



**APLIKASI SISTEM PENGIRIMAN INFORMASI
DATA LOKASI**

**“APLIKASI TRACKING MENGGUNAKAN HANDPHONE
DAN PEMANTAUAN LOKASI MENGGUNAKAN WEBCAM”**

LAPORAN TUGAS AKHIR

**Disusun dan Diajukan Untuk Melengkapi Syarat – Syarat Yang Diperlukan
Guna Memeroleh Diploma Tiga Politeknik**

Oleh :

Firman Nugraha

NIM -----

**KONSENTRASI TELEMATIKA
PROGRAM STUDI TEKNIK TELEKOMUNIKASI
JURUSAN TEKNIK ELEKTRO**

POLITEKNIK NEGERI JAKARTA

DEPOK

2009

PERSETUJUAN PEMBIMBING

Tugas Akhir dengan dengan judul "Aplikasi Sistem Pengiriman Informasi Data Lokasi" pada program Diploma Tiga Politeknik, Konsentrasi Telematika, Program Studi Teknik Telekomunikasi, Jurusan Teknik Elektro, Politeknik Negeri Jakarta untuk diuji dalam sidang Tugas Akhir Periode III pada 6 Oktober 2009.

Depok, 26 September 2009

Disetujui Pembimbing

-----, Ssi., MT.
NIP -----

HALAMAN PENGESAHAN

Tugas Akhir dengan judul "Aplikasi Sistem Pengiriman Informasi Data Lokasi" pada program Diploma Tiga Politeknik, Konsentrasi Telematika, Program Studi Teknik Telekomunikasi, Jurusan Teknik Elektro, Politeknik Negeri Jakarta telah diuji dalam sidang Tugas Akhir Periode III pada Senin 6 Oktober 2009 dan dinyatakan **lulus**.

Depok, 26 Oktober 2009

Disahkan oleh
Jurusan Teknik Elektro Politeknik Negeri Jakarta
Ketua,

-----, ST., M.Kom.
NIP. -----

ABSTRAK

Tujuan dari sistem ini adalah membuat suatu metode pengiriman data suatu lokasi secara otomatis tanpa banyak interferensi dari pengguna. Identy adalah dengan menggunakan *webcam* dan kamera *handphone* sebagai perangkat pengambil gambar dan program mengirimkannya data tersebut secara terus menerus ke server. Data-data tersebut dikirimkan melalui jaringan internet untuk kemudian disimpan dalam 2 bentuk yaitu file sistem dan *database record*.

Komponen-komponen penyiaran data sistem dibuat dengan menggunakan Microsoft Visual C# dan JME untuk program di sisi pengguna dan skrip PHP untuk pemrosesan data di server. Pengaturan penyiaran data dilakukan hanya dengan tahap-tahap mudah dan singkat sehingga sisanya diatur oleh sistem untuk pemrosesan dan pengiriman data. Dengan komponen-komponen penyiaran yang ada pada sistem data dapat dikirimkan ke server secara terus menerus sejalan dengan aktifitas pengguna sehari-hari.

Kata kunci: penyiaran data lokasi, otomatisasi, penyedia data informasi lokasi

KATA PENGANTAR

Alhamdulillah Puji dan syukur penulis panjatkan kepada Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul “Aplikasi Sistem Pengiriman Informasi Data Lokasi” ini tepat pada waktunya. Laporan Tugas Akhir ini disusun sebagai persyaratan untuk mengikuti sidang Tugas Akhir program Diploma 3 jurusan Teknik Elektro konsentrasi Telematika di Politeknik Negeri Jakarta.

Dalam penyusunan laporan ini penulis banyak menemui kendala, namun berkat adanya bimbingan dari pembimbing maka akhirnya penyusunan laporan ini dapat terlaksana dengan baik dan tepat pada waktu yang telah ditentukan.

Penulis mengucapkan terima kasih kepada :

1. Mera Kartika Delimayanti, Ssi. MT. selaku pembimbing Tugas Akhir dari Politeknik Negeri Jakarta.
2. Keluarga penulis atas dukungan dan bantuannya selama penulis mengerjakan laporan
3. Rekan satu tim Tugas Akhir penulis, Putra Setia Utama, yang telah bekerja sama dengan penulis untuk membuat laporan ini
4. Pak Vincent Roy atas bantuannya dalam pemahaman design software
5. Teman-teman penulis terutama sesama Telematika yang telah mendukung penulis selama pembuatan laporan ini.

Dengan segala keterbatasan yang ada, penulis menyadari bahwa laporan ini masih banyak kekurangan dan jauh dari kata sempurna. Oleh karena itu saran ataupun kritik yang bersifat membangun sangat diharapkan oleh penulis. Dan pada akhirnya harapan penulis laporan ini dapat bermanfaat bagi penulis sendiri dan pembaca sehingga dapat menambah pengetahuan dan wawasan.

Bogor, 26 September 2009

Penulis

DAFTAR ISI

PERSETUJUAN PEMBIMBING.....	ii
HALAMAN PENGESAHAN	iii
ABSTRAK	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vi
DAFTAR ISTILAH	viii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN	xiii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Tujuan	2
1.3. Batasan Masalah	3
1.4. Metode Penyelesaian Masalah	3
BAB II TEORI DASAR SISTEM.....	5
2.1. Linux	5
2.2. Apache HTTP Server	5
2.3. Google Maps	6
2.4. MySQL	6
2.5. PHP	7
2.6. .NET Framework	9
2.7. Java	10
2.8. GPS	11
BAB III PERANCANGAN DAN REALISASI SISTEM.....	13
3.1. Perancangan Sistem	13
3.1.1. Deskripsi Sistem.....	13
3.1.2. Diagram Blok Sistem	14
3.1.3. Analisa Keperluan	15
3.1.4. Spesifikasi Data	17
3.1.5.1. Design Fisik Data	17
3.1.5.2. Design Skema XML	18
3.1.5.3. Detail Elemen Data di Database	20

3.1.5.4.	Detail File-File Multimedia	27
3.1.5.	Komponen PiCo	28
3.1.5.5.	Use Case Pico	29
3.1.5.6.	Activity Diagram Pico	29
3.1.6.	Komponen TrackOn	31
3.1.6.1.	Use Case TrackOn	32
3.1.6.2.	Activity Diagram TrackOn	32
3.2.	Realisasi Sistem	35
3.2.1.	Realisasi Komponen PiCo.....	35
3.2.2.	Realisasi Komponen TrackOn.....	39
BAB IV PENGUJIAN DAN ANALISIS DATA		44
4.1.	Pengujian Perangkat Lunak.....	44
4.1.1.	Deskripsi Pengujian.....	44
4.1.2.	Prosedur Pengujian.....	46
4.1.3.	Data Hasil Pengujian	49
4.2.	Analisis Data Hasil Pengujian.....	54
BAB V PENUTUP.....		57
5.1.	Kesimpulan	57
5.2.	Saran	57
DAFTAR PUSTAKA		58
BIOGRAFI PENULIS		
.....		ER
ROR! BOOKMARK NOT DEFINED.		
LAMPIRAN.....		L1

DAFTAR ISTILAH

LiveReport

Komponen aplikasi sistem pengiriman informasi data lokasi yang bertugas untuk melakukan streaming video dari *handphone*

Media Gateway

Komputer yang dipindah fungsikan menjadi pemroses dan pengirim data-data dari *webcam* ke server pada komponen PiCo

PiCo

Komponen aplikasi sistem pengiriman informasi data lokasi yang bertugas untuk pemantauan lokasi menggunakan *webcam*

TrackOn

Komponen aplikasi sistem pengiriman informasi data lokasi yang bertugas untuk mencatat trek perjalanan beserta dengan gambar-gambar yang diambil dari lingkungan yang dilewatinya selama perjalanan menggunakan *handphone* dan GPS

