

**DESAIN DAN IMPLEMENTASI *MOBILE PROXY*
STUDI KASUS PENGUMPULAN DATA LAPANGAN
BERBASIS MOBILE**

TESIS

Karya tulis sebagai salah satu syarat
Untuk memperoleh gelar Magister dari
Institut Teknologi Bandung

oleh
ARIS PRAWISUDATAMA
23215131



**Institut Teknologi Bandung
2016**

LEMBAR PEMERIKSAAN

**DESAIN DAN IMPLEMENTASI *MOBILE PROXY*
STUDI KASUS PENGUMPULAN DATA LAPANGAN
BERBASIS MOBILE**

Oleh :
ARIS PRAWISUDATAMA
NIM : 23215131

**Program Studi Magister Teknik Elektro
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung**

Bandung, April 2016

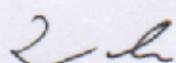
Pemeriksa I,



Angga Erwina Bayu

23215073

Pemeriksa II,



Rolian Ardi

23215070

**DESAIN DAN IMPLEMENTASI *MOBILE PROXY*
STUDI KASUS PENGUMPULAN DATA LAPANGAN
BERBASIS MOBILE**

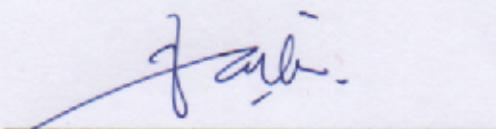
TESIS

oleh
ARIS PRAWISUDATAMA
23215131

**Program Studi Magister Teknik Elektro
Opsi Layanan Teknologi Informasi
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung**

Menyetuji

Pembimbing I



(Dr. I Gusti Bagus Baskara Nugraha)
NIP. 197601242010121001

Daftar Isi

Lembar Pengesahan	i
Abstrak	ii
Abstract	iii
Pedoman Penggunaan Tesis	v
Kata Pengantar	vi
Daftar Isi	vii
Daftar Tabel	viii
Daftar Gambar	ix
Bab I PENDAHULUAN	1
I.1 Latar Belakang	1
I.2 Rumusan Masalah	4
I.3 Tujuan Penelitian	4
I.4 Batasan Masalah	5
Bab II STUDI LITERATUR	6
II.1 <i>Generic Statistical Business Process Model (GSBPM)</i>	6
II.2 <i>Generic Statistical Information Model (GSIM)</i>	7
II.3 <i>Common Statistical Production Architecture (CSPA)</i>	7
II.4 Sistem Terdistribusi	8
II.5 <i>Grid Computing</i>	9
II.6 <i>Mobile Grid Computing</i>	9
II.7 <i>Connectionless Grid Computing</i>	10
II.8 Pola Implementasi SOA untuk Sistem Data-Intensif Terdistribusi	11
Bab III METODOLOGI	20
Daftar Pustaka	23
Lampiran 1	25

Daftar Tabel

Tabel II.1	Response Time Pengujian Skenario I	18
Tabel II.2	Total Traffic Size Pengujian Skenario I	18
Tabel II.3	Response Time Pengujian Skenario II	18
Tabel II.4	Total Traffic Size Pengujian Skenario II	18

Daftar Gambar

Gambar I.1	<i>Statistical Business Process Phases</i> dalam GSBPM	1
Gambar I.2	Ilustrasi CAPI	3
Gambar I.3	Skema Usulan, Takdir	4
Gambar II.1	<i>Literature Map</i>	6
Gambar II.2	<i>Statistical Business Process Phases</i> dalam GSBPM	6
Gambar II.3	Kategori Objek pada GSIM	7
Gambar II.4	Gambaran Sederhana Relasi Objek pada GSIM	7
Gambar II.5	Layer pada CSPA	8
Gambar II.6	Skenario Eksekusi Logik Workflow dan Web Services	14
Gambar II.7	Sinkronisasi Data dan Web Service	14
Gambar II.8	Proses Replikasi Data dan Web Service	15
Gambar II.9	Skenario Deployment Web Service dan Data Terpusat	15
Gambar II.10	Skenario Deployment Web Service Terdistribusi, Data Terpusat	16
Gambar II.11	Skenario Deployment Web Service dan Data Terdistribusi .	17
Gambar II.12	Proses Replikasi Data dengan IMDG	17
Gambar II.13	Network Utilization Proses Patching/Updating Web Service	19
Gambar II.14	Grafik CPU Utilization Proses Updating Web Service	19
Gambar III.1	Tahapan <i>Design Science Research Methodology (DSRM)</i> .	20

Bab I

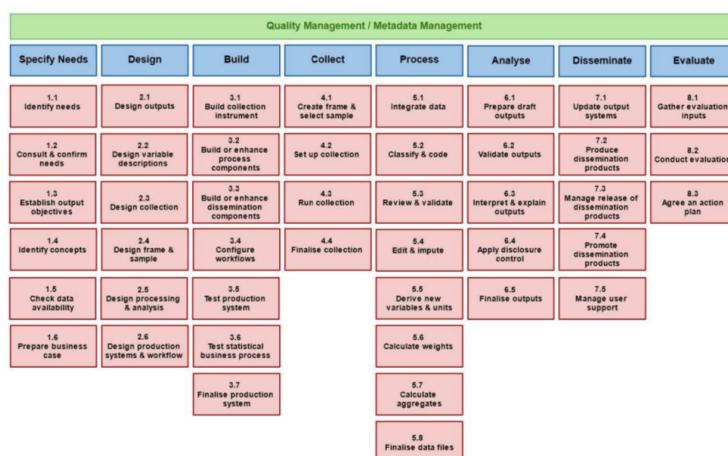
PENDAHULUAN

I.1 Latar Belakang

Badan Pusat Statistik (BPS) merupakan suatu lembaga pemerintah non-departemen yang bertanggung jawab dalam penyediaan statistik dasar. Dalam peranannya sebagai penyedia data, BPS melakukan pengumpulan data dengan 2 (dua) metode : primer dan sekunder. Pengumpulan data primer berarti BPS secara mandiri mengumpulkan data dengan menggunakan metode wawancara langsung dengan responden, baik responden individu, rumah tangga, maupun perusahaan. Sementara pengumpulan data sekunder berarti BPS memperoleh data dari pihak lain.

Dalam melakukan kegiatan perstatistikian, yang selanjutnya merujuk kepada pengumpulan data primer, BPS merujuk kepada *General Statistical Business Process Model* (GSBPM) [1], yang merupakan suatu standard arsitektur bisnis kegiatan perstatistikian yang dirumuskan oleh *United Nations Economic Commission for Europe* (UNECE). Dalam GSBPM, *Business Process Statistik* dibagi menjadi 7 (tujuh) phase : *Specify Needs, Design, Build, Collect, Process, Analyze, Disseminate, Evaluate*, dimana masing-masing phase dipecah menjadi beberapa sub-proses.

