames");
    }
}

```

```

        return link_kotak;
    }

```

Kode Sumber 5.4 Fungsi cekLokasiKec pada class Geonames

```

public String cekGeo(double lat,double lng) {

    // making url request
    try {

        URL url = new
        URL("https://maps.googleapis.com/maps/api/geo
        code/json?latlng="+lat+", "+lng+"&sensor=true&
        key=AIzaSyBPxwjvBsDggStlbcDUtf7h9w9VoGAlqjk")
        ;

        // making connection
        HttpURLConnection conn =
        (HttpURLConnection) url.openConnection();

        conn.setRequestMethod("GET");

        conn.setRequestProperty("Accept",
        "application/json");

        if
        (conn.getResponseCode() != 200) {

            throw new
            RuntimeException("Failed : HTTP error code : "
            +
            conn.getResponseCode());

        }

        // Reading data's from url
        BufferedReader br = new
        BufferedReader(new InputStreamReader(

        (conn.getInputStream())));

        String output;
        String out="";

```

```

        System.out.println("Output
from Server .... \n");
        while      ((output      =
br.readLine()) != null) {

            //System.out.println(output);
            out+=output;
        }
        //      Converting      Json
formatted string into JSON object
        JSONObject      json      =
(JSONObject) JsonSerializer.toJSON(out);
        JSONArray
results=json.getJSONArray("results");
        JSONObject      rec      =
results.getJSONObject(0);
        JSONArray
address_components=rec.getJSONArray("address_
components");

        for(int
i=0;i<address_components.size();i++){
            JSONObject rec1 =
address_components.getJSONObject(i);

//trace(rec1.getString("long_name"));
            JSONArray
types=rec1.getJSONArray("types");
            String
comp=types.getString(0);

if(comp.equals("locality")){

        System.out.println("city      -----
"+rec1.getString("long_name"));
            }
            else
if(comp.equals("administrative_area_level_2")
){

```

```

        System.out.println("country      _____
"+rec1.getString("long_name"));

hasil_geo=rec1.getString("long_name");
        }
    }
    String formatted_address =
rec.getString("formatted_address");

//pecah_geo=formatted_address.split(", ");

//panjang=pecah_geo.length;

//hasil_geo=pecah_geo[panjang-3];

    System.out.println("formatted_address—
_____"+formatted_address);
        conn.disconnect();
    } catch (MalformedURLException e)
{
        // TODO Auto-generated
catch block
        e.printStackTrace();
    } catch (IOException e) {
        // TODO Auto-generated
catch block
        e.printStackTrace();
    }
    return hasil_geo;
}

```

Kode Sumber 5.5 Fungsi cekGeo pada Class Geocoding

5.3.5 Implementasi Class GenerateRDF

Pada class GenerateRDF ini memproses data yang telah diinputkan oleh Administrator dan juga data artefak yang telah ditautkan dengan geonames dan Wikipedia untuk dijadikan skema metadata berupa file RDF. Dalam class ini juga digunakan untuk

menambahkan data artefak ke database yang berguna pada saat menampilkan data artefak yang telah ditambahkan. Kode Sumber 5.6 merupakan kode untuk Fungsi yang ada pada class GenerateRDF.

```

public void GenerateRDF(String filename,String
nama,String provinsi,String kota,String
desa,String bahan,String Ukuran,String
kondisi,String tahun,String
link_geonames1,String link_geonames2,String
link_geonames3,String link_wiki1,String
link_wiki2,String link_wiki3) throws Exception
{
    String pecahlink_1[];
    String pecahlink_2[];
    String pecahlink_3[];
    String
lat_1,longi_1,lat_2,longi_2,lat_3,longi_3;
    pecahlink_1=link_geonames1.split("_");
    link_geonames1=pecahlink_1[0];
    lat_1=pecahlink_1[1];
    longi_1=pecahlink_1[2];
    pecahlink_2=link_geonames2.split("_");
    link_geonames2=pecahlink_2[0];
    lat_2=pecahlink_2[1];
    longi_2=pecahlink_2[2];
    pecahlink_3=link_geonames3.split("_");
    link_geonames3=pecahlink_3[0];
    lat_3=pecahlink_3[1];
    longi_3=pecahlink_3[2];
    File newFile = new
File("C:/xampp/htdocs/siartefak/rdf/"+filenam
e);
    if(newFile.exists())
        JOptionPane.showMessageDialog(null,"Data
Artefak Sudah Ada");
    else{

```

```

    FileWriter      fileW      =      new
FileWriter(newFile);
    try      (BufferedWriter      buffW      =      new
BufferedWriter(fileW)) {
        buffW.write("<rdf:RDF\n" +
            "
            xmlns:rdf=\"http://www.w3.org/1999/02/22
-rdf-syntax-ns#\n" +
            "
            xmlns:owl=\"http://www.w3.org/2002/07/ow
l#\n" +
            "
            xmlns:dc=\"http://purl.org/dc/elements/1
.1/\n" +
            "
            xmlns:han=\"http://www.han.com/artifact#
\n" +
            "
            xmlns:rdfs=\"http://www.w3.org/2000/01/r
df-schema#\n" +
            "
            xmlns:geo=\"http://rdf.fm/geographic-
location#\n" +
            "
            xmlns:pos=\"http://www.w3.org/2003/01/geo/wgs
84_pos#\n" +
            "      xmlns:cidoc-
crm=\"http://www.cidoc-crm.org/cidoc-
crm#\n">\n" +
            "\n" +
            "\n" +
            "      <owl:ObjectProperty
rdf:about=\"http://www.cidoc-crm.org/cidoc-
crm#P53_has_former_or_current_location\n"/>\n"
+
            "      <owl:ObjectProperty
rdf:about=\"http://www.cidoc-crm.org/cidoc-
crm#P43_has_dimension\n"/>\n" +

```