DAFTAR TABEL

Tabel 3.1.	Analisa Keperluan Sistem	15
Tabel 3.2.	Spesifikasi fungsi-fungsi pada PiCo	16
Tabel 3.3.	Spesifikasi fungsi-fungsi pada TrackOn	16
Tabel 3.4.	Spesifikasi fungsi-fungsi pada LiveReport	17
Tabel 3.5.	Elemen data “id” pada tabel “acc_user”	20
Tabel 3.6.	Elemen data “name” pada tabel “acc_user”	20
Tabel 3.7.	Elemen data “password” pada tabel “acc_user”	20
Tabel 3.8.	Elemen data “id” pada tabel “trk_data”	21
Tabel 3.9.	Elemen data “id_user” pada tabel “trk_data”	21
Tabel 3.10.	Elemen data “track” pada tabel “trk_data”	22
Tabel 3.11.	Elemen data “status” pada tabel “trk_data”	22
Tabel 3.12.	Elemen data “start_time” pada tabel “trk_data”	22
Tabel 3.13.	Elemen data “end_time” pada tabel “trk_data”	23
Tabel 3.14.	Elemen data “track” pada tabel “trk_data”	23
Tabel 3.15.	Elemen data “id” pada tabel “trk_gallery”	23
Tabel 3.16.	Elemen data “id_track” pada tabel “trk_gallery”	24
Tabel 3.17.	Elemen data “capture_time” pada tabel “trk_gallery”	24
Tabel 3.18.	Elemen data “position” pada tabel “trk_gallery”	24
Tabel 3.19.	Elemen data “altitude” pada tabel “trk_gallery”	25
Tabel 3.20.	Elemen data “speed” pada tabel “trk_gallery”	25
Tabel 3.21.	Elemen data “heading” pada tabel “trk_gallery”	25
Tabel 3.22.	Elemen data “id” pada tabel “pic_station”	26
Tabel 3.23.	Elemen data “id_user” pada tabel “pic_station”	26

Tabel 3.24. Elemen data “name” pada tabel “pic_station”	27
Tabel 3.25. Elemen data “location” pada tabel “pic_station”	27
Tabel 3.26. Tipe data NMEA yang diproses TrackOn.....	43
Tabel 4.1. Hasil pengujian TrackOn terhadap kondisi jaringan pada lokasi pertama	53
Tabel 4.2. Hasil pengujian TrackOn terhadap kondisi jaringan lokasi kedua..	53
Tabel 4.3. Hasil pengujian TrackOn terhadap kondisi jaringan lokasi ketiga .	53

DAFTAR GAMBAR

Gambar 3.1.	Diagram blok sistem.....	14
Gambar 3.2.	Diagram fisik data sistem.....	17
Gambar 3.3.	Diagram skema XML conf_cam.xml.....	18
Gambar 3.4.	Diagram skema XML archived.xml.....	19
Gambar 3.5.	Diagram skema XML live.xml.....	19
Gambar 3.6.	Diagram alur kerja PiCo.....	28
Gambar 3.7.	Use Case PiCo.....	29
Gambar 3.8.	Activity diagram PiCo.....	29
Gambar 3.9.	Activity diagram pengecekan konfigurasi PiCo.....	30
Gambar 3.10.	Activity Diagram Penyiaran Webcam PiCo.....	30
Gambar 3.11.	Diagram Alur Kerja TrackOn.....	31
Gambar 3.12.	Use Case TrackOn.....	32
Gambar 3.13.	Activitiy diagram login pada TrackOn.....	32
Gambar 3.14.	Activity diagram pemilihan GPS TrackOn.....	33
Gambar 3.15.	Activity Diagram Penyiaran Trek TrackOn.....	34
Gambar 3.16.	Form Login PiCo.....	35
Gambar 3.17.	Form penamaan Media Gateway dan pengisian koordinat.....	36
Gambar 3.18.	Form pengaturan penyiaran webcam PiCo.....	37
Gambar 3.19.	Form konfigurasi webcam PiCo.....	38
Gambar 3.20.	Antarmuka login pada TrackOn.....	39
Gambar 3.21.	Antarmuka pemilihan perangkat GPS pada TrackOn.....	40
Gambar 3.22.	Antarmuka penampilan informasi geografis pada TrackOn ...	41
Gambar 3.23.	Antarmuka penampilan informasi geografis pada TrackOn ...	42

Gambar 4.1.	Peralatan pengujian komponen PiCo	44
Gambar 4.2.	Peralatan pengujian komponen TrackOn.....	45
Gambar 4.3.	Pemasukan data pengguna pada form login PiCo.....	46
Gambar 4.4.	Pemasukan nama Media Gateway pada form pembuatan Media Gateway PiCo	46
Gambar 4.5.	Pemasukan lokasi Media Gateway pada form lokasi Media Gateway PiCo	46
Gambar 4.6.	Pemasukan data pengguna pada form login TrackOn.....	47
Gambar 4.7.	Pemilihan perangkat GPS pada form daftar perangkat TrackOn	47
Gambar 4.8.	Pengisian nama trek pada form pembuatan trek TrackOn	48
Gambar 4.9.	Pembuatan file konfigurasi Media Gateway pada PiCo	49
Gambar 4.10.	Data Media Gateway baru pada PiCo	49
Gambar 4.11.	Pendaftaran webcam-webcam pada PiCo	50
Gambar 4.12.	Hasil penyiaran webcam oleh PiCo pada website demo.....	50
Gambar 4.13.	File-file PiCo di server	51
Gambar 4.14.	File conf_cam.xml Media Gateway PiCo di server	51
Gambar 4.15.	Antarmuka penampilan informasi geografis pada TrackOn	52
Gambar 4.16.	Hasil pengambilan gambar selama perjalanan.....	52
Gambar 4.17.	Antarmuka penampilan informasi geografis pada TrackOn	55

DAFTAR LAMPIRAN

1. PiCo - Mainframe.cs
LError! Bookmark not defined.
2. PiCo - create_gateway.php.....
LError! Bookmark not defined.
3. TrackOn - Tracker.java
LError! Bookmark not defined.
4. TrackOn - create_track.php
LError! Bookmark not defined.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Meningkatnya arus informasi yang terjadi pada saat ini membuat manusia sebagai penikmat informasi semakin ingin mencari informasi terbaru dalam bidang teknologi informasi. Banyak sekali penyedia konten lokal maupun mancanegara yang berlomba-lomba untuk menciptakan inovasi terbaru dalam penyediaan informasi berbasis multimedia dikarenakan multimedia dapat mendeskripsikan sebuah informasi secara menarik dan tepat sasaran.

Terkadang dengan semakin banyaknya penyedia informasi yang ada seringkali terjadi beberapa *publisher* memiliki info yang relatif sama baik dari isi maupun *layout*-nya. Kemudian info yang diberikan biasanya di dapat setelah ada laporan dari suatu kejadian ataupun acara serta informasi yang didapat juga struktural. Pengambilan informasi yang telah disebutkan berupa gambar maupun teks sering memiliki hambatan contohnya seperti wartawan yang harus siaga 24 jam untuk mengawasi suatu *event* atau kejadian lainnya. Hal-hal seperti ini membuat kebutuhan akan melihat dan memperlihatkan informasi secara fleksibel, efektif, dan efisien tanpa harus mengeluarkan biaya yang besar dan memakan waktu lama semakin meningkat.

Untuk menyajikan suatu informasi secara langsung melalui media internet, metode *live streaming* sering digunakan. Metode *live streaming* merupakan metode yang memungkinkan pengguna dapat melihat informasi yang sama pada saat yang relatif sama ketika informasi dikirimkan oleh orang lain melalui perangkat seperti mikrofon ataupun kamera. Metode ini sangat membantu untuk penyampaian informasi secara otomatis agar orang lain dapat mengetahui suatu informasi terkini walaupun berada di lain tempat.