Pengumpulan dan pengolahan data dalam GSBPM tercakup dalam 3 (tiga) fase, yaitu : *Collect Phase, Process Phase, dan Analyze Phase*. *Collect phase* adalah



Gambar I.1: *Statistical Business Process Phases* dalam GSBPM

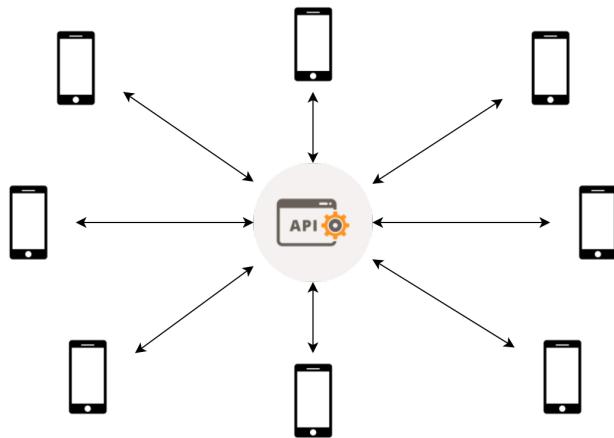
fase dimana semua informasi (data dan metadata) dikumpulkan dengan menggunakan beberapa metode pengumpulan (termasuk ekstraksi dari register dan database statistik, administratif, maupun yang lain), dan memuatkannya ke dalam suatu *environment* untuk pemrosesan lebih lanjut. *Process phase* adalah fase dimana data dibersihkan dan dipersiapkan untuk tahap berikutnya, *analysis phase*. *Collect phase* dan *process phase* dapat dilakukan secara berulang dan paralel. Fase terakhir sebelum data siap untuk didesiminasi adalah *analyze phase*. Pada tahap *analyze phase*, data ditransformasikan kedalam bentuk *statistical output* yang disesuaikan dengan kebutuhan (*fit for purpose*).

Kondisi saat ini, *process phase* dan *analyze phase* merupakan tahapan yang memiliki ketergantungan akan Teknologi Informasi dan Komunikasi (TIK) yang sangat besar. *Process phase* merupakan tahapan dimana dilakukan input data hasil pendaftaran lapangan dari format kuesioner ke dalam format digital, termasuk didalamnya pengkodean, imputasi, validasi, dan penghitungan penimbang. Sementara *analyze phase* memerlukan keterlibatan software analisis yang membantu mentransformasikan data menjadi sebuah informasi. Adapun *collect phase*, meskipun saat ini masih menggunakan pengumpulan data dengan mengadopsi *paper questionnaire*, tetapi kedepannya akan dilakukan transformasi dengan menggunakan metode *Computer Assisted Personal Interviewing* (CAPI)¹, meskipun feasibility-nya belum pernah diujicobakan².

Penggunaan metode CAPI dalam pengumpulan data yang dilakukan BPS, sedikit banyak akan mengubah paradigma pengumpulan dan pengolahan data yang selama ini telah berjalan. Pengumpulan dan pengolahan data yang selama ini merupakan dua buah tahapan yang terpisah, dengan diterapkannya CAPI maka beberapa subproses dari *process phase*, seperti pengkodean dan validasi, dapat dilakukan secara terintegrasi dengan pengumpulan data. Metode CAPI sebenarnya bukanlah sebuah hal yang baru. Metode ini sudah ada sejak beberapa dekade terakhir [2]. Bahkan sebuah penelitian yang dilakukan oleh Gary Klein dkk menyatakan pengumpulan data dengan menggunakan metode CAPI berpotensi terjadi bias, terutama dalam akurasi, *completeness*, dan *item omission* [3]. Akan tetapi, dengan semakin berkembangnya teknologi *mobile computing* yang dipadukan dengan penggunaan *Web service* [4], maka potensi bias dapat dikurangi dengan merancang beberapa *web service* yang digunakan untuk menvalidasi hasil pendataan.

¹Keterangan Dr. Said Mirza Pahlevi, M.Eng., Kepala Subdirektorat Pengembangan Basis Data, 24 Februari 2016

²Keterangan Dr. Muchammad Romzi, Kepala Subdirektorat Pengembangan Model Statistik, 4 Maret 2016

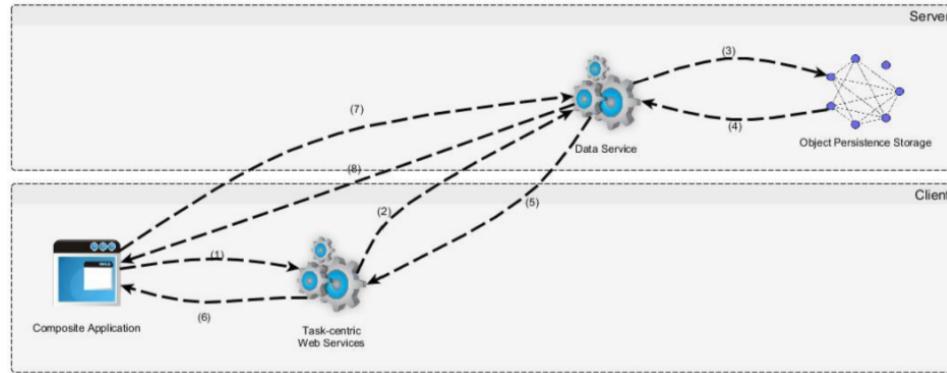


Gambar I.2: Ilustrasi CAPI

Implementasi *Web service* pada pengumpulan data dengan menggunakan CAPI bukanlah tanpa kendala. Petugas pengumpulan data harus berpindah-pindah dari satu lokasi pendataan ke lokasi yang lain untuk mengunjungi responden. Dikarenakan keterbatasan infrastruktur seperti telekomunikasi dan daya tahan baterai *device*, seiringkali sulit bagi *device* untuk selalu terhubung dengan *Web service*. *Device* dapat kapan saja berubah dari *connected node* menjadi *disconnected node* dan sebaliknya.

Brian DeRenzi dkk telah melakukan penelitian tentang cara pengumpulan data berbasis *mobile phone* pada lingkungan yang *highly disconnected* [5] dengan menggunakan *CAM Framework*. *CAM framework* [6] terbukti dapat digunakan dalam pengumpulan data dalam lingkungan yang *disconnected*, dan setelah device kembali ke *connected environment*, data yang terkumpul akan terupload ke server. Akan tetapi *CAM framework* memiliki kelemahan, antara lain : 1) CAM berbasis *fix-length text-based input*, yang membuatnya tidak cocok digunakan untuk pengumpulan data yang berbasis *data-intensive*; 2) Tidak terdapat *conflict resolution*, sehingga masih memungkinkan dua device atau lebih mengeksekusi data yang sama.

Sementara itu, untuk meningkatkan kinerja sistem yang bersifat data-intensif, Takdir dkk mengusulkan penggunaan pola terdistribusi berbasis SOA [7]. Pada implementasi pola terdistribusi berbasis SOA, digunakan metode *proxy* baik pada *workflow* (*Web service*) maupun *data-service*. Mekanisme yang digunakan dalam perancangan service pola terdistribusi mencakup 3 (tiga) hal : sinkronisasi, replikasi, dan *routing*. *Composite application* yang dijalankan pada sisi *client* akan melakukan replikasi data maupun *Web service*, kemudian data dan *Web service* tersebut digunakan secara lokal. Sementara itu, untuk menjamin konsistensi data, Takdir dkk mengakomodir mekanisme *sinkronisasi*.



Gambar I.3: Skema Usulan, Takdir

Berdasarkan hasil pengujian, pola distribusi berbasis SOA usulan Takdir dkk terbukti mampu memberikan hasil yang lebih baik, dengan penggunaan *resource CPU* dan *memory* yang lebih rendah. Akan tetapi, lingkup perancangan dan pengujian sistem hanya terbatas pada perangkat komputer (*desktop* maupun *laptop*). Pada pola terdistribusi, usulan Takdir dkk, *client* perlu untuk melakukan replikasi data maupun *workflow* yang berupa *Web service* agar dapat berjalan. Sementara pada pendataan data berbasis CAPI, perangkat yang digunakan adalah *mobile device*, yang mempunyai arsitektur maupun *resource* yang berbeda jika dibandingkan dengan komputer. Proses replikasi data *workflow* bisa menjadi sesuatu yang *challenging* atau bahkan sulit dilakukan semenjak *mobile device* bukanlah sesuatu yang umum digunakan sebagai *service provider*. Oleh karena itu, penelitian ini akan berfokus kepada perancangan desain dan implementasi sistem terdistribusi yang dapat diimplementasikan pada perangkat *mobile*.