```

"          <owl:ObjectProperty
rdf:about="http://www.cidoc-crm.org/cidoc-
crm#P44_has_condition"/>\n" +
"          <owl:ObjectProperty
rdf:about="http://www.cidoc-crm.org/cidoc-
crm#P4_has_time-span"/>\n" +
"          <owl:ObjectProperty
rdf:about="http://www.cidoc-crm.org/cidoc-
crm#P45_consist_of"/>\n" +
"          <owl:DatatypeProperty
rdf:about="https://www.w3.org/2003/01/geo/wg
s84_pos#latitude"/>\n" +
"          <owl:DatatypeProperty
rdf:about="https://www.w3.org/2003/01/geo/wg
s84_pos#longitude"/>\n" +
"          <rdf:Description
rdf:about="http://www.cidoc-crm.org/cidoc-
crm#" + nama + "\">\n" +
"\n" +
"          <rdf:type
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#E19.Physical-Object"/>\n" +
"\n" +
"          <cidoc-
crm:P53_has_former_or_current_location
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#" + provinsi + "\">\n" +
"          <cidoc-
crm:P53_has_former_or_current_location
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#" + kota + "\">\n" +
"          <cidoc-
crm:P53_has_former_or_current_location
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#" + desa + "\">\n" +
"          <cidoc-
crm:P43_has_dimension
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#" + Ukuran + "\">\n" +
"          <cidoc-
crm:P4 has time-span

```

```

rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#+tahun+"/>\n" +
"                <cidoc-
crm:P44_has_condition
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#+kondisi+"/>\n" +
"                <cidoc-
crm:P45_consist_of
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#+bahan+"/>\n" +
"
  </rdf:Description>\n" +
"  \n" +
"    <rdf:Description
rdf:about="http://www.cidoc-crm.org/cidoc-
crm#+Ukuran+"/>\n" +
"  \n" +
"    <rdf:type
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#E54_Dimension"/>\n" +
"  \n" +
"
  </rdf:Description>\n" +
"    <rdf:Description
rdf:about="http://www.cidoc-crm.org/cidoc-
crm#+bahan+"/>\n" +
"  \n" +
"    <rdf:type
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#E57_Material"/>\n" +
"  \n" +
"
  </rdf:Description>\n" +
"    <rdf:Description
rdf:about="http://www.cidoc-crm.org/cidoc-
crm#+kondisi+"/>\n" +
"  \n" +
"    <rdf:type
rdf:resource="http://www.cidoc-
crm.org/cidoc-crm#E3_Condition_State"/>\n" +

```



```

        "\n" +
        "
        </rdf:Description>\n" +
        "      <rdf:Description
rdf:about=\"http://www.cidoc-crm.org/cidoc-
crm#+tahun+\">\n" +
        "\n" +
        "          <rdf:type
rdf:resource=\"http://www.cidoc-
crm.org/cidoc-crm#E52_Time-Span\"/>\n" +
        "\n" +
        "      </rdf:Description>\n" +
        "      <rdf:Description
rdf:about=\"http://www.cidoc-crm.org/cidoc-
crm#+provinsi+\">\n" +
        "\n" +
        "          <rdf:type
rdf:resource=\"http://rdf.fm/geographic-
location#Province\"/>\n" +
        "
        <rdfs:seeAlso>"+link_geonames1+"</rdfs:s
eeAlso>\n" +
        "
        <rdfs:seeAlso>"+link_wiki1+"</rdfs:seeAl
so>\n" +
        "
        <pos:latitude>"+lat_1+"</pos:latitude> \n" +
        "
        <pos:longitude>"+longi_1+"</pos:longitude> \n"
        +
        "
        </rdf:Description>\n" +
        "\n" +
        "      <rdf:Description
rdf:about=\"http://www.cidoc-crm.org/cidoc-
crm#+kota+\">\n" +
        "\n" +

```

```

"                                <rdf:type
rdf:resource=\\"http://rdf.fm/geographic-
location#District\"/>\n" +
"
    <rdfs:seeAlso>"+link_geonames2+"</rdfs:s
eeAlso>\n" +
"
    <rdfs:seeAlso>"+link_wiki2+"</rdfs:seeAl
so>\n" +
"
<pos:latitude>"+lat_2+"</pos:latitude> \n" +
"
<pos:longitude>"+longi_2+"</pos:longitude> \n"
+
"\\n" +
"
</rdf:Description>\n" +
"\\n" +
"    <rdf:Description
rdf:about=\\"http://www.cidoc-crm.org/cidoc-
crm#" +desa+"\">\n" +
"\\n" +
"
    <rdfs:seeAlso>"+link_geonames3+"</rdfs:s
eeAlso>\n" +
"
    <rdfs:seeAlso>"+link_wiki3+"</rdfs:seeAl
so>\n" +
"
<pos:latitude>"+lat_3+"</pos:latitude> \n" +
"
<pos:longitude>"+longi_3+"</pos:longitude> \n"
+
"\\n" +
"                                <rdf:type
rdf:resource=\\"http://www.han.com/artifact#Su
bdistrict\"/>\n" +
"\\n" +
"
</rdf:Description>\n" +

```

```

        "\n" +
        "</rdf:RDF>");
    Connection koneksi =
    DriverManager.getConnection("jdbc:mysql://localhost/siartefak","root","");
    String link
    ="C:/xampp/htdocs/siartefak/rdf/";
    String link_rdf=(link+filename);
    koneksi.createStatement().execute("Insert
    into artefak(`nama_artefak`,`link_rdf`)
    values('"+nama+"','"+link_rdf+"')");

    JOptionPane.showMessageDialog(null,"Generate
    Skema Metadata berhasil");
    }
}
}

```

Kode Sumber 5.6 Fungsi GenerateRDF

5.3.6 Implementasi Menampilkan Data Artefak

Proses menampilkan data artefak pertama membaca file rdf salah satu artefak dengan menggunakan RAP Library. Membaca file rdf diawali dengan melakukan pendataan seluruh object property. Kode sumber 5.7 adalah kode program yang berisikan fungsi untuk mendapatkan seluruh object property.