Salah satu contoh penggunaan *live streaming* adalah untuk memonitor kepadatan jalan raya atau kondisi suatu lokasi. Dengan *live streaming* tidak lagi diperlukan orang yang selalu sedia setiap saat. Namun begitu, tidak dapat dipungkiri bahwa teknologi *live streaming* masih sangat mahal dalam pengimplementasiannya karena memerlukan perangkat keras yang mahal dan juga proses pemasangan yang tidak mudah. Tetapi

disamping semua itu konten lokal belum cukup menarik dan dari segi kuantitas sehingga konten yang ada terkesan selalu sama bagi semua kalangan.

Selain penyajian informasi secara langsung melalui *live streaming*, pada masyarakat saat ini juga telah terbentuk sebuah tren baru yang membuat masyarakat ingin menyampaikan segala macam informasi yang ditemui ke khalayak umum dengan cepat. Menjamurnya situs-situs web yang bersifat *social community* dengan peminatnya yang sangat banyak telah membuktikan hal tersebut. Contohnya, pada saat seseorang ingin bepergian dengan menggunakan sepeda atau motor ketika jalan-jalan, orang tersebut ingin memperlihatkan hal apa saja yang dilihat ke orang lain. Saat ini jika seseorang ingin mengabadikan apa yang dilihat ketika bersepeda atau menggunakan kendaraan lain, akan repot jika harus berhenti terlebih dahulu, lalu memotonya, lalu mencari tempat yang dapat digunakan untuk mengirimkan gambar yang didapat ke banyak orang. Dengan melakukan otomatisasi pada proses tersebut maka dengan begitu konten akan menjadi lebih banyak dan menarik serta berkualitas karena pengambilan gambar secara otomatis bisa saja menangkap momen-momen langka yang bisa berguna dan juga menghibur. Akan sangat membantu para *digital social networkers* jika ada suatu sistem yang dapat mempublikasikan hal-hal yang ditemui secara otomatis tanpa mengorbankan mobilitas.

Dengan latar belakang 2 poin yang telah dijabarkan di atas, dibuatlah inovasi dimana *live streaming* dan publikasi informasi secara otomatis dan *mobile* dapat digunakan oleh berbagai kalangan secara mudah. Inilah poin-poin yang dijadikan dalam dunia nyata di Indonesia pada khususnya untuk membuat suatu sistem pengiriman informasi otomatis yang dapat bekerja pada objek yang dinamis maupun statis. Semakin banyak nya kegiatan yang dilakukan manusia saat ini serta beragamnya peristiwa yang muncul tiap saat mendorong manusia mencari media yang dapat menangkap momen-momen dari banyaknya aktifitas manusia.

1.2. Tujuan

Secara umum, tujuan sistem ini dibuat adalah untuk memberikan suatu medium kepada para pengguna teknologi yang membutuhkan penyebaran dan penyajian data secara cepat dan efisien sehingga informasi dapat diakses dalam waktu yang singkat. Serta memberi sebuah bentuk konten lokal yang baru yang memungkinkan lebih banyak informasi yang didapat dari sistem ini melalui fitur-fiturnya yang dapat mengirimkan data

secara otomatis sehingga pengguna tidak perlu lagi repot-repot memasukkan data secara manual.

Secara khusus tujuannya adalah:

- Menciptakan sistem yang dapat mengambil data secara otomatis agar tidak mengganggu rutinitas dari para pengguna sistem ini. Dengan kata lain akan tercipta sistem yang praktis untuk digunakan.
- Menciptakan metode dan sistem yang efisien serta optimal untuk mengirimkan gambaran suatu lokasi dari pemberi masukan ke pusat data agar dapat digunakan oleh orang lain dengan media keluaran berupa tulisan dan gambar serta media pengantar internet
- Menciptakan sistem yang bersifat murah tetapi berdaya guna tinggi serta memiliki kuantitas dan kualitas yang baik dari segi teknis maupun segi konten

1.3. Batasan Masalah

- Sistem hanya sebagai pemasok data secara otomatis dan penyedia data
- Sistem tidak menyediakan tampilan untuk *end user*, tetapi hanya menjadi penyedia data yang dapat dipakai secara fleksibel oleh orang lain
- Sistem bekerja dengan menggunakan *webcam* serta *handphone* untuk alat masukan data utama
- Tipe data utama yang dihasilkan sistem berupa gambar, video, dan teks
- Sistem menggunakan teknik kompresi yang sudah ada dan mencari teknik kompresi yang paling optimal untuk sistem dari sekian banyak teknik kompresi data
- Sistem menggunakan jaringan internet sebagai medium pengiriman data

1.4. Metode Penyelesaian Masalah

1. Perancangan Sistem

Perancangan sistem yang dibuat meliputi perancangan data keluaran sistem agar dapat digunakan oleh orang lain dengan mudah dan fleksibel, perancangan alur pengiriman data, serta perancangan aplikasi dari masing-masing komponen sistem.

2. Realisasi

Pada tahap realisasi sistem ini rancangan yang telah dibuat sebelumnya direalisasikan. Eclipse Pulsar digunakan untuk pengembangan aplikasi mobile berbasis JME dengan

fiturnya yang mempermudah pembuatan file instalasi ke *handphone*. Adobe Dreamweaver digunakan untuk pembuatan skrip php agar penulisan sintaks PHP dipermudah. Microsoft Visual C# 2007 Express Edition digunakan untuk pembuatan aplikasi desktop agar dapat mengakses hardware-hardware yang terpasang pada komputer dengan mudah. Untuk basis data digunakan MySQL karena gratis dan umum digunakan serta mendukung data berbasis geospasial. Beberapa *library* juga digunakan untuk mempermudah pengembangan aplikasi.

3. Konsultasi dengan Dosen Pembimbing

Melakukan konsultasi dengan dosen pembimbing untuk mencari kekurangan-kekurangan yang ada pada sistem yang dibuat.

4. Pengujian lapangan dan revisi

Pengujian dilakukan untuk melihat apakah sistem yang telah dibuat dapat berjalan dengan baik sekaligus mencari-cari kesalahan yang ada pada sistem agar dapat diperbaiki.

5. Penulisan Laporan Tugas Akhir

Metode ini termasuk kedalam syarat yang harus dilampirkan pada saat pengajuan dan daftar sidang tugas akhir.

BAB II

TEORI DASAR SISTEM

2.1. Linux

Linux adalah duplikat Unix, *kernel*-nya ditulis oleh Linus Torvalds dan dikembangkan dengan bantuan *programmer* dan *hacker* dari seluruh dunia. Linux memiliki semua fitur yang dimiliki oleh Unix, termasuk *multitasking*, *virtual memory*, *shared libraries*, *demand loading*, *shared copy-on-write executables*, *proper memory management* dan *TCP/IP networking*.

Dengan fitur sekelas sistem operasi komersial tersebut tidak membuat Linux menjadi mahal harganya, justru Linux dapat diperoleh secara gratis. Walaupun ada sedikit biaya itu hanya sebagai ongkos distribusi atau pembelian cd saja.

Linux didistribusikan dibawah *GNU General Public License* yaitu suatu lisensi dimana pemilik program tetap memegang hak ciptanya tetapi orang lain dimungkinkan menyebarkan, memodifikasi atau bahkan menjual kembali program tersebut tapi dengan syarat *source code* asli harus tetap disertakan dalam distribusinya.¹

2.2. Apache HTTP Server

Apache HTTP Server, biasa disebut sebagai Apache, adalah sebuah *web server* terkemuka yang memiliki peranan penting dalam awal perkembangan *World Wide Web*. Pada tahun 2009, Apache menjadi *web server* pertama yang melebihi tonggak 100 juta situs. Apache adalah alternatif pertama dari *web server* Netscape Communications Corporation (saat ini dikenal sebagai Sun Java System Web Server), dan telah berkembang dan bersaing dengan *web server* berbasis Unix lainnya dalam hal fungsi dan performa. Mayoritas pengguna *web server* Apache menjalankan OS Linux.