I.2 Rumusan Masalah

Berdasarkan uraian latar belakang permasalahan diatas, maka dapat dirumuskan suatu permasalahan penelitian yaitu bagaimana merancang desain dan implementasi *mobile proxy* untuk pengumpulan data lapangan berbasis *mobile*.

I.3 Tujuan Penelitian

Tujuan utama dari penelitian ini adalah untuk merancang desain dan implementasi *mobile proxy* untuk pengumpulan data lapangan berbasis *mobile*. Adapun tujuan khusus penelitian ini adalah :

- Merancang dan mengimplementasikan *mobile proxy* yang dapat digunakan pada *connected* maupun *disconnected environment*,

- Menganalisis kompatibilitas *mobile device* yang memenuhi spesifikasi ran-cangan *mobile proxy*,
- Melakukan ujicoba atas desain dan implementasi *mobile proxy* yang diran-cang dengan studi kasus pengumpulan data lapangan,
- Menganalisis dan mengevaluasi hasil ujicoba sistem.

I.4 Batasan Masalah

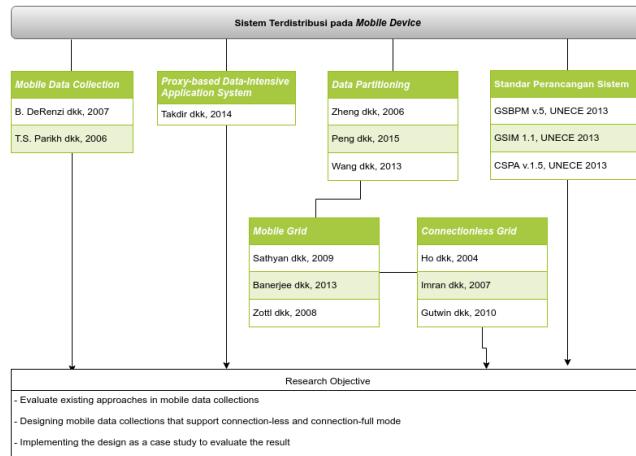
Batasan masalah dalam penelitian ini adalah :

- Penelitian ini hanya berfokus pada desain dan implementasi sistem pada *mo-bile device*,
- *Mobile device* yang digunakan dalam ujicoba terbatas hanya *mobile device* dengan sistem operasi *Android*,
- Perancangan GUI hanya digunakan sebagai bahan uji coba, dan bukan termasuk ke dalam poin inti penelitian.

Bab II

STUDI LITERATUR

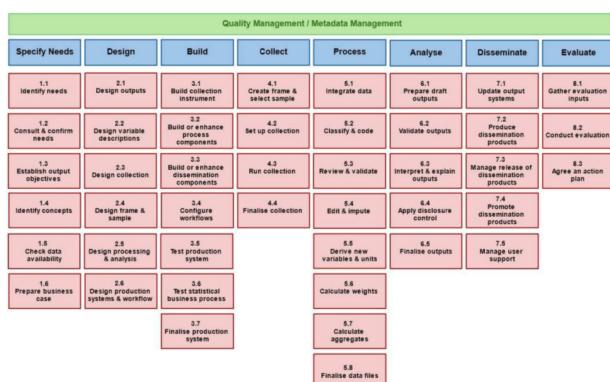
Untuk mendukung penelitian ini, telah dilakukan studi literatur terkait topik penelitian. Penelitian terkait tersebut direpresentasikan dalam peta literatur Gambar II.1



Gambar II.1: *Literature Map*

II.1 Generic Statistical Business Process Model (GSBPM)

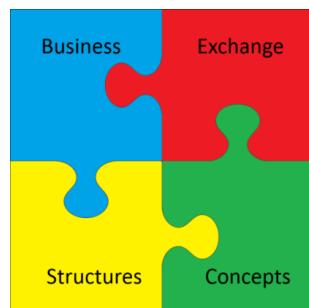
GSBPM [1] mendefinisikan sekumpulan bisnis proses yang dibutuhkan untuk untuk kegiatan perstatistikian. GSBPM ditujukan untuk membantu institusi perstatis-tikan di dunia melakukan modernisasi pada kegiatan perstatistikian yang dijalankan. BPS juga telah mencanangkan untuk mengikuti bisnis proses pada GSBPM melalui program *Statistical Capacity Building-Change and Reform for the Development of Statistics (STATCAP-CERDAS)*



Gambar II.2: *Statistical Business Process Phases* dalam GSBPM

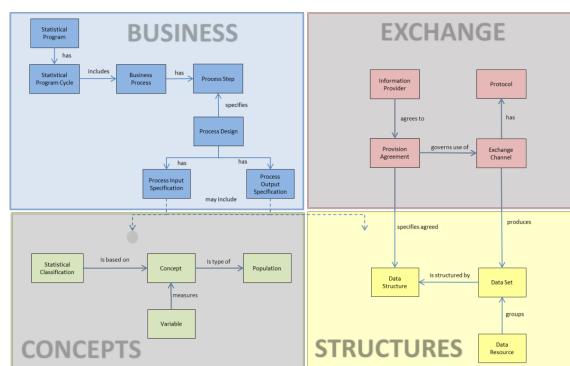
II.2 Generic Statistical Information Model (GSIM)

GSIM merupakan *framework model* informasi statistik internasional pertama yang diciptakan. GSBPM merupakan *reference framework* untuk information objects pada kegiatan perstatistikian yang mendefinisikan penjelasan, pengelolaan dan penggunaan data dan metadata [8]. GSIM dibentuk untuk memenuhi kebutuhan informasi pada proses-proses yang ada pada GSBPM. Pada level atas, GSIM mengelompokkan objek informasi menjadi 4 kategori, yakni *Business*, *Exchange*, *Concepts*, dan *Structures*.



Gambar II.3: Kategori Objek pada GSIM

Selain itu, GSIM juga telah menetapkan spesifikasi teknis struktur data untuk tiap-tiap objek. Terdapat sebanyak 114 entitas yang telah didefinisikan oleh GSIM.

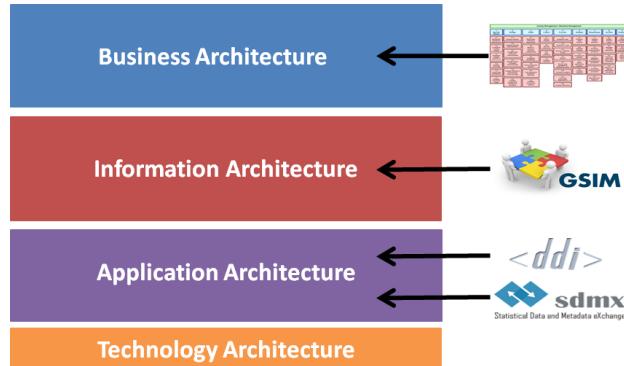


Gambar II.4: Gambaran Sederhana Relasi Objek pada GSIM

II.3 Common Statistical Production Architecture (CSPA)

CSPA merujuk pada GSBPM sebagai arsitektur bisnis, GSIM sebagai arsitektur informasi, serta menggunakan pendekatan SOA (*Service Oriented Architecture*) untuk arsitektur aplikasi dan arsitektur teknologi. Saat ini CSPA merupakan *project*

yang sedang aktif di UNECE. Project CSPA merupakan kolaborasi antara beberapa wakil di bidang TIK organisasi penyelenggara kegiatan perstatistikian dari berbagai negara guna merancang SOA untuk sistem perstatistikian [9]. Tahap awal pada CSPA adalah mengidentifikasi kandidat layanan statistik (*statistical services*). Hingga saat ini, CSPA telah menghasilkan beberapa kandidat *statistical services* yang nantinya akan distandarisasi.