```

$model = ModelFactory::getDefaultModel();
// Load and parse document
$model->load($base);
// Output model as HTML table
$it = $model->getStatementIterator();
// Traverse model and output statements
$nn="";
while ($it->hasNext()) {
    $statement = $it->next();
    $tes[$it->
    >getCurrentPosition()]=$statement->
    >getLabelSubject();
}

```

```

    $tes2[$it-
>getCurrentPosition()]= $statement-
>getLabelPredicate();
    $tes3[$it-
>getCurrentPosition()]= $statement-
>getLabelObject();
    $nama=explode("#", $tes[$it-
>getCurrentPosition()]);
    $obj=explode("#", $tes3[$it-
>getCurrentPosition()]);
    if(isset($obj[1]))
    {
        if($obj[1]=="ObjectProperty")
        {
            $nn=$nn.$nama[1];
            $nn=$nn.", ";
        }
    }
}

```

Kode Sumber 5.7 Fungsi mendapat seluruh object property

Kemudian setelah mendapatkan seluruh object property langkah selanjutnya adalah menampilkan informasi yang ada dalam file rdf artefak. Kode sumber 5.8 adalah kode program yang berisikan fungsi untuk menampilkan informasi dari file rdf artefak.

```

$nm=explode(", ", $nn);
$jml=count($nm)-1;
$kal="";
$link="";
$n=0;
for($i=0;$i<$jml;$i++)
{
    //echo $nm[$i]."<BR>";
    $it = $model->getStatementIterator();
    while ($it->hasNext()) {
        $statement = $it->next();
        $tes[$it->getCurrentPosition()]= $statement->getLabelSubject();
        $tes2[$it->getCurrentPosition()]= $statement-
>getLabelPredicate();
    }
}

```

```

$tes3[$it->getCurrentPosition()]=$statement->getLabelObject();
$nama=explode("#",$tes[$it->getCurrentPosition()]);
$pre=explode("#",$tes2[$it->getCurrentPosition()]);
$obj=explode("#",$tes3[$it->getCurrentPosition()]);
if($nama[1]==$artefak && $pre[1]==$nm[$i])
{
    if($pre[1]=="P43_has_dimension"){
        echo $nama[1]." ". "memiliki dimensi antara lain." ".$obj[1];
    }
    else if($pre[1]=="P4_has_time-span"){
        echo $nama[1]." ". "merupakan artefak yang ada pada." ".$obj[1];
    }
    else if($pre[1]=="P53_has_former_or_current_location"){
        echo $nama[1]." ". "ditemukan di " ".$obj[1];
    }
    else if($pre[1]=="P45_consist_of"){
        echo $nama[1]." ". "terbuat dari" ".$obj[1];
    }
    else if($pre[1]=="P44_has_condition"){
        echo $nama[1]." ". "ditemukan dalam keadaan" ".$obj[1];
    }
}
}
}
}
}

```

Kode Sumber 5.8 Kode Program untuk menampilkan informasi dari file rdf artefak

Setelah itu akan ditampilkan informasi lokasi ditemukannya yang telah memiliki tautan dari membaca file rdf artefak. Kode sumber 5.9 adalah kode program yang berisikan fungsi untuk menampilkan informasi lokasi dengan tautannya dari file rdf artefak.

```

$n=1;
$it = $model->getStatementIterator();
while ($it->hasNext()) {
    $statement = $it->next();
    $tes[$it->getCurrentPosition()]=$statement->getLabelSubject();
    $tes2[$it->getCurrentPosition()]=$statement-
>getLabelPredicate();
}

```

```

$tes3[$it->getCurrentPosition()]=$statement->getLabelObject();
$nama=explode("#",$tes[$it->getCurrentPosition()]);
$pre=explode("#",$tes2[$it->getCurrentPosition()]);
if($pre[1]=="seeAlso")
{
    if($n%2==0)
    {
        if($tes3[$it->getCurrentPosition()]=="" || $tes3[$it-
>getCurrentPosition()=="http://geonames.org/0")
            echo "";
        else
            echo "Inilah halaman wikipedia dari "<a href=".$tes3[$it-
>getCurrentPosition()]">".$nama[1]."</a>".<BR>";
    }
    else
    {
        if($tes3[$it->getCurrentPosition()]=="" || $tes3[$it-
>getCurrentPosition()=="http://geonames.org/0")
            echo "";
        else
            echo "Inilah halaman geonames dari "<a href=".$tes3[$it-
>getCurrentPosition()]">".$nama[1]."</a>".<BR>";
    }
    $n++;
}
}
}

```

Kode Sumber 5.9 Kode Program untuk menampilkan informasi lokasi dengan tautannya dari file rdf artefak

Dalam sistem ini juga dapat melihat data artefak berdasarkan lokasi kabupaten/kota. Pengguna akan memilih kabupaten/kota, kemudian system akan menampilkan data artefak yang ada di kabupaten/kota berdasarkan pilihan pengguna. Kode sumber 5.10 adalah kode program yang berisikan fungsi untuk menampilkan data artefak sesuai dengan kabupaten/kota yang telah dipilih.