Apache dikembangkan dan dikelola oleh sebuah komunitas pengembang di bawah naungan dari Apache Software Foundation. Aplikasi Apache tersedia untuk berbagai jenis sistem operasi, termasuk Unix, GNU, FreeBSD, Linux, Solaris, Novell NetWare, Mac OS

¹ Katriena, Flori. 5 September 2009, Apa itu Linux?, <http://pemula.linux.or.id/intro/apa.html>

X, Microsoft Windows, OS / 2, TPF, dan eComStation. Dirilis di bawah Lisensi Apache, Apache mempunyai karakteristik sebagai software yang bebas dan perangkat *open source*.

Sejak April 1996 Apache menjadi *server* HTTP paling populer di World Wide Web. Pada Maret 2009 Apache melayani lebih dari 46% dari semua situs di dunia dan lebih dari 66% dari jutaan tersebut adalah web tersibuk.²

2.3. Google Maps

Google Maps (dulu disebut Google Local) adalah sebuah aplikasi web layanan pemetaan dan teknologi yang disediakan oleh Google secara gratis (untuk penggunaan non-komersial), yang menjadi tonggak dari banyak web berbasis layanan peta, termasuk situs Google Maps, Google Ride Finder, Google Transit, dan peta-peta yang terdapat pada situs pihak ketiga melalui Google Maps API. Google Maps menawarkan peta jalan, sebuah rute perjalanan untuk perencanaan menggunakan jalan kaki, mobil, atau angkutan umum dan pencari tempat bisnis untuk berbagai negara di seluruh dunia. Menurut salah satu pembuat Google Maps (Lars Rasmussen), Google Maps adalah "salah satu cara untuk mengorganisir informasi dunia secara geografis".

Google Maps menggunakan proyeksi Mercator, sehingga tidak dapat menunjukkan daerah-daerah di sekitar kutub. Produk-produk yang berhubungan dengan Google Maps adalah Google Earth, sebuah program yang berdiri sendiri untuk Microsoft Windows, Mac OS X, Linux, SymbianOS, iPhone dan OS yang menawarkan fitur untuk melihat bumi, termasuk menunjukkan daerah kutub.³

2.4. MySQL

MySQL adalah sistem manajemen *database* SQL yang *open source* paling populer, yang dikembangkan, didistribusikan, dan didukung oleh MySQL AB. MySQL AB adalah perusahaan komersil, yang didirikan oleh pengembang MySQL. Berikut adalah fondasi utama MySQL:

- MySQL adalah sistem manajemen *database*
- MySQL adalah sistem manajemen *database* relasional
- Software MySQL adalah *open source*
- *Server database* MySQL sangat cepat, dapat dipercaya, dan mudah untuk digunakan

² Wikipedia. 5 September 2009, Apache HTTP Server, http://en.wikipedia.org/wiki/Apache_HTTP_Server

³ Wikipedia. 5 September 2009, Google Maps, http://en.wikipedia.org/wiki/Google_Maps

- *Server* MySQL bekerja dalam sistem terintegrasi atau *client/server*
- *Server* MySQL digunakan oleh banyak aplikasi

MySQL Spatial

MySQL juga mendukung *database* untuk menyimpan data spasial. MySQL mengimplementasikan ekstensi spasial dengan menggunakan spesifikasi dari Open Geospatial Consortium (OGC). Spesifikasi yang digunakan dari OGC adalah subset dari lingkungan SQL dengan tipe-tipe geometri. Kolom SQL yang berisi data geometri diimplementasikan sebagai kolom yang memiliki tipe data geometri. Spesifikasi tersebut menjelaskan satu set tipe-tipe geometri dan juga fungsi-fungsi dari tipe-tipe tersebut untuk membuat dan menganalisa data geometri.

Tipe data spasial yang didukung oleh MySQL Spatial adalah sebagai berikut:

- *Geometry*
- *Point*
- *Curve*
- *LineString*
- *Surface*
- *Polygon*
- *Geometry Collection*
- *MultiPoint*
- *MultiCurve*
- *MultiLineString*
- *MultiSurface*
- *MultiPolygon*⁴

2.5. PHP

PHP singkatan dari PHP Hypertext Preprocessor yaitu skrip pemrograman web yang bersifat *open source*. PHP merupakan skrip yang menyatu dengan HTML dan berada pada *server* (*server side HTML embedded scripting*). PHP adalah skrip yang digunakan untuk membuat halaman web yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan

⁴ MySQL AB. 5 September 2009, MySQL 5.1 Reference Manual, <http://dev.mysql.com/doc/refman/5.1/en/>

informasi yang diterima *client* selalu yang terbaru. Semua skrip PHP dieksekusi pada *server* dimana skrip tersebut dijalankan

Sejarah PHP

Dilihat dari perkembangannya, bahasa pemrograman ini memiliki perkembangan yang sangat cepat dengan jumlah pemakai yang terus bertambah.

1. PHP/FI

Ini merupakan cikal bakal PHP yang sekarang. Pertama dibuat oleh Rasmus Lerdorf pada 1995, pada awalnya menamakan skrip ini dinamakan “Personal Home Page Tool” yang merupakan bahasa sederhana dari bahasa pemrograman C dimana Personal Home Page Tool ini dapat berkomunikasi dengan *database* dan bersifat *open source*. Pada awalnya Rasmus membuat bahasa pemrograman ini bertujuan untuk menyimpan data pengunjung yang melihat biodata pada situs webnya.

Perkembangannya, pada pertengahan tahun 1997 pemakai bahasa PHP semakin banyak, terlihat dari jumlah statistik domain yang menggunakan PHP hampir lebih dari 50.000 situs web. Kemudian karena perkembangannya yang sangat pesat, Rasmus mengembangkan bahasa pemrograman ini, dan pada bulan November tahun 1997 muncul PHP/FI versi 2.0 yang merupakan cikal bakal PHP 3.

2. PHP Versi 3

PHP Versi 3 merupakan versi penyempurna dari bugs-bugs pada PHP/FI versi 1.0 dan PHP/FI versi 2.0. PHP Versi 3 ini dikembangkan oleh Andi Gutmans and Zeev Suraski pada tahun 1997 yang berhasil ditulis secara sempurna pada waktu itu. Fasilitas tambahan PHP Versi 3 dibandingkan versi sebelumnya, selain tambahan fungsi-fungsi baru, juga mendukung beberapa akses ke banyak *database*, pengelolaan protokol, dan API. Dari versi 3 lah singkatan PHP muncul, yaitu PHP: Hypertext Preprocessor, dan pada tahun 1998 hampir 10% situs web di dunia menggunakan PHP sebagai *web server*-nya.

3. PHP Versi 4

Pada musim dingin di tahun 1998, menulis ulang bahasa pemrograman PHP ini untuk membuat ketangguhan bahasa pemrograman ini. Akhirnya pada pertengahan tahun 1999 diperkenalkanlah PHP versi 4.0 yang menggunakan skrip *engine* Zend untuk meningkatkan penampilan (*performa*) dan mempunyai dukungan yang sangat banyak terhadap ekstensi dan berbagai *library* beserta modul. PHP versi 4.0 ini juga mempunyai keunggulan dibandingkan versi-versi sebelumnya, diantaranya

mendukung ke beberapa *web server*, fasilitas HTTP session, output buffer dan sistem keamanan. Pada perkembangannya, pada saat itu hampir 20% *web server* menggunakan bahasa pemrograman PHP sebagai *interpreter*-nya.