Gambar II.5: Layer pada CSPA

II.4 Sistem Terdistribusi

Salah satu mekanisme untuk menangani masalah pada pendekatan centralized adalah dengan menggunakan pendekatan sebaliknya, yakni dengan menggunakan sistem terdistribusi. Sistem terdistribusi dapat diartikan sebagai sebuah jaringan komputer untuk menyelesaikan masalah komputasi (*computational problem*) dengan cara membagi *task* ke dalam beberapa komputer (Godfrey, 2006) [10]. Setiap komputer dalam jaringan berkomunikasi dengan menggunakan metode *message passing* [11]. Keuntungan dari pemanfaatan sistem terdistribusi antara lain [7]:

- Mempercepat *response time* dan mengurangi *traffic jaringan*,
- Meningkatkan kecepatan akses dan *query* terhadap data,
- Meningkatkan sistem *availability* dan menghindari kegagalan sistem dengan menerapkan *single point of failure*.

Selain keuntungan, sistem terdistribusi juga memiliki beberapa tantangan, antara lain :

- Pembagian data (*data partitioning*) dan proses komputasi (*task*),
- Keselarasan data (*data synchronization*).

II.5 Grid Computing

Grid computing dapat dianggap sebagai sebuah sistem terdistribusi yang bersifat *non-interactive* antar *node*. Masing-masing *node* pada *grid computers* menjalankan proses komputasi (*task*) secara terpisah. Prinsip umum yang menjadi karakteristik *grid* [12]:

- Tersusun dari beberapa domain administrasi dan otonomi (*geographically dispersed*),
- *Heterogen* dan tersusun dari beragam teknologi yang bervariasi,
- *Scalable*, dapat berkembang dari hanya beberapa *integrated resources* menjadi ribuan bahkan jutaan, *Dynamic* dan *adaptable*, mampu mendeteksi *resource failure* dan memanfaatkan *available resource*.

Untuk mengakomodir *heterogen resources*, maka *grid* harus mengikuti prinsip berikut :

- Tidak menginterfensi domain administrasi dan otonomi yang sudah ada,
- Tidak perlu mengganti sistem operasi, protokol jaringan, atau *services*,
- Memungkinkan *remote sites* untuk bergabung atau meninggalkan jaringan kapanpun,
- Menyediakan sistem yang handal dan infrastruktur yang *fault tolerance* dengan *no single point of failure*,
- Menyediakan *support* untuk komponen yang heterogen,
- Menggunakan teknologi yang telah ada dan terstandarisasi,
- Menyediakan sinkronisasi.

II.6 Mobile Grid Computing

Mobile grid computing dapat didefinisikan sebagai sebuah sistem terdistribusi berbasis *grid computing* yang mengakomodir penggunaan *mobile device*. *Mobile device*, seperti *PDA* dan *smartphone*, biasanya diindikasikan dengan kekuatan pemrosesan yang lemah, kapasitas baterai yang terbatas, dan media penyimpanan yang

terbatas. Menjalankan proses yang intensif pada *mobile device* memerlukan kemampuan pemrosesan yang lebih, media penyimpanan, penyelesaikan proses dengan waktu seminimal mungkin, sehingga penggunaan *mobile grid* dapat menjadi salah satu solusi [13]. *Mobile grid computing* memungkinkan mobility dalam sebuah *grid framework*, sehingga memungkinkan *mobile device* dapat beroperasi secara halus (*seamless*), transparan, aman dan efisien.

Zotll dkk telah meneliti sebuah metode untuk menyimpan informasi dalam tiga tingkatan (*Three-tier*), (i) *base tier*, (ii) *information tier*, dan (iii) *organization tier* yang dinamakan IRMA. Metode IRMA diterapkan pada *mobile grid environment* dengan menggunakan *Erasure Code* [14]. *Erasure Code* (EC) [15] menggunakan skema *replication* dan *parity coding*. Skema replikasi bekerja dengan menduplikasi data pada beberapa *node*, sementara skema *parity coding* tidak menduplikasi data, tetapi menggunakan *checksum* untuk *re-recover* dari *failure*. Pada *mobile computing*, replikasi adalah cara yang umum digunakan untuk menjamin ketersediaan (*persistence*). Beberapa strategi replikasi yang telah dikembangkan antara lain *uniform*, *proportional*, dan *square-root* [16]. Replikasi dapat menjamin *persistence* pada saat koneksi putus secara tiba-tiba (*unpredicted disconnected*). Berdasarkan hasil penelitian, metode IRMA dapat bekerja secara konsisten, persisten, dan efisien dalam jaringan *mobile grid* yang dinamis.

II.7 Connectionless Grid Computing

Pada sistem terdistribusi berbasis *mobile grid*, sebagian besar *node* berupa *mobile device*, seperti *PDA* dan *smartphone*. *Mobile device* mempunyai karakteristik yang *mobile*, dapat dengan mudah berpindah dari suatu lokasi ke lokasi yang lain. Lokasi dimana *mobile device* ini berada terkadang merupakan lokasi yang tidak terjangkau oleh sinyal telekomunikasi atau *wifi* area. Tidak terhubungnya suatu *mobile node* dalam sebuah jaringan dapat dikelompokkan dalam beberapa jenis [17]:

- *Delay-based Interruption*, merupakan sebuah gap singkat (*short-term gap*) dalam pengiriman pesan,
- *Network Outage*, merupakan kondisi dimana sebuah *mobile node* terputus dari jaringannya karena hilangnya jaringan komunikasi,
- *Explicit Departures*, merupakan kondisi *mobile node* keluar dari jaringan atau keluar dari aplikasi secara eksplisit

Ho dkk meneliti tentang penggunaan pendekatan *connection-less* untuk mengakomodir permasalahan komunikasi [18]. Pendekatan *connection-less* ini bekerja secara *ad-hoc* pada jaringan tanpa infrastruktur (*infrastructure-less network*), sehingga memungkinkan *mobile nodes* juga berperan sebagai *router* yang mengakomodir komunikasi dengan *mobile nodes* yang lain. Pendekatan *connection-less* juga diteliti oleh Imran dkk [19] dengan menggunakan *proxy-based checkpoint* dan *message logging* yang dinamakan *Mobile Host Proxies [MHP]*. Metode MHP menjadikan setiap *mobile device* sebagai *Mobile Host [MH]* dan *Mobile Service Station [MSS]* dan menyimpan *checkpoint* secara *asynchronous*.

II.8 Pola Implementasi SOA untuk Sistem Data-Intensif Terdistribusi

Service Oriented Architecture (SOA) dapat dipandang dari dua perspektif yang berbeda : sebagai arsitektur bisnis (business architecture) dan arsitektur teknis (technical architecture). Dari sudut pandang bisnis, SOA menawarkan solusi dalam mengelola bisnis dengan menekankan pada layanan (services) sebagai unit/satuan terkecil yang perlu dirancang dan diorganisir. Sementara dari sudut pandang teknis, SOA memperkenalkan prinsip-prinsip dalam merancang dan mengembangkan perangkat lunak yang berorientasi service.