```

<?php
$url =
"https://maps.googleapis.com/maps/api/geocode/js

```

```

on?address=$lokasi&sensor=false&key=AIzaSyCioHnk
UvZ3re2kj8cEfkF2ik-6LMyuON8";
$result = file_get_contents($url);
$result_json = json_decode($result);
echo '<div id="HasilPencarian">';
if ($result_json->status == 'OK')
{
    $lat = $result_json->results[0]->geometry-
>location->lat;
    $lng = $result_json->results[0]->geometry-
>location->lng;
    $id= $result_json->results[0]->geometry-
>bounds->northeast->lat;
    $id2= $result_json->results[0]->geometry-
>bounds->northeast->lng;
    $id3= $result_json->results[0]->geometry-
>bounds->southwest->lat;
    $id4= $result_json->results[0]->geometry-
>bounds->southwest->lng;
    $query="Select * from artefak";
    $result=mysqli_query($conn,$query);
    if($result)
    { ?>
        <h4>Berikut ini adalah daftar artefak dari
<?php echo $lokasi; ?></h4>
        <?php
while($row=mysqli_fetch_row($result))
    {
        $id_artefak=$row[0];
        $nama=$row[1];
        $link="index5.php";
        $base=$row[2];
        $model = ModelFactory::getDefaultModel();

```

```

// Load and parse document
$model->load($base);
// Output model as HTML table
$it = $model->getStatementIterator();
while ($it->hasNext()) {
    $statement = $it->next();
    /*echo "Statement number: " . $it-
>getCurrentPosition() . "<BR>";
    echo "Subject: " . $statement-
>getLabelSubject() . "<BR>";
    echo "Predicate: " . $statement-
>getLabelPredicate() . "<BR>";
    echo "Object: " . $statement-
>getLabelObject() . "<P>";*/
    $tes[$it->getCurrentPosition()]=$statement-
>getLabelSubject();
    $tes2[$it->getCurrentPosition()]=$statement-
>getLabelPredicate();
    $tes3[$it->getCurrentPosition()]=$statement-
>getLabelObject();
    if(isset($tes3[27]) && isset($tes3[28]))
    {
        $latrdf=$tes3[27];
        $longrdf=$tes3[28];
    }
}
    if($latrdf>$id3 && $latrdf<$id &&
$longrdf>$id4 && $longrdf<$id2)
    { ?>

    <a href="<?php echo
$link."?id=".$id_artefak; ?>"><?php echo
$angka.". " . $nama; ?></a>

```



```
<?php $angka++; ?>
    </br>
<?php }
} ?>
<?php }
}
else
{ echo "Error: ".$result_json->status;
}
echo '</div>';
}
?>
```

Kode Sumber 5.10 Kode Program untuk menampilkan data artefak berdasarkan kabupaten/kota

5.4 Implementasi Antarmuka Pengguna

Subbab ini akan membahas implementasi antarmuka berdasarkan rancangan antarmuka yang telah dibahas pada bab IV. Antarmuka yang akan dibahas terdiri dari antarmuka halaman input data dan halaman yang menampilkan data artefak.

5.4.1 Antarmuka Halaman Input Data

Antarmuka halaman Input data merupakan antarmuka yang diimplementasikan untuk administrator yang telah dijelaskan pada subbab 4.3.1. Berikut implementasi halaman penambahan data artefak dan membuat skema metadata berupa file rdf dapat dilihat pada Gambar 5.2.

The image shows a Java Swing window with a title bar. Inside the window, there are eight text input fields arranged vertically, each with a label to its left: 'Nama', 'Provinsi', 'Kota/Kabupaten', 'Tahun Pembuatan', 'Kecamatan', 'Bahan Pembuatan', 'Ukuran', and 'Kondisi'. At the bottom right of the window, there is a button labeled 'Create RDF'.

Gambar 5.2 Implementasi Antarmuka Halaman Penambahan Data

Untuk memulai proses input data yang akan dijadikan file RDF, administrator dapat menekan tombol “*Create RDF*”. Berikut adalah kode program untuk menampilkan antarmuka halaman input data.

```
public class CobaTA extends Object {

    // some definitions

    public static void main (String args[])
    throws Exception {
        FormArtefak fm=new FormArtefak();
        fm.setLocationRelativeTo(null);
        fm.setVisible(true);
    }
}
```

Kode Sumber 5.11 Kode Program untuk Menampilkan Antarmuka Halaman Input Data

5.4.2 Antarmuka Halaman Menampilkan Data Artefak

Berikut implementasi halaman menampilkan data artefak dapat dilihat pada Gambar 5.3 dan 5.4. Untuk Kode yang menampilkan halaman awal yang menampilkan daftar data artefak terdapat pada Kode Sumber 5.11 dan Kode untuk menampilkan daftar data artefak berdasarkan kabupaten/kota terdapat pada Kode Sumber 5.12.



Gambar 5.3 Halaman Awal Daftar Data Artefak

```

<form name='Tanya' method='POST'
action="cobadulu.php">
Lokasi:
<select class="form-control" name="keyword"
id="keyword">
    <?php
while($row=mysqli_fetch_row($result_kota)) {?>
    <option value="<?php echo($row[1]);?>">
<?php echo($row[1]);?> </option>
    <?php } ?>
    </select>
<input type="submit" name="cari"
value="Search">
</form>

```

```

<br/>
<h3>Berikut ini adalah Daftar Artefak
Seluruh Indonesia</h3>
<table class="kd-table kd-tableone"
align="center" style="width:100%">
  <thead>
    <tr>
      <th>Nama Artefak</th>
      <th>Lihat Artefak</th>
    </tr>
  </thead>
  <tbody>
    <?php
while($row=mysqli_fetch_row($result)) {?>
      <tr>
        <?php
          $id=$row[0];
          $nama=$row[1];
          $link="index3.php";
          ?>
          <td><?php echo
$nama; ?></td>
          <td><a href="<?php echo
$link."?id=".$id; ?>">Link</a></td>
        </tr>
      </tbody>
    <?php
    }
    ?>
  </table>

```

Kode Sumber 5.12 Kode Program untuk Menampilkan Antarmuka Halaman Awal Daftar Data Artefak



Gambar 5.4 Halaman Daftar Data Artefak Berdasarkan Kabupaten/Kota