4. PHP Versi 5

Pada bulan Juli tahun 2005 muncul PHP versi 5.0 yang menggunakan Zend Engine 2.0 dengan penambahan beberapa fitur dan beberapa objek baru. PHP Versi 5 ini sangat mendukung pemrograman berbasis Object Oriented Programming alias OOP dan memang di peruntukan untuk OOP.⁵

2.6. .NET Framework

Microsoft .NET Framework (dibaca *Microsoft Dot Net Framework*) adalah sebuah komponen yang dapat ditambahkan ke sistem operasi Microsoft Windows atau telah terintegrasi ke dalam Windows (mulai dari Windows Server 2003 dan versi-versi Windows terbaru). Kerangka kerja ini menyediakan sejumlah besar solusi-solusi program untuk memenuhi kebutuhan-kebutuhan umum suatu program baru, dan mengatur eksekusi program-program yang ditulis secara khusus untuk framework ini. .NET Framework adalah kunci penawaran utama dari Microsoft, dan dimaksudkan untuk digunakan oleh sebagian besar aplikasi-aplikasi baru yang dibuat untuk platform Windows.

Pada dasarnya, .NET Framework memiliki 2 komponen utama: CLR dan .NET Framework Class Library.

Program - program yang ditulis untuk .NET Framework dijalankan pada suatu lingkungan software yang mengatur persyaratan-persyaratan runtime program. *Runtime environment* ini, yang juga merupakan suatu bagian dari .NET Framework, dikenal sebagai *Common Language Runtime* (CLR). CLR menyediakan penampilan dari *application virtual machine*, sehingga para *programmer* tidak perlu mengetahui kemampuan CPU tertentu yang akan menjalankan program. CLR juga menyediakan layanan-layanan penting lainnya seperti jaminan keamanan, pengaturan memori, *garbage collection* dan *exception handling* / penanganan kesalahan pada saat *runtime*. *Class library* dan CLR ini merupakan komponen inti dari .NET Framework. Kerangka kerja itu pun dibuat sedemikian rupa agar para *programmer* dapat mengembangkan program komputer

⁵ Supono. 5 September 2009, Apa Itu PHP?, <http://supono.wordpress.com/2006/09/14/apa-itu-php/>

dengan jauh lebih mudah, dan juga untuk mengurangi kerawanan aplikasi dan juga komputer dari beberapa ancaman keamanan.

CLR adalah turunan dari CLI (*Common Language Infrastructure*) yang saat ini merupakan standar ECMA. Untuk keterangan lebih lanjut, silakan mengunjungi situs ECMA atau kunjungi sumber pranala di bawah artikel ini.

Solusi-solusi program pembentuk *class library* dari .NET Framework mengcover area yang luas dari kebutuhan program pada bidang *user interface*, pengaksesan data, koneksi basis data, kriptografi, pembuatan aplikasi berbasis web, algoritma numerik, dan komunikasi jaringan. Fungsi-fungsi yang ada dalam *class library* dapat digabungkan oleh *programmer* dengan kodenya sendiri untuk membuat suatu program aplikasi baru.

Pada berbagai literatur dan referensi di Internet, .NET Framework seringkali disingkat menjadi .NET saja.⁶

2.7. Java

Java merupakan suatu teknologi perangkat lunak yang di dalamnya mencakup bahasa pemrograman. Selain itu Java juga merupakan suatu platform yang memiliki *virtual machine* dan *library* yang diperlukan untuk menulis dan menjalankan suatu program. Pada awalnya Java dikembangkan pada lingkungan komputer oleh Sun Microsystems Inc. dengan tujuan untuk menghasilkan suatu bahasa komputer sederhana tanpa harus terikat pada arsitektur tertentu.

Pada tahun 1995 Sun meluncurkan sebuah *browser* berbasis Java dengan julukan Hot Java, kemudian diikuti Netscape yang memutuskan untuk membuat *browser* dengan dilengkapi bahasa Java. Setelahnya, ikut bergabung pula berbagai pengembang ternama diantaranya IBM dan Microsoft. Pada tahun berikutnya, Sun Microsystems Inc. merilis Java Software Development Kit (JDK) pertamanya, yaitu JDK 1.1. Kemudian terus berkembang dari pemrograman *applet* yang berjalan di *browser* menjadi pemrograman kelas dunia yang banyak digunakan untuk pengembangan aneka ragam aplikasi.

Bahasa pemrograman Java sendiri secara garis besar dikelompokkan menjadi 3, yaitu Java 2 Standar Edition (J2SE), Java 2 Enterprise Edition (J2EE) dan Java 2 Micro Edition (J2ME). J2SE digunakan untuk mengembangkan aplikasi-aplikasi *desktop* dan

⁶ Wikipedia. 5 September 2009, .NET Framework, http://id.wikipedia.org/wiki/.NET_Framework

applet. J2EE dipergunakan untuk mengembangkan aplikasi-aplikasi berbasis *client/server* berskala *enterprise*. Dan J2ME diaplikasikan pada berbagai perangkat kecil dengan jumlah memori, kapasitas penyimpanan dan *user interface* terbatas, seperti ponsel dan PDA.

Hingga saat ini ada dua versi platform Java J2ME yang umum digunakan pada telepon genggam yaitu MIDP 1.0 yang masih memiliki spesifikasi sederhana dan menyediakan fungsi dasar untuk aplikasi *mobile*, di antaranya *basic user interface* dan keamanan jaringan. Dan MIDP 2.0 yang sudah ditambahkan berbagai fasilitas seperti *game*, multimedia, dukungan berbagai jenis konektivitas, maupun OTA.⁷

2.8. GPS

GPS (Global Positioning System) merupakan sistem navigasi satelit yang dikembangkan oleh Departemen Pertahanan Amerika Serikat (US DoD = United States Department of Defense). GPS dapat memberitahu posisi geografis dari pengguna (lintang, bujur, dan ketinggian di atas permukaan laut). Jadi dimanapun seseorang berada di muka bumi ini, posisinya dapat diketahui dengan tepat. GPS terdiri dari 3 segmen: Segmen angkasa, kontrol/pengendali, dan pengguna:

- Segmen angkasa: terdiri dari 24 satelit yang beroperasi dalam 6 orbit pada ketinggian 20.200 km dan inklinasi 55 derajat dengan periode 12 jam (satelit akan kembali ke titik yang sama dalam 12 jam). Satelit tersebut memutar orbitnya sehingga minimal ada 6 satelit yang dapat dipantau pada titik manapun di bumi ini. Satelit tersebut mengirimkan posisi dan waktu kepada pengguna seluruh dunia.
- Segmen Kontrol/Pengendali: terdapat pusat pengendali utama yang terdapat di Colorado Springs, dan 5 stasiun pemantau lainnya dan 3 antena yang tersebar di bumi ini. Stasiun pemantau memantau semua satelit GPS dan mengumpulkan informasinya. Stasiun pemantau kemudian mengirimkan informasi tersebut kepada pusat pengendali utama yang kemudian melakukan perhitungan dan pengecekan orbit satelit. Informasi tersebut kemudian dikoreksi dan dilakukan pemuktahiran dan dikirim ke satelit GPS.

⁷ Anonim. 5 September 2009, Apa Itu Java?, <http://www.zahroel.co.cc/2008/12/apa-itu-java.html>

- Segmen Pengguna: Pada sisi pengguna dibutuhkan penerima GPS (selanjutnya disebut perangkat GPS) yang biasanya terdiri dari penerima, prosesor, dan antena, sehingga dimanapun seseorang berada di muka bumi ini (tanah, laut, dan udara) dapat menerima sinyal dari satelit GPS dan kemudian menghitung posisi, kecepatan dan waktu.⁸

⁸ Telecom Education. 5 September 2009, Apa Itu GPS?, <http://telecomeducation.blogspot.com/2009/06/apa-itu-gps.html>

BAB III

PERANCANGAN DAN REALISASI SISTEM

3.1. Perancangan Sistem

3.1.1. Deskripsi Sistem

Sistem ini merupakan sistem yang dapat melakukan proses pengambilan data multimedia berbasis lokasi secara otomatis dari *handphone* (kamera *handphone* dan GPS) dan juga komputer desktop (*webcam*) yang akan diproses sehingga menjadi data yang akurat (dapat menampilkan data sesuai aslinya) dan teroptimisasi untuk dikirimkan melalui jaringan internet ke *server*. Keluaran dari sistem ini berupa data yang bersifat universal, mengikuti standar yang umum digunakan (contoh: XML, gambar JPEG, dan lain-lain), serta memiliki ekstensibilitas tinggi sehingga dapat digunakan dengan mudah oleh pengembang aplikasi lainnya yang membutuhkan data-data seperti ini.