Service dalam konteks bisnis merupakan kumpulan aktifitas yang dijalankan oleh tiap-tiap unit kerja dalam suatu perusahaan. Dengan demikian, sebuah unit kerja dapat berperan sebagai penyedia layanan (service provider) bagi unit kerja lain yang membutuhkan (service consumer), dan sebaliknya. Hubungan antar penyedia dan pengguna layanan diatur dalam sebuah perjanjian (contract), yang berisi penjelasan dari layanan serta syarat dan ketentuan yang harus dipenuhi kedua belah pihak.

Service dalam konteks teknis, mengadopsi konsep service yang diterapkan dalam konteks bisnis. Akan tetapi, service dalam konteks teknis diinterpretasikan dalam fungsi-fungsi yang didukung oleh software, atau dikenal dengan logika bisnis (business logic). Logic business dikemas dalam bentuk komponen software dengan mekanisme tertentu sehingga dapat didistribusikan.

Thomas Erl memperkenalkan prinsip-prinsip dalam merancang software service, yang bertujuan untuk memudahkan integrasi sistem informasi, yaitu:

- Standardize Service Contract

Services yang terdapat pada sistem inventarisasi yang sama memiliki desain kontrak yang serupa.

- Service Loose Coupling

Service tidak memiliki ketergantungan terhadap lingkungan atau platform tertentu.

- Service Abstraction

Service provider hanya menyediakan informasi dasar, sementara teknis implementasi service tidak disertakan dalam kontrak.

- Service Reusability

Service tidak dirancang untuk keperluan tunggal, tetapi bersifat multi-purpose. Sebuah service merupakan enterprise resource yang dapat digunakan kembali.

- Service Statelessness

Meminimalisir penggunaan resources dengan tidak menyertakan pengelolaan state pada service.

- Service Discoverability

Service dilengkapi dengan metadata, sehingga dapat ditelusuri secara efektif.

- Service Composability

Sebuah service dapat tersusun dari penggabungan beberapa service.

SOA hanyalah merupakan sebuah arsitektur, sehingga diperlukan teknologi pendukung agar dapat direalisasikan. Web service merupakan salah satu teknologi yang dapat digunakan untuk merealisasikan konsep SOA menjadi sebuah sistem software. Web service menggunakan teknologi XML yang didesain sedemikian rupa sehingga mendukung komunikasi antar komponen software yang bersifat platform independent (W3C 2004).

XML sendiri merupakan dokumen teks yang mengikuti format dan spesifikasi tertentu sehingga mudah untuk diproses oleh mesin/komputer. Tujuan yang ingin dicapai dari XML (W3C 2006) antara lain :

- XML harus dapat digunakan pada jaringan Internet,

- XML harus dapat diaplikasikan secara luas,

- Adanya kemudahan pembuatan program (software) untuk memproses dokument XML,

- Meminimalkan jumlah fitur yang tidak harus ada,
- Dokumen XML harus dapat terbaca dengan layak oleh manusia,
- Dokumen XML harus mudah dibuat.

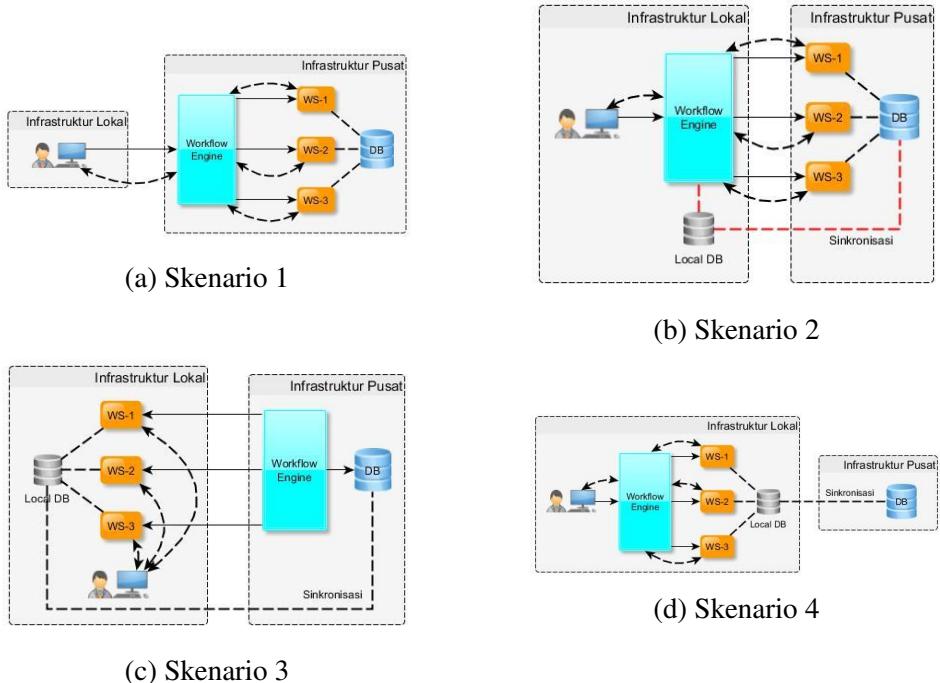
Teknologi Web service mengadopsi XML dalam format Web Service Definition Language (WSDL). WSDL menyediakan informasi kepada service consumer berupa struktur data yang diperlukan, jenis operasi yang didukung, data binding, protokol komunikasi data, maupun endpoint. SOAP merupakan protokol komunikasi yang ditujukan untuk menangani pertukaran data antara service provider dan service consumer. Selain mendefinisikan struktur data, SOAP juga menyertakan nilai dari setiap atribut.

Pada penelitiannya, Takdir dkk telah mencoba menemukan pola implementasi SOA pada aplikasi yang bersifat data intensif. Aplikasi data intensif memiliki ciri : (i) tersusun dari sederetan operasi yang mengolah data dalam jumlah besar, dan (ii) adanya aliran data dengan volume yang besar antar operasi-operasi tersebut (Habich, Richly, Preissler, et al. 2008). Pola implementasi SOA rancangan Takdir dkk bertujuan untuk menjembatani gap antara desain business services dan software service. Proses perancangan pola implementasi menitikberatkan pada layanan statistik (statistical services) yang bersifat data intensif.

Perancangan pola implementasi dibagi menjadi dua, implementasi logik workflow dan implementasi web services. Mekanisme eksekusi logik workflow dan web services, sebagaimana dalam Gambar II.6, dalam pengujian dibagi dalam 4 (empat) skenario :

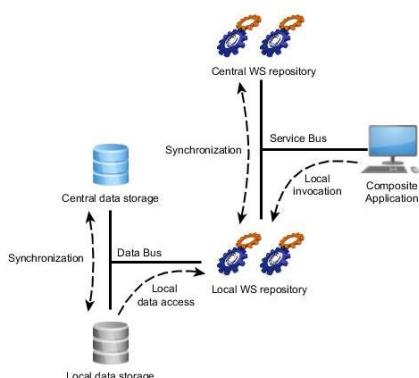
1. Pengeksekusian workflow dan web services dilakukan pada infrastruktur pusat,
2. Pengeksekusian workflow dilakukan pada infrastruktur lokal, sementara pengeksekusian web service dilakukan pada infrastruktur pusat,
3. Pengeksekusian workflow dilakukan pada infrastruktur pusat, dan web service pada infrastruktur lokal,
4. Pengeksekusian workflow dan web services dilakukan pada infrastruktur lokal.

Instilah infrastruktur pusat disini merujuk pada infrastruktur TIK yang ada di BPS Pusat, sementara istilah infrastruktur lokal merujuk pada infrastruktur TIK pada BPS Daerah.