```

    <h4>Berikut ini adalah daftar artefak dari
    <?php echo $lokasi; ?></h4>

    <a href="<?php echo
    $link."?id=".$id_artefak; ?>">?php echo
    $angka." " ".$nama; ?></a>
    </br>

```

Kode Sumber 5.13 Kode Program untuk Menampilkan Antarmuka Halaman Daftar Data Artefak Berdasarkan Kabupaten/Kota

5.4.3 Antarmuka Halaman Menampilkan Detail Artefak

Berikut implementasi halaman menampilkan detail artefak dapat dilihat pada Gambar 5.5. Untuk Kode yang menampilkan halaman detail artefak terdapat pada Kode Sumber 5.13.



Gambar 5.5 Halaman Detail Informasi Artefak

```
<div class="content-wrapper">
    <div class="inner-container
container">
        <div class="row">
            <div class="section-header
col-md-12">
                <h2>Deskripsi
Artefak</h2>
            </div> <!-- /.section-header
-->
        </div> <!-- /.row -->
        <div class="row">
            <div class="blog-info col-
md-12">
```

```
                                <div class="box-
content">
                                <?php

define("RDFAPI_INCLUDE_DIR", "lib/rap/api/");

include(RDFAPI_INCLUDE_DIR . "RDFAPI.php");
$id=$_GET['id'];
$dbname="monas";
$host="localhost";
$username="root";
$password="";
$conn=mysqli_connect($host,$username,$password,$
dbname);

if(mysqli_connect_errno())
{
    echo "Koneksi ke server gagal";
    exit();
}
$query="Select * from artefak where
id_artefak='$id'";
$result=mysqli_query($conn,$query);
if($result)
{
    $row=mysqli_fetch_array($result);
}
$base=$row[2];
$artefak=$row[1];
?>
<h2 class="blog-title"><?php echo
$artefak; ?></h2>
```

```

<?php
$model = ModelFactory::getDefaultModel();
$model->load($base);
// Output model as HTML table
$it = $model->getStatementIterator();
  $nn="";
  while ($it->hasNext()) {
    $statement = $it->next();
    $stes[$it->getCurrentPosition()]=$statement->getLabelSubject();
    $stes2[$it->getCurrentPosition()]=$statement->getLabelPredicate();
    $stes3[$it->getCurrentPosition()]=$statement->getLabelObject();
    $nama=explode("#",$stes[$it->getCurrentPosition()]);
    $obj=explode("#", $stes3[$it->getCurrentPosition()]);
    if(isset($obj[1]))
    {
      if($obj[1]=="ObjectProperty")
      {
        $nn=$nn.$nama[1];
        $nn=$nn.", ";
      }
    }
  }
  $nm=explode(",",$nn);
  $jml=count($nm)-1;
  $kal="";
  $link="";
  $n=0;

```



```

for($i=0;$i<$jml;$i++)
{
    $it = $model->getStatementIterator();
    while ($it->hasNext()) {
        $statement = $it->next();
        $tes[$it->getCurrentPosition()]= $statement->getLabelSubject();
        $tes2[$it->getCurrentPosition()]= $statement->getLabelPredicate();
        $tes3[$it->getCurrentPosition()]= $statement->getLabelObject();
        $nama=explode("#",$tes[$it->getCurrentPosition()]);
        $pre=explode("#",$tes2[$it->getCurrentPosition()]);
        $obj=explode("#", $tes3[$it->getCurrentPosition()]);
        if($nama[1]==$artefak && $pre[1]==$nm[$i])
        {
            if($pre[1]=="P43_has_dimension")
            { ?>
                <p><?php echo $nama[1]." ". "memiliki dimensi antara lain"." ".$obj[1].". "; ?>
                <?php
            }
            else if($pre[1]=="P4_has_time-span"){ ?>
                <?php echo $nama[1]." ". "merupakan artefak yang ada pada"." ".$obj[1].". "; ?>
                <?php
            }
            else
            if($pre[1]=="P53_has_former_or_current_location")
            ){ ?>

```

```
<?php echo $nama[1]." ". "ditemukan di ".  
".$obj[1].". "; ?>  
<?php  
}  
else if($pre[1]=="P45_consist_of"){ ?>  
<?php echo $nama[1]." ". "terbuat dari".  
".$obj[1].". "; ?>  
<?php  
}  
else if($pre[1]=="P44_has_condition"){ ?>  
    <?php echo $nama[1]." ". "ditemukan dalam  
keadaan". " ".$obj[1].". "; ?>  
    <?php  
    }  
    }  
    }  
}  
}  
$n=1;  
$lat="";  
$longi="";  
$it = $model->getStatementIterator();  
while ($it->hasNext()) {  
    $statement = $it->next();  
    $stes[$it->getCurrentPosition()]=$statement->  
    getLabelSubject();  
    $stes2[$it->getCurrentPosition()]=$statement->  
    getLabelPredicate();  
    $stes3[$it->getCurrentPosition()]=$statement->  
    getLabelObject();  
    $nama=explode("#",$stes[$it->  
    getCurrentPosition()]);
```