- **Nama Sistem**

Aplikasi Sistem Pengiriman Informasi Data Lokasi

- **Fungsi Sistem**

Untuk mengirimkan informasi berupa gambar beserta dengan koordinat bumi dimana gambar tersebut diambil secara otomatis dan menampilkannya dengan menggunakan antar muka peta.

- **Spesifikasi**

Server

- | | | |
|----|---------------------|-----------------------|
| 1. | Sistem Operasi | : Linux |
| 2. | <i>Server Web</i> | : Apache HTTP Server |
| 3. | <i>Server Peta</i> | : Google Maps |
| 4. | RDBMS | : MySQL 5 |
| 5. | <i>Skrip Server</i> | : PHP 5 |
| 6. | <i>Video Codec</i> | : FFMPEG dan LIBGJPEG |
| 7. | <i>Audio Codec</i> | : AMR |

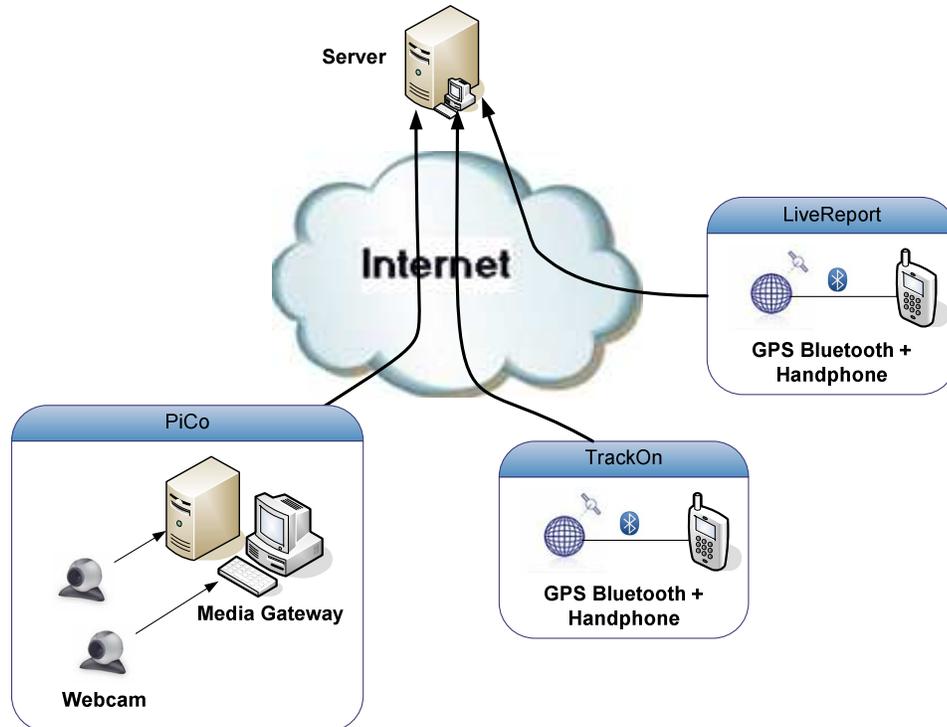
Client Dekstop

- | | | |
|----|----------------|----------------------|
| 1. | Sistem Operasi | : Windows XP SP2 |
| 2. | .NET Framework | : .Net Framework 3.5 |

Client Handphone

1. *Java Runtime Environment* : JRE 1.6
2. *MIDP* : MIDP 2.0
3. *CLDC* : CLDC 1.1
4. *Video Codec* : FFMPEG dan LIBJPEG
5. *Audio Codec* : AMR

3.1.2. Diagram Blok Sistem



Gambar 3.1. Diagram blok sistem

Sistem ini terbagi menjadi 3 komponen utama yang bertugas untuk mengirimkan data ke *server* dari perangkat pencitraan digital, yaitu komponen pemantauan lokasi menggunakan *webcam* (PiCo), komponen pengetrekan menggunakan *handphone* (TrackOn), komponen streaming video menggunakan *handphone* (LiveReport). Kesemua komponen tersebut selain mengirimkan gambar-gambar yang didapat dari perangkat pencitraan digital dan juga mengirimkan koordinat bumi dimana gambar tersebut didapat.

PiCo bekerja dengan menggunakan *webcam* untuk mengambil gambar suatu lokasi dengan menggunakan sebuah komputer sebagai pusat pengiriman data ke *server* yang keluarannya berupa gambar yang terus diperbaharui setiap interval waktu tertentu dan *file*

XML yang berisi informasi-informasi tambahan dari gambar yang dikirim (lokasi, nama gambar, dan lain-lain).

TrackOn dan LiveReport bekerja pada satu lingkungan yang sama yaitu menggunakan *handphone*, GPS, dan kamera *handphone* untuk mengirimkan data ke *server* menggunakan konektivitas *handphone* ke jaringan internet. Untuk TrackOn, sistem akan bekerja sebagai pengambil data otomatis untuk pencatatan jalur perjalanan seseorang beserta gambar dari lingkungan yang dilewati selama perjalanan dan mengirimkannya ke *server* menggunakan jaringan internet. Untuk LiveReport, sistem akan mengirimkan data video dari perangkat *handphone* secara langsung ditambah dengan pencatatan lokasi saat pengambilan video (menggunakan GPS). Pengiriman data tersebut akan menggunakan konektivitas *handphone* ke jaringan internet, kemudian data yang didapat akan dioptimalisasikan pada *server streaming* yang keluarannya berupa data XML dan *file* video itu sendiri.

3.1.3. Analisa Keperluan

Sistem ini terdiri dari 3 komponen utama seperti yang telah dijelaskan sebelumnya, yaitu PiCo, TrackOn, dan LiveReport.

Tabel 3.1. Analisa Keperluan Sistem

Komponen	Keperluan Primer	Keperluan Sekunder
PiCo	Performa tinggi	Penggunaan CPU kecil
		Penggunaan memori kecil
		Penggunaan <i>bandwith</i> internet kecil
	Fleksibilitas tinggi	Dapat mengkonfigurasi hasil pengambilan gambar
	Implementasi umum	Perangkat yang digunakan harus mudah dicari
TrackOn	Mudah digunakan	Sedikit interaksi oleh pengguna
		Antarmuka pengguna sederhana
	Performa tinggi	Penggunaan memori kecil
		Penggunaan <i>bandwith</i> internet kecil
	Aman digunakan	Kemungkinan eror kecil
LiveReport	Mudah digunakan	Sedikit interaksi oleh pengguna
		Antarmuka pengguna sederhana
	Performa tinggi	Penggunaan <i>bandwith</i> internet kecil

Berdasarkan dari kebutuhan pengguna di atas, berikut adalah spesifikasi dari fungsi-fungsi yang terdapat dari setiap komponen:

PiCo

Tabel 3.2. Spesifikasi fungsi-fungsi pada PiCo

Fungsi	Atribut	Nilai	Alasan
Pencitraan digital	Perangkat	<i>Webcam</i>	Umum ditemukan dan murah harganya
Pengambilan Gambar	Masukan	DirectShow	Performa dan fleksibilitas tinggi
	Format Keluaran	RGB	Didukung oleh semua komputer
Pengkodean Gambar	Standar	JPEG	Mudah digunakan dan cocok untuk pengambilan gambar bersifat fotografi
Transfer data ke Media Gateway	Media	USB	Umum digunakan dan didukung oleh banyak komputer
Transfer Data ke Server	Protokol Transfer Data	FTP	Mudah digunakan dan cocok untuk transfer data dalam jumlah besar ke <i>server</i>
		HTTP	Mudah digunakan dan <i>server</i> dapat mengirimkan balasan ke <i>client</i>