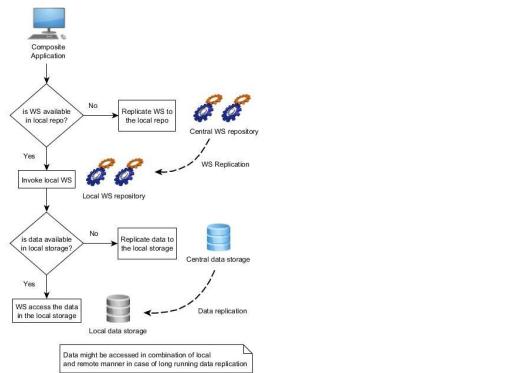
Gambar II.6: Skenario Eksekusi Logik Workflow dan Web Services

Pada skenario yang melibatkan infrastruktur pusat dan infrastruktur lokal, terdapat mekanisme sinkronisasi pada database. Selain itu, pada skenario yang menempatkan web services pada repository lokal juga diperlukan sinkronisasi web services, antara central repository dan local repository. Proses sinkronisasi, baik database maupun web services, antara infrastruktur pusat dan infrastruktur lokal dilakukan untuk menjaga konsistensi antara keduanya.



Gambar II.7: Sinkronisasi Data dan Web Service

Selain mekanisme sinkronisasi, pada skenario yang melibatkan infrastruktur pusat dan daerah juga terdapat mekanisme replikasi. Replikasi, baik database maupun web services dilakukan dengan menggunakan pendekatan cache. Tujuan replikasi adalah untuk mendekatkan data maupun web service kepada service consumer. Replikasi data sangat tergantung pada replikasi web service, karena hanya data yang dibutuhkan oleh web service yang akan direplika.

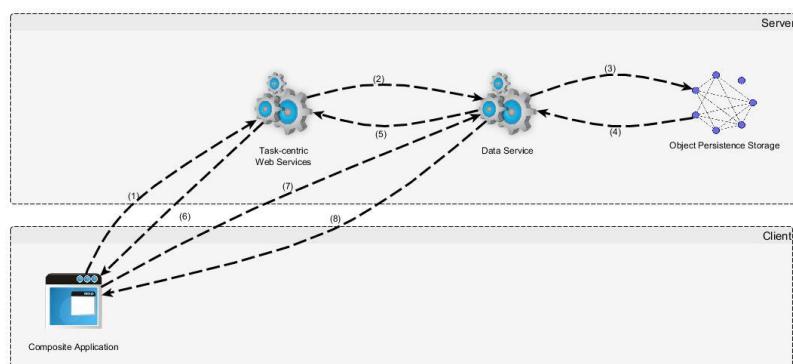


Gambar II.8: Proses Replikasi Data dan Web Service

Terdapat beberapa skenario deployment yang mengakomodir empat pendekatan eksekusi logik dan web service :

1. Web service dan data terpusat

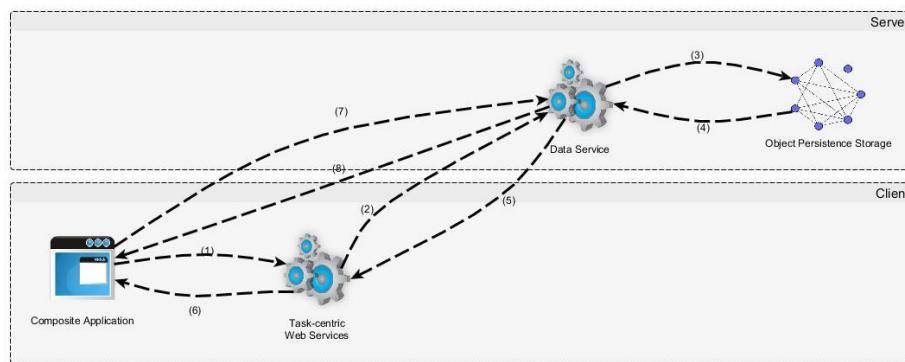
Pada skenario ini, web services (task-centric web service dan data service) dan database dipusatkan pada server seperti pada Gambar II.9 Skenario ini cocok untuk web service yang membutuhkan akses terhadap data yang bersifat global dan memiliki parameter dengan ukuran data yang kecil.



Gambar II.9: Skenario Deployment Web Service dan Data Terpusat

2. Web service terdistribusi, data terpusat

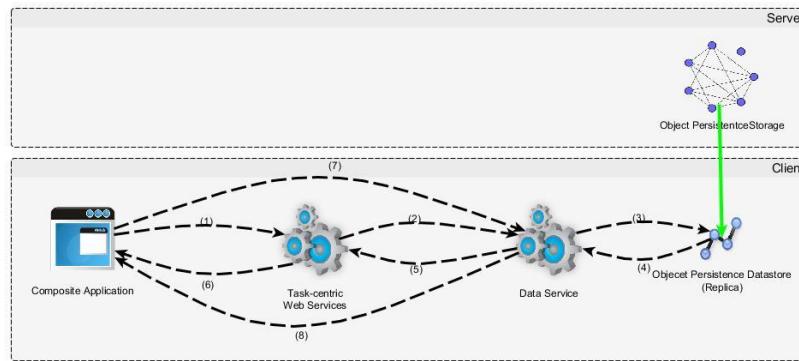
Logik task-centric web services direplika ke client untuk dieksekusi. Logik direpresentasikan dalam bentuk file *.class untuk kemudian di-load ke runtime dengan Java Reflection API . Setelah file yang berisi logik direplika, maka client akan menjalankan servlet engine pada Apache CXF untuk melakukan hosting web service. Dalam hal ini, client juga berperan sebagai server (service provider) bagi dirinya sendiri, maupun client lain yang terkoneksi dengannya.



Gambar II.10: Skenario Deployment Web Service Terdistribusi, Data Terpusat

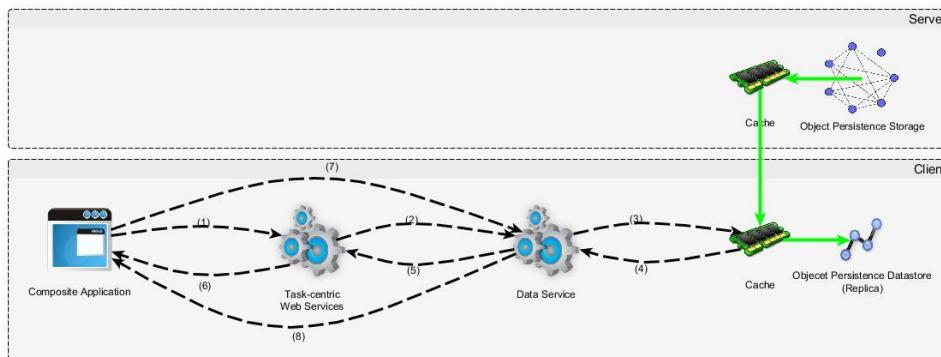
3. Web service dan data terdistribusi

Skenario ini ditujukan untuk memungkinkan sistem bekerja secara offline (disconnected) dimana semua komponen software yang dibutuhkan oleh client direplikasi, baik data, maupun web services. Komunikasi antarkomponen berjalan pada infrastruktur lokal, kecuali proses sinkronisasi data dengan server yang dilakukan secara asynchronous. DataObjek yang direplika ke object persistence datastore pada client adalah DataObject yang hanya dibutuhkan oleh task-centric web sevices yang telah direplika pada client. Setelah replikasi berlangsung, maka infrastruktur lokal dapat beroperasi meskipun kehilangan koneksi dengan infrastruktur pusat. Proses sinkronisasi data dapat dilakukan secara asynchronous ketika koneksi ke infrastruktur pusat tersedia kembali. Gambar II.11 merupakan gambaran skenario tersebut.