```

$pre=explode("#",$stes2[$it-
>getCurrentPosition()]);
if($pre[1]=="seeAlso")
{
    if($n%2==0)
    {
if($stes3[$it->getCurrentPosition()]==" " ||
$stes3[$it-
>getCurrentPosition()=="http://geonames.org/0")
{?>
<?php echo " "; ?>
<?php
}
else { ?>
<?php echo "Inilah halaman wikipedia dari ".<a
href="'. $stes3[$it-
>getCurrentPosition()]."'>". $nama[1]. ". "</a>";
?>
    <?php
}
}
else
{
if($stes3[$it->getCurrentPosition()]==" " ||
$stes3[$it-
>getCurrentPosition()=="http://geonames.org/0")
{ ?>
<?php echo " "; ?>
<?php
}
else { ?>
<?php echo "Inilah halaman geonames dari ".<a
href="'. $stes3[$it-

```

```

>getCurrentPosition()]"'>".$nama[1]."."."</a>";
?>
<?php } ?>
<?php
} ?>
</p>
<?php
$n++;
}
else if($pre[1]=="latitude"){
    if($tes3[$it->getCurrentPosition()]!=0)
    {
        $lat=$tes3[$it->getCurrentPosition()];
    }
}
else if ($pre[1]=="longitude") {
    if($tes3[$it->getCurrentPosition()]!=0)
    {
        $longi=$tes3[$it->getCurrentPosition()];
    }
} ?>
<?php
}
?>
<div id="googleMap"
style="width:500px;height:380px;"></div>
</div> <!-- /.blog-info -->
</div> <!-- /.row -->

</div> <!-- /.inner-content -->

```

```
        </div> <!-- /.content-wrapper -->
<script>
var myCenter=new google.maps.LatLng(<?php echo
$lat; ?>,<?php echo $longi; ?>);

function initialize()
{
var mapProp = {
    center:myCenter,
    zoom:5,
    mapTypeId:google.maps.MapTypeId.ROADMAP
};

var map=new
google.maps.Map(document.getElementById("googleM
ap"),mapProp);

var marker=new google.maps.Marker({
    position:myCenter,
});

marker.setMap(map);
}

google.maps.event.addDomListener(window, 'load',
initialize);
</script>
```

Kode Sumber 5.14 Kode Program untuk Menampilkan Antarmuka Detail Data Artefak

[Halaman ini sengaja dikosongkan]

BAB VI

PENGUJIAN DAN EVALUASI

Bab ini menjelaskan tentang pengujian dan evaluasi aplikasi yang dikembangkan. Pengujian dilakukan untuk mengetahui fungsionalitas aplikasi. Sistem akan diuji coba fungsionalitasnya dengan menjalankan skenario yang sudah ditentukan. Hasil evaluasi menjelaskan mengenai rangkuman pengujian yang dilakukan pada aplikasi ini.

6.1. Lingkungan Pengujian

Pada subbab ini dijelaskan mengenai gambaran lingkungan yang digunakan untuk melakukan uji coba aplikasi. Uji coba aplikasi ini menggunakan Notebook Lenovo G40-48 dengan spesifikasi sebagai berikut:

- Spesifikasi Perangkat Keras:
 - AMD® APU A8-6410 Processor (4CPUs) ~2.00 GHz
 - 8GB DDR3 1600 MHz SDRAM
- Spesifikasi Perangkat Lunak:
 - Sistem Operasi Windows 8.1 Professional 64 bit
 - Web Server berupa XAMPP v3.2.1. dengan PHP versi 5.5.15

6.2. Data Uji

Pengujian yang akan dilakukan pada sistem yang dibangun di Tugas Akhir ini menggunakan data artefak Museum Indonesia sebanyak 50 data. Data uji yang digunakan adalah data artefak dengan beberapa atribut yang meliputi Nama, Tempat Penemuan (Provinsi, Kabupaten/Kota, Kecamatan), ukuran, kondisi, tahun pembuatan.

Nama Kolom	Tempat_Artifak	Tempat_Artifak_Kab	Tempat_Artifak_Prov	Tahun_Pembuatan	Bahan	Panjang	Lebar	Tinggi	Tebal	Diameter	Kondisi
Kolom C	Musik 210
Kolom D
Kolom E

Gambar 6.1 Data Uji

Gambar di atas adalah potongan data uji yang digunakan dalam Tugas Akhir ini. Penjelasan dari setiap kolom pada gambar 6.1 di atas akan dijelaskan pada tabel berikut.

Tabel 6.1 Penjelasan Kolom pada Data Uji

Nama Kolom	Keterangan
Kolom C	Nama Artefak
Kolom D	Tempat Penemuan(Kecamatan)
Kolom E	Tempat Penemuan(Kabupaten/Kota)
Kolom F	Tempat Penemuan(Provinsi)
Kolom G	Tahun Pembuatan
Kolom H	Bahan
Kolom I	Panjang(Ukuran)
Kolom J	Lebar(Ukuran)
Kolom K	Tinggi(Ukuran)
Kolom L	Tebal(Ukuran)
Kolom M	Diameter(Ukuran)
Kolom N	Kondisi

6.3. Skenario Pengujian

Pada sub bab ini dijelaskan skenario pengujian kakas. Pengujian yang dilakukan adalah pengujian fungsionalitas dan pengujian performa. Pengujian fungsionalitas dilakukan dengan metode *black box*. Metode pengujian ini cenderung melihat hasil keluaran sistem. Pengujian performa digunakan untuk menguji kemampuan sistem dalam menambahkan data artefak, melihat data artefak, melihat detail artefak.

6.4. Pengujian Fungsionalitas

Pengujian ini dilakukan dengan sejumlah skenario sebagai tolok ukur keberhasilan sistem. Pengujian didasarkan pada kasus penggunaan yang digunakan dalam pembangunan sistem, seperti yang telah dijelaskan pada subbab 4.1.5.

6.4.1. Pengujian Menambahkan Data Artefak

Pengujian fungsionalitas untuk menambahkan data artefak dan membuat skema metadata pada sistem data artefak oleh Administrator. Skenario rinci untuk melakukan pengujian ini dijelaskan pada Tabel 6.2 dan hasil pengujian yang telah dilakukan yaitu hasil menautkan data geospasial dan pembuatan skema metadata dalam file RDF ditunjukkan pada Gambar 6. 2.

Tabel 6.2 Pengujian Fitur Menambahkan Data Artefak

ID	UJ.UC01
Referensi Kasus Penggunaan	UC01
Nama	Menambahkan Data Artefak
Kondisi Awal	Skema metadata Artefak berupa file rdf belum dibuat

Masukan	Data artefak yang diinputkan ke dalam form.
Skenario	<ol style="list-style-type: none"> 1. Pengguna membuka aplikasi. 2. Pengguna memasukkan data artefak. 3. Sistem melakukan proses menautkan data geospasial. 4. Sistem berhasil menautkan dan menjadikan file RDF.
Kondisi Akhir	Sistem menampilkan notifikasi berhasil penambahan data artefak, Data artefak berhasil ditautkan dengan Geonames dan Wikipedia, serta skema metadata berupa file RDF artefak berhasil dibuat
Hasil Pengujian	Berhasil