TrackOn

Tabel 3.3. Spesifikasi fungsi-fungsi pada TrackOn

Fungsi	Atribut	Nilai	Alasan
Pengambilan Gambar	Format Keluaran	RGB	Didukung oleh kebanyakan <i>handphone</i> berkamera
Pengkodean Gambar	Standar	JPEG	Mudah digunakan dan cocok untuk pengambilan gambar bersifat fotografi
Pengambilan Koordinat Bumi	Masukan	GPSTrack	Mudah dan aman digunakan
Koneksi Bluetooth	Konektor	J4ME	Mudah dan aman digunakan
Transfer Data	Protokol Transfer Data	HTTP	Mudah digunakan dan didukung di banyak <i>handphone</i>

LiveReport

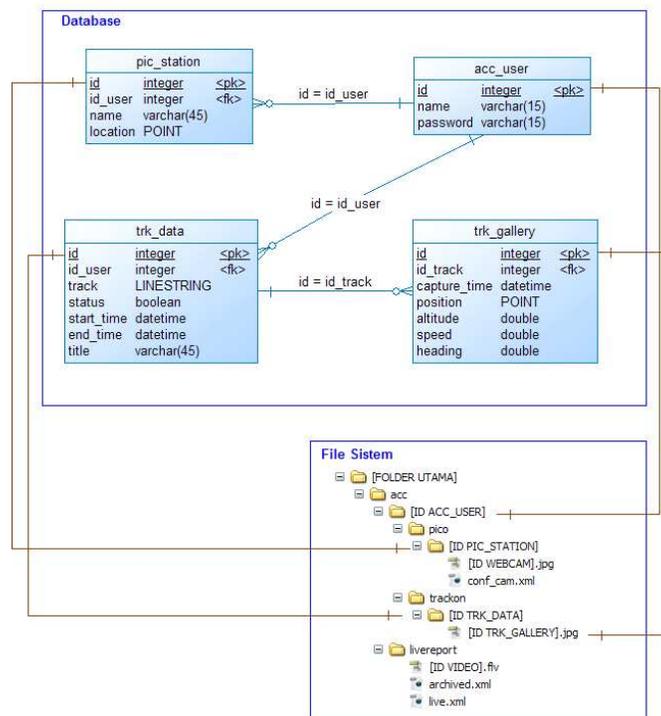
Tabel 3.4. Spesifikasi fungsi-fungsi pada LiveReport

Fungsi	Atribut	Nilai	Alasan
Pengambilan Video	Format Keluaran	YUV	Didukung oleh kebanyakan <i>handphone</i> berkamera
Pengkodean Video	Standar	JPEG	Karena video lebih mulus, <i>bandwidth</i> video kecil, dan penggunaan memori kecil
Transfer Data	Protokol Transfer Data	TCP/IP	Dapat mengirimkan data secara terus menerus

3.1.4. Spesifikasi Data

3.1.5.1. Design Fisik Data

Data keluaran sistem terdiri dari 2 jenis data yaitu file dan *database record*. Data tipe *file* terdiri dari *file* gambar dan video suatu lokasi yang didapat dari komponen-komponen penyiar sistem, selain itu juga terdapat file xml yang digunakan sebagai deskriptor dari tiap file gambar dan video. Sedangkan untuk data tipe *database record* digunakan untuk menyimpan data-data lokasi yang berbentuk teks dan data pengguna.



Gambar 3.2. Diagram fisik data sistem

Tabel `acc_user` pada *database* menyimpan data pengguna untuk dapat *login* ke sistem dan menyimpan data-data dari komponen-komponen penyiaran sistem di *database*. Tabel `trk_data` menyimpan data mengenai trek-trek yang telah dibuat oleh pengguna sistem. Tabel `trk_gallery` menyimpan data-data mengenai gambar hasil jepretan pengguna ketika sedang membuat trek. Tabel `pic_station` menyimpan data Media Gateway yang telah dibuat oleh pengguna.

Data yang ada di *database* digunakan sebagai referensi utama dari setiap file sistem hasil keluaran komponen-komponen penyiaran sistem. *File-file* yang dibuat oleh komponen-komponen penyiaran sistem harus tersinkron dengan data yang ada di *database*, jika data yang ada di *database* dihapus maka file-file yang bersangkutan juga harus ikut dihapus begitu juga jika data di *database* diubah maka file-file yang bersangkutan harus ikut diubah. Jumlah setiap *file-file* dan *folder-folder* yang memiliki relasi di *database* harus sama jumlahnya.

3.1.5.2. Design Skema XML

File-file xml yang berada di beberapa *folder* di file sistem dibuat otomatis oleh sistem yang berisi data-data tambahan mengenai *file-file* yang dibuat oleh komponen-komponen penyiar sistem untuk mempermudah pengambilan *file-file* tersebut dan memberi info-info tambahan pada *file-file* tersebut. Sistem ini memiliki 3 *file* xml utama, yaitu `conf_cam.xml`, `archived.xml`, dan `live.xml`.

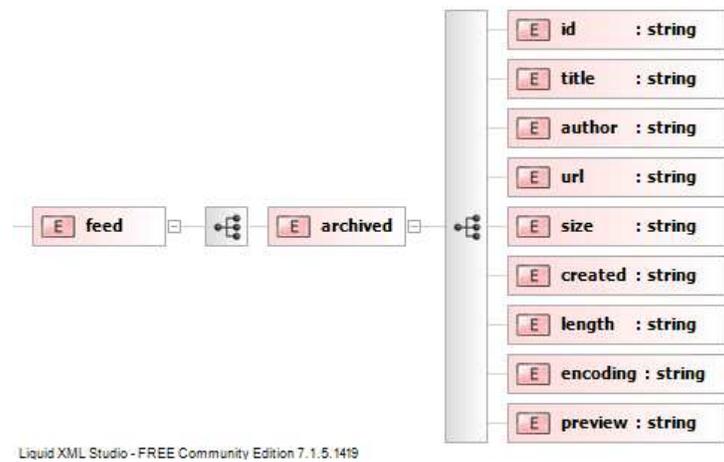


Gambar 3.3. Diagram skema XML `conf_cam.xml`

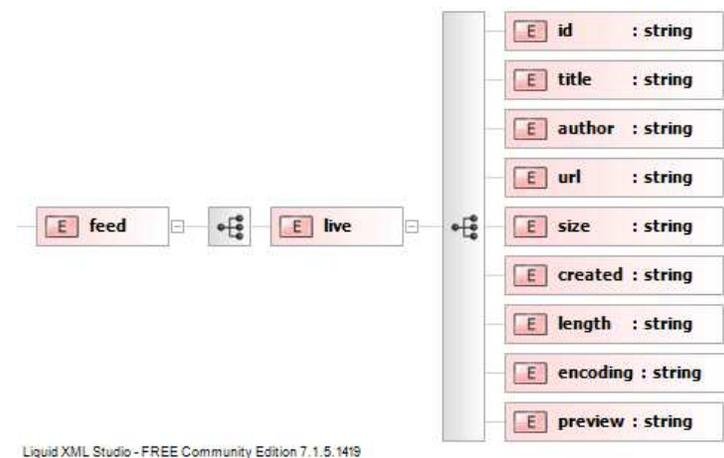
File `conf_cam.xml` adalah xml yang bertugas sebagai deskriptor dari gambar-gambar yang dikirimkan oleh PiCo ke *server*. `conf_cam.xml` terdapat pada setiap *folder* Media Gateway di *folder* PiCo. `conf_cam.xml` dapat digunakan untuk mengetahui berapa banyak *file-file* gambar pada suatu Media Gateway yang dikirimkan ke *server* beserta nama-nama dari gambar tersebut.

`conf_cam.xml` mempunyai Media Gateway sebagai *root element*-nya. Media Gateway memiliki elemen-elemen `cam` yang menyimbolkan satu per satu kamera yang dikirimkan oleh PiCo ke *server*. Elemen `cam` memiliki elemen `id` dan `name`, elemen `id`

bertugas untuk memberikan id dari kamera tersebut dan elemen “name” memberi nama dari kamera tersebut.