Gambar II.11: Skenario Deployment Web Service dan Data Terdistribusi

Pada saat replikasi data dari infrastruktur pusat ke infrastruktur lokal sedang berlangsung, teknologi In-Memory Data Grids (IMDG) digunakan agar operasi di client dapat tetap berjalan sementara proses replikasi berjalan secara asynchronous. Hal ini juga berlaku untuk proses sinkronisasi data dari infrastruktur lokal ke infrastruktur pusat, maupun sebaliknya.



Gambar II.12: Proses Replikasi Data dengan IMDG

Pada tahap pengujian, data yang digunakan adalah data Kor SUSENAS 2009, yang terdiri dari 129 fields. Adapun variabel yang diamati dalam mengevaluasi kinerja sistem adalah : 1) Response Time, 2) Network utilization, dan 3) CPU utilization. Pengujian yang dilakukan pada environment virtual machine, dengan konfigurasi yang telah disesuaikan dengan kondisi yang sebenarnya, diperoleh hasil sebagai berikut :

1. Skenario web service dan data terpusat

Tabel II.1: Response Time Pengujian Skenario I

Jumlah Records	Fully Centralized	Distributed
5	00:00:04.627	00:00:02.484
10	00:00:08.320	00:00:03.205
100	00:02:00.142	00:00:12.648
200	00:04:01.497	00:00:22.653
500	00:10:02.997	00:01:50.549

Tabel II.2: Total Traffic Size Pengujian Skenario I

Jumlah Records	Fully Centralized	Distributed
5	748	7.64
10	1382.4	7.53
100	20449.28	7.76
200	40908.8	7.64
500	102256.64	7.64

2. Skenario web service terdistribusi, data terpusat

Tabel II.3: Response Time Pengujian Skenario II

Jumlah Records	Fully Centralized	Distributed
5	00:00:14.470	00:00:09.530
10	00:00:14.872	00:00:10.745
100	00:01:15.759	00:00:23.183
200	00:02:22.225	00:00:37.554
500	00:05:39.980	00:02:43.895

Tabel II.4: Total Traffic Size Pengujian Skenario II

Jumlah Records	Fully Centralized	Distributed
5	365.62	449.88
10	692.19	934.07
100	10240	12400.64
200	20490.24	27084.8
500	51210.24	38840.32

3. Skenario web service dan data terdistribusi

(a) Response Time

Waktu yang dibutuhkan untuk melakukan updating web service (error correction service) adalah 00:00:04,647.

(b) Network Utilization

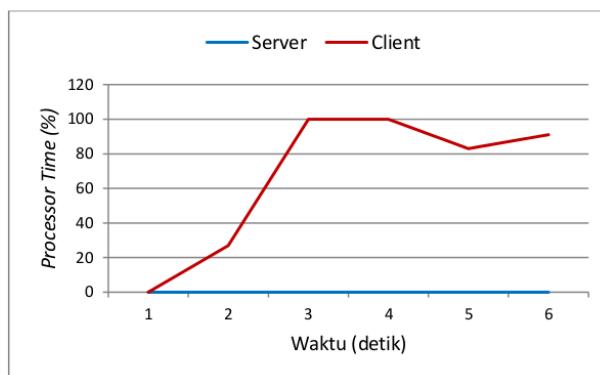
Total traffic yang dihasilkan sebesar 17.21 kilobytes. Ukuran tersebut merupakan file logik web service yang di-download oleh client.

Statistics Item	Current Value					
Traffic	Bytes	Packets	Utilization	bps	pps	
Total	17.21 KB	33	0.000%	0.000 bps	0	
Broadcast	110.00 B	2	0.000%	0.000 bps	0	
Multicast	1.75 KB	2	0.000%	0.000 bps	0	
Average Packet Size	534.000 Bytes					
Pkt Size Distribution	Bytes	Packets	Utilization	bps	pps	
<=64	911.00 B	15	0.000%	0.000 bps	0	
65-127	140.00 B	2	0.000%	0.000 bps	0	
128-255	243.00 B	1	0.000%	0.000 bps	0	
256-511	1.21 KB	4	0.000%	0.000 bps	0	
512-1023	1.75 KB	2	0.000%	0.000 bps	0	
1024-1517	1.13 KB	1	0.000%	0.000 bps	0	
>=1518	11.86 KB	8	0.000%	0.000 bps	0	

Gambar II.13: Network Utilization Proses Patching/Updating Web Service

(c) CPU Utilization

Processor time yang dihasilkan mesin client pada Gambar II.14 merupakan proses untuk melakukan hosting web service yang di-download dari server. Proses tersebut menggunakan sekitar 50 persen processor time dari masing-masing CPU core.

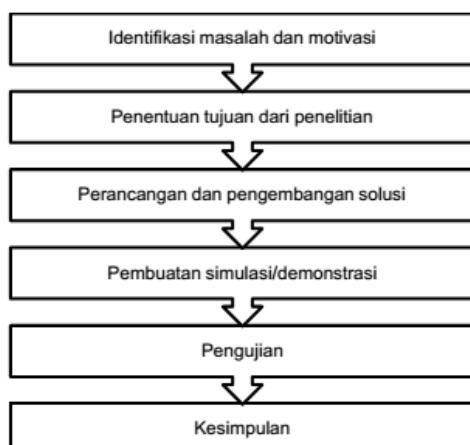


Gambar II.14: Grafik CPU Utilization Proses Updating Web Service

Bab III

METODOLOGI

Metodologi penelitian yang akan dilakukan di dalam penelitian ini adalah melalui pendekatan *Design Science Research Methodology for Information Systems Research* (Peffers, et al, 2008) [20]. Metodologi penelitian terdiri dari enam tahapan seperti digambarkan pada Gambar III.1 yaitu: identifikasi masalah dan motivasi, penentuan tujuan dari penelitian, perancangan dan pengembangan solusi/demonstrasi, pengujian dan pembahasan, pengambilan kesimpulan. Penelitian ini akan menggunakan strategi *sequential exploratory design* yang akan dilakukan dalam dua tahap, tahap pertama dilakukan secara kualitatif, dan tahap kedua secara kuantitatif. Hasil temuan kualitatif pada tahap pertama, akan dilanjutkan dengan analisis kuantitatif pada tahap kedua.



Gambar III.1: Tahapan *Design Science Research Methodology (DSRM)*

Pada tahap pertama atau kualitatif, akan dilakukan studi literatur yang terkait dengan permasalahan penelitian, kemudian akan dikombinasikan dengan hasil pemeriksaan dokumen. Hal ini dilakukan untuk mengimplementasikan konsep triangulasi yaitu perbedaan sumber data dan metode pengumpulan data. Pada tahap kedua atau tahap kuantitatif, akan dilakukan pengujian terhadap variabel-variabel penelitian untuk menjelaskan hubungan antara variabel-variabel tersebut melalui pengujian hipotesis menggunakan metode statistik. Metode yang digunakan dalam pengumpulan data adalah metode survei dengan menghimpun persepsi responden mengenai

variabel-variabel yang ditanyakan melalui instrumen berupa daftar pertanyaan berstruktur (kuesioner).

1. Identifikasi Masalah dan Motivasi

Pada tahapan ini melakukan identifikasi terhadap masalah yang ada, terutama berdasarkan teknologi dan kondisi yang ada. Identifikasi dilakukan dengan melakukan kajian untuk memahami dan menentukan motivasi berdasarkan hasil dari studi literatur. Studi dilakukan terhadap jurnal penelitian internasional, tesis, dan buku-buku teori pendukung nasional dan internasional. Peneliti melakukan analisis, interpretasi, dan generalisasi fakta-fakta dari literatur yang didapatkan. Studi juga dilakukan terhadap kondisi saat ini melalui pengumpulan data-data yang tersedia. Studi pustaka dilakukan untuk mendapatkan konsep dari fenomena yang terjadi. Dari fenomena dan konsep tersebut, kemudian dapat dirumuskan masalah, tujuan penelitian, batasan masalah dan pertanyaan penelitian.