```

<rdf:type rdf:resource="http://www.ddac-orm.org/ddac-orm:Jawa_Timur"/>
<ddac:hasGeoName http://www.geonames.org/1842686/rdf:resource/>
<ddac:hasWikipedia http://id.wikipedia.org/wiki/Jawa_Timur/rdf:resource/>
</rdf:Description>

<rdf:Description rdf:about="http://www.ddac-orm.org/ddac-orm:Gusutan_Rakung">
<ddac:hasGeoName http://www.geonames.org/103332/rdf:resource/>
<ddac:hasWikipedia http://id.wikipedia.org/wiki/Jawa_Timur/rdf:resource/>
</rdf:Description>

<rdf:Description rdf:about="http://www.ddac-orm.org/ddac-orm:Kacamatan_Turuk">
<ddac:hasGeoName http://geonames.org/751674/rdf:resource/>
<ddac:hasWikipedia http://www.id.com/wiki/Kacamatan_Turuk">
</rdf:Description>

```

Gambar 6.2 Hasil Keluaran Pengujian Menambahkan Data Artefak

6.4.2. Pengujian Melihat Data Artefak

Melihat Daftar Data Artefak adalah melihat daftar data artefak yang telah ditambahkan oleh administrator. Skenario rinci pengujian ini dijelaskan pada Tabel 6.3 dan hasil pengujian ditunjukkan pada Gambar 6.3.

Tabel 6.3 Pengujian Fitur Melihat Data Artefak

ID	UJ.UC02
Referensi Kasus Penggunaan	UC02
Nama	Melihat Data Artefak
Kondisi Awal	Data artefak yang telah ditambahkan oleh Administrator
Masukan	-
Skenario	<ol style="list-style-type: none"> 1. User membuka <i>url</i> untuk menampilkan daftar data artefak melalui <i>Web Browser</i>. 2. Sistem akan menampilkan daftar data artefak dalam sebuah tabel. 3. User memilih kabupaten/kota 4. Sistem menampilkan daftar artefak berdasarkan kabupaten/kota yang dipilih
Kondisi Akhir	Sistem menampilkan daftar data artefak.
Hasil Pengujian	Berhasil



Gambar 6.3 Hasil Keluaran Pengujian Melihat Data Artefak

6.4.3. Pengujian Melihat Detail Artefak

Melihat Detail Artefak adalah melihat informasi setiap artefak yang telah ditambahkan oleh administrator. Informasi yang terdapat setiap artefak meliputi nama artefak, bahan pembuatan artefak, tahun pembuatan, lokasi penemuan (kecamatan, kabupaten/kota, provinsi), kondisi, dimensi, serta visualisasi peta yang ditandai sebuah *marker* tempat ditemukannya artefak. Skenario rinci pengujian ini dijelaskan pada Tabel 6.4 dan hasil pengujian ditunjukkan pada Gambar 6. 4.

Tabel 6.3 Pengujian Fitur Melihat Detail Artefak

ID	UJ.UC03
Referensi Kasus Penggunaan	UC03
Nama	Melihat Detail Artefak
Kondisi Awal	Data artefak telah ditampilkan
Masukan	-

Skenario	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar data artefak dalam sebuah tabel. 2. User mengklik salah satu artefak untuk mengetahui informasi tentang artefak tersebut. 3. Sistem menampilkan detail informasi tentang artefak yang dipilih oleh User. 4. User mendapatkan detail informasi artefak yang meliputi nama artefak, lokasi penemuan (kecamatan, kabupaten/kota, Provinsi), bahan pembuatan, tahun pembuatan, dimensi artefak, kondisi artefak dan juga link yang berisi informasi spatial peninggalan warisan budaya.
Kondisi Akhir	Sistem menampilkan detail artefak dan link yang berisi informasi spatial peninggalan warisan budaya.
Hasil Pengujian	Berhasil

dapat dilihat bahwa fungsional sistem berjalan dengan baik. Sistem dapat menambahkan data artefak menjadi file RDF dan menampilkan detail informasi artefak.

Tabel 6.6 Rangkuman Hasil Pengujian Fungsionalitas

ID	Nama	Hasil
UJ.UC01	Pengujian fitur Menambahkan Data Artefak	Berhasil
UJ.UC02	Pengujian fitur Melihat Data Artefak	Berhasil
UJ.UC03	Pengujian Fitur Melihat Detail Artefak	Berhasil

6.6.2 Evaluasi Pengujian Performa

Untuk menentukan seberapa baik aplikasi dalam menautkan data geospasial artefak maka diadakan evaluasi pengujian performa. Hasil evaluasi pengujian performa untuk seluruh skenario pada bab 6.3 dapat dilihat pada tabel 6.6 berikut.

Tabel 6.6 Hasil Evaluasi Pengujian Performa

	Geonames	Wikipedia
Kecamatan	29 data artefak	0 data artefak
Kabupaten/Kota	50 data artefak	29 data artefak
Provinsi	50 data artefak	50 data artefak

[Halaman ini sengaja dikosongkan]

BAB VII

KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan kesimpulan yang dapat diambil dalam pengerjaan tugas akhir dan saran tentang kemungkinan pengembangan yang dapat dilakukan pada tugas akhir ini di masa yang akan datang.

7.1 Kesimpulan

Dari proses analisa, perancangan, implementasi dan pengujian perangkat lunak sistem artefak peneliti dapat ditarik kesimpulan sebagai berikut:

1. Aplikasi dapat menghasilkan skema metadata baru berupa file RDF setiap artefak yang berisi data artefak sesuai dengan yang diinputkan.
2. Aplikasi dapat menautkan data lokasi penemuan artefak dengan Geonames dan Wikipedia dengan memanfaatkan library Geonames dan Geocoding API.
3. Aplikasi dapat menampilkan informasi artefak beserta link lokasi penemuan artefak hasil tautan yang berisi informasi spatial peinggalan warisan budaya.

7.2 Saran

Berikut ini beberapa saran untuk kemungkinan pengembangan perangkat lunak sistem artefak peneliti di masa mendatang berdasarkan hasil rancangan, implementasi dan uji coba yang telah dilakukan.

1. Untuk menautkan data geospasial lokasi penemuan artefak, sistem yang dibangun pada Tugas Akhir ini menggunakan Geonames dengan lisensi gratis dengan data Indonesia yang kurang lengkap. Sehingga perlu ada penambahan data pada Geonames.

[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

- [1] "Warisan Budaya dan Makna Pelestariannya," [Online]. Available: <http://www.mahyudinalmudra.com/work/detail/275/Warisan-Budaya-dan-Makna-Pelestariannya>. [Accessed 10 Juni 2016].
- [2] "What Is Metadata," Juli 2014. [Online]. Available: <http://whatis.techtarget.com/definition/metadata>. [Accessed 24 07 2016].
- [3] "Resource Description Framework (RDF) Overview," 2003. [Online]. Available: <http://www.w3.org/RDF/>. [Accessed 28 November 2015].
- [4] "Owl-Web Ontology Language Features," 2003. [Online]. Available: <http://www.w3.org/TR/owl-features/>. [Accessed 28 November 2015].
- [5] "What geospatial information," 2015. [Online]. Available: <http://www.linz.govt.nz/about-linz/our-location-strategy/what-geospatial-information>. [Accessed 11 Desember 2015].
- [6] "Linked Data," [Online]. Available: <https://www.w3.org/DesignIssues/LinkedData.html>. [Accessed 10 Juni 2016].
- [7] "Linked Geo Data," 2015. [Online]. Available: <http://linkedgeodata.org/About>. [Accessed 13 Desember 2015].
- [8] "Geonames," [Online]. Available: <http://www.geonames.org/>. [Accessed 15 Desember 2015].
- [9] "Wikipedia-Sumber Artikel Terpercaya?Pengertian dan definisi," 2012. [Online]. Available: <http://www.kamusq.com/2012/10/wikipedia-sumber-artikel-terpercaya.html>. [Accessed 11 Desember 2015].

- [1 "Geocoding," [Online]. Available:
0] <https://developers.google.com/maps/documentation/geocoding/intro#Geocoding>. [Accessed 07 Mei 2016].
- [1 "RDF API For PHP," [Online]. Available: <http://wifo5-13.informatik.uni-mannheim.de/bizer/rdfapi/>. [Accessed 1 Mei 2016].
- [1 D. Mariana and D. Dana, "Reason-Able View of Linked Data
2] for Cultural Heritage," *Third International Conference on Software, Services and Semantic Technologies S3T*, vol. 101, pp. 17-24, 2011.
- [1 "Tutorial 4: Introducing RDFS & OWL," [Online]. Available:
3] <http://www.linkeddatatools.com/introducing-rdfs-owl>. [Accessed 20 Maret 2016].
- [1 "Geocoding API," [Online]. Available:
4] <https://developers.google.com/maps/documentation/geocoding/start>. [Accessed 06 Mei 2016].
- [1 "The CIDOC CRM," [Online]. Available: <http://www.cidoc-crm.org/>. [Accessed 15 Desember 2015].
- [1 "Factmint Schemas," [Online]. Available:
6] <http://rdf.fm/#/welcome>. [Accessed 04 April 2016].
- [1 "WGS84 Geo Positioning: an RDF vocabulary," [Online].
7] Available: https://www.w3.org/2003/01/geo/wgs84_pos. [Accessed 14 Mei 2016].

BIODATA PENULIS



Mochamad Bakhtiar H, anak pertama dari dua bersaudara yang lahir di Surabaya pada 13 November 1994. Penulis telah menempuh pendidikan formal mulai dari SDN Cangu II (2000- 2006), SMP Negeri 2 Jetis Mojokerto (2006-2009), SMA Negeri 1 Puri Mojokerto (2009-2012) dan terakhir sebagai mahasiswa Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember dengan rumpun mata kuliah Manajemen Informasi (2012-2016). Lulus dari SMA penulis melanjutkan pendidikan di jurusan teknik informatika. Semasa kuliah penulis aktif mengikuti berbagai kepanitiaan diantaranya Schematics 2013, Schematics 2014, dan kegiatan lain di jurusan. Penulis juga aktif menjadi anggota Himpunan Mahasiswa Teknik Computer-Informatika dan anggota Keluarga Muslim Informatika pada tahun 2013. Penulis dapat dihubungi melalui email mochbakhtiar23@gmail.com.