2. Penentuan Fokus dari Penelitian

Penentuan fokus ditentukan berdasarkan hasil identifikasi masalah dan motivasi yang mendorong dilakukannya penelitian. Fokus penelitian adalah perancangan desain dan implementasi sistem terdistribusi pada perangkat mobile. Pembuatan proposal dilakukan sebagai pedoman dalam melakukan penelitian.

3. Perancangan dan Pengembangan Solusi

Perancangan dan pembuatan solusi berdasarkan fokus dari penelitian yaitu perancangan desain dan implementasi sistem terdistribusi pada perangkat mobile. Tahap pengembangan desain dilakukan dengan berbasis pada teori, yang terdiri dari tahap identifikasi masalah pada sistem terdistribusi berbasis perangkat mobile, kemudian perancangan desain sistem terdistribusi dan dilanjutkan dengan implementasi desain pada perangkat mobile. Tahap terakhir adalah pengujian desain serta model yang diteliti.

4. Pembuatan Simulasi / Demonstrasi

Berdasarkan rancangan solusi yang dibuat, pembuatan simulasi/demonstrasi dibangun dengan tujuan menguji rancangan solusi yang dibuat untuk melihat kesesuaian rancangan dengan harapan yang ingin dicapai.

5. Pengujian

Pengujian dilakukan untuk memastikan desain yang dirancang sesuai dengan kebutuhan sehingga dapat menyelesaikan permasalahan yang diangkat pada penelitian ini. Dalam penelitian, desain dan prototype diuji melalui beberapa skenario, antara lain : connection-full, connection-less, dan delay-based. Dalam pengujian, variable yang akan diukur antara lain : konsistensi, transparansi, waktu respon, dan trafik.

6. Analisis

Analisis dilakukan terhadap hasil pengujian yang didapatkan. Analisis bertujuan memberikan gambaran kondisi aplikasi dan masukan mengenai arah pengembangan lebih lanjut. Setelah data diolah, kemudian hasilnya dianalisis dan dibahas sesuai teori yang mendasari dan kriteria-kriteria pada model persamaan struktural. Hasil analisis harus dapat menjawab hipotesis yang ditentukan di awal penelitian. Selain itu, karena penelitian ini studi kasus, maka hasil dari perhitungan statistik dan berdasarkan teori maka dibuat pembahasan sesuai kontekstual tempat studi kasus. Dalam pembahasan diulas hasil penelitian ini dibandingkan dengan penelitian sebelumnya.

7. Kesimpulan

Kesimpulan berdasarkan hasil dari tahapan penelitian yang telah dilakukan sebelumnya yang merupakan jawaban dari permasalahan serta perwujudan dari tujuan yang dicapai dari penelitian. Diharapkan hasil penelitian dapat memberikan kontribusi dalam mengatasi permasalahan yang ada.

Daftar Pustaka

- [1] “GSBPM v5.0 - Generic Statistical Business Process Model - UNECE Statistics Wikis.”
- [2] “Redesigning a Questionnaire for Computer Assisted Data Collect - cp940020.pdf.”
- [3] G. Klein and M. G. Sobol, “Bias in computer-assisted surveys,” *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 26, pp. 566–571, Sept. 1996.
- [4] R. TerguEFF, J. Haajanen, J. Leppanen, and S. Toivonen, “Mobile SOA: Service Orientation on Lightweight Mobile Devices,” in *IEEE International Conference on Web Services, 2007. ICWS 2007*, pp. 1224–1225, July 2007.
- [5] B. DeRenzi, Y. Anokwa, T. Parikh, and G. Borriello, “Reliable Data Collection in Highly Disconnected Environments Using Mobile Phones,” in *Proceedings of the 2007 Workshop on Networked Systems for Developing Regions*, NSDR ’07, (New York, NY, USA), pp. 4:1–4:5, ACM, 2007.
- [6] T. S. Parikh and E. D. Lazowska, “Designing an Architecture for Delivering Mobile Information Services to the Rural Developing World,” in *Proceedings of the 15th International Conference on World Wide Web*, WWW ’06, (New York, NY, USA), pp. 791–800, ACM, 2006.
- [7] Takdir and A. I. Kistijantoro, “Multi-layer SOA implementation pattern with service and data proxies for distributed data-intensive application system,” in *2014 International Conference on ICT For Smart Society (ICISS)*, pp. 37–41, Sept. 2014.
- [8] “GSIM Specification - Generic Statistical Information Model - UNECE Statistics Wikis.”
- [9] “CSPA v1.5 - Common Statistical Production Architecture - UNECE Statistics Wikis.”
- [10] “A primer on distributed computing..”
- [11] G. R. Andrews, *Foundations of Multithreaded, Parallel, and Distributed*. Addison Wesley, Oct. 1999.
- [12] M. Baker, R. Buyya, and D. Laforenza, “Grids and Grid technologies for wide-area distributed computing,” *Software: Practice and Experience*, vol. 32, pp. 1437–1466, Dec. 2002.
- [13] J. Sathyan and M. Rijas, “Job Management in Mobile Grid Computing,” in *Proceedings of the 7th International Conference on Advances in Mobile Com-*

- puting and Multimedia*, MoMM '09, (New York, NY, USA), pp. 422–426, ACM, 2009.
- [14] J. Zottl, G. Haring, and G. Kotsis, “A Three-tier Information Management Architecture for Mobile Grid Environments,” in *Proceedings of the 6th International Conference on Advances in Mobile Computing and Multimedia*, MoMM '08, (New York, NY, USA), pp. 243–247, ACM, 2008.
 - [15] H. Weatherspoon and J. Kubiatowicz, “Erasure Coding Vs. Replication: A Quantitative Comparison,” in *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, IPTPS '01, (London, UK, UK), pp. 328–338, Springer-Verlag, 2002.
 - [16] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker, “Search and Replication in Unstructured Peer-to-peer Networks,” in *Proceedings of the 16th International Conference on Supercomputing*, ICS '02, (New York, NY, USA), pp. 84–95, ACM, 2002.
 - [17] C. Gutwin, T. N. Graham, C. Wolfe, N. Wong, and B. de Alwis, “Gone but Not Forgotten: Designing for Disconnection in Synchronous Groupware,” in *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*, CSCW '10, (New York, NY, USA), pp. 179–188, ACM, 2010.
 - [18] Y. H. Ho, A. H. Ho, K. A. Hua, and G. L. Hamza-Lup, “A connectionless approach to mobile ad hoc networks,” in *Ninth International Symposium on Computers and Communications, 2004. Proceedings. ISCC 2004*, vol. 1, pp. 188–195 Vol.1, June 2004.
 - [19] N. Imran, I. Rao, Y.-K. Lee, and S. Lee, “A Proxy-based Uncoordinated Checkpointing Scheme with Pessimistic Message Logging for Mobile Grid Systems,” in *Proceedings of the 16th International Symposium on High Performance Distributed Computing*, HPDC '07, (New York, NY, USA), pp. 237–238, ACM, 2007.
 - [20] G. L. Geerts, “A design science research methodology and its application to accounting information systems research,” *International Journal of Accounting Information Systems*, vol. 12, pp. 142–151, June 2011.

Lampiran 1