Gambar 3.4. Diagram skema XML `archived.xml`



Gambar 3.5. Diagram skema XML `live.xml`

File XML `archived.xml` dan `live.xml` digunakan sebagai deskriptor dari *file-file* video yang dibuat oleh LiveReport dan disimpan di *server*. Kedua file tersebut memiliki elemen “feed” sebagai *root element*. Hampir keseluruhan struktur XML sama, hanya elemen “archived” dan “live” yang membedakan kedua file tersebut. Elemen “id” digunakan sebagai id video dan id dari video tersebut dienkripsi menggunakan MD5. Elemen “title” digunakan untuk nama video. Elemen “author” digunakan untuk memberitahukan pembuat video. Elemen “url” digunakan untuk mencari URL lokasi penyimpanan video. Elemen “size” digunakan untuk menyimpan ukuran video. Elemen “created” digunakan untuk mendapatkan kapan video tersebut dibuat. Elemen “length”

merupakan panjang paket data dari video tersebut. Elemen “encoding” digunakan untuk mencari *folder encoder* untuk video tersebut. Elemen “preview” digunakan sebagai URL dari *path* ke gambar untuk preview video.

3.1.5.3. Detail Elemen Data di Database

Tabel *acc_user*

Tabel 3.5. Elemen data “id” pada tabel “*acc_user*”

acc_user: id	
Nama elemen data	Id
Tipe data	<i>Unsigned integer</i>
Definisi	ID pengguna sistem pada <i>database</i>
Fungsi	Digunakan untuk mencari data lokasi baik yang disimpan di <i>database</i> atau di file sistem
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	Diisi otomatis oleh <i>database</i>

Tabel 3.6. Elemen data “name” pada tabel “*acc_user*”

acc_user: name	
Nama elemen data	Name
Tipe data	<i>Varchar</i>
Definisi	Nama pengguna sistem pada <i>database</i>
Fungsi	Digunakan untuk <i>login</i>
Besar data	15 karakter
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel 3.7. Elemen data “password” pada tabel “*acc_user*”

acc_user: password	
Nama elemen data	Name
Tipe data	<i>Varchar</i>
Definisi	<i>Password</i> pengguna sistem pada <i>database</i>
Fungsi	Digunakan untuk <i>login</i>

Besar data	15 karakter
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	<i>Password</i>

Tabel *trk_data*

Tabel 3.8. Elemen data “id” pada tabel “*trk_data*”

trk_data: id	
Nama elemen data	Id
Tipe data	<i>Unsigned integer</i>
Definisi	ID trek pada <i>database</i>
Fungsi	Digunakan untuk mengidentifikasi trek pengguna
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	Diisi otomatis oleh <i>database</i>

Tabel 3.9. Elemen data “id_user” pada tabel “*trk_data*”

trk_data: id_user	
Nama elemen data	id_user
Tipe data	<i>Unsigned integer</i>
Definisi	ID pengguna yang mempunyai trek bersangkutan
Fungsi	Untuk mencari pemilik dari sebuah trek
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel 3.10. Elemen data “track” pada tabel “trk_data”

trk_data: track	
Nama elemen data	Track
Tipe data	OGC <i>LineString</i>
Definisi	Jalur trek dalam bentuk kumpulan koordinat-koordinat yang beraturan
Fungsi	Digunakan untuk melihat trek menggunakan peta
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	Data berbasis geospasial

Tabel 3.11. Elemen data “status” pada tabel “trk_data”

trk_data: status	
Nama elemen data	Status
Tipe data	<i>boolean</i>
Definisi	Flag untuk menandakan status dari trek
Fungsi	Untuk menandakan apakah data dapat dilihat untuk pengguna yang bersangkutan atau bebas
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel 3.12. Elemen data “start_time” pada tabel “trk_data”

trk_data: start_time	
Nama elemen data	Start_time
Tipe data	<i>Datetime</i>
Definisi	Waktu dimana trek mulai dibuat pada <i>database</i>
Fungsi	Digunakan untuk mengetahui kapan trek dibuat
Besar data	-
Format	YYYY-MM-DD HH:MM:SS
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel 3.13. Elemen data “end_time” pada tabel “trk_data”

trk_data: end_time	
Nama elemen data	End_time
Tipe data	Datetime
Definisi	Waktu dimana trek berakhir
Fungsi	Digunakan untuk mengetahui kapan trek berakhir
Besar data	-
Format	YYYY-MM-DD HH:MM:SS
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel 3.14. Elemen data “track” pada tabel “trk_data”

trk_data: title	
Nama elemen data	Title
Tipe data	<i>Varchar</i>
Definisi	Judul dari trek
Fungsi	Digunakan untuk melihat judul dari trek
Besar data	45
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel trk_gallery

Tabel 3.15. Elemen data “id” pada tabel “trk_gallery”

trk_gallery: id	
Nama elemen data	Id
Tipe data	<i>Unsigned integer</i>
Definisi	ID gambar hasil jepretan pada suatu trek di <i>database</i>
Fungsi	Digunakan untuk mengidentifikasi gambar hasil jepretan pada suatu trek
Besar data	-
Format	-
Satuan ukuran	-

Presisi	-
Karakteristik	Diisi otomatis oleh <i>database</i>

Tabel 3.16. Elemen data “*id_track*” pada tabel “*trk_gallery*”

trk_gallery: id_track	
Nama elemen data	<i>id_track</i>
Tipe data	<i>Unsigned integer</i>
Definisi	ID trek yang mempunyai gambar hasil jepretan bersangkutan
Fungsi	Untuk mencari pemilik dari sebuah gambar hasil jepretan pada suatu trek
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel 3.17. Elemen data “*capture_time*” pada tabel “*trk_gallery*”

trk_gallery: capture_time	
Nama elemen data	<i>Capture_time</i>
Tipe data	<i>Datetime</i>
Definisi	Waktu dimana gambar hasil jepretan masuk ke <i>server</i>
Fungsi	Untuk mencari kapan gambar hasil jepretan disimpan
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel 3.18. Elemen data “*position*” pada tabel “*trk_gallery*”

trk_gallery: position	
Nama elemen data	<i>Position</i>
Tipe data	<i>OGC Point</i>
Definisi	Koordinat bumi dimana gambar hasil jepretan diambil
Fungsi	Untuk mencari dimana gambar hasil jepretan diambil
Besar data	-
Format	-

Satuan ukuran	-
Presisi	-
Karakteristik	Data berbasis geospasial

Tabel 3.19. Elemen data “altitude” pada tabel “trk_gallery”

trk_gallery: altitude	
Nama elemen data	Altitude
Tipe data	<i>Double</i>
Definisi	Ketinggian dimana gambar hasil jepretan masuk diambil
Fungsi	Untuk mencari ketinggian dimana gambar hasil jepretaan diambil
Besar data	-
Format	-
Satuan ukuran	Meter
Presisi	4 angka di belakang koma
Karakteristik	-

Tabel 3.20. Elemen data “speed” pada tabel “trk_gallery”

trk_gallery: speed	
Nama elemen data	Speed
Tipe data	<i>double</i>
Definisi	Kecepatan gerak pengguna ketika sistem mengambil gambar
Fungsi	Untuk mencari kecepatan pengguna ketika sistem mengambil gambar
Besar data	-
Format	-
Satuan ukuran	km/jam
Presisi	4 angka di belakang koma
Karakteristik	-

Tabel 3.21. Elemen data “heading” pada tabel “trk_gallery”

trk_gallery: heading	
Nama elemen data	Heading
Tipe data	<i>double</i>
Definisi	Arah gerak pengguna ketika sistem mengambil gambar
Fungsi	Untuk mencari arah gerak pengguna ketika sistem mengambil gambar

Besar data	-
Format	-
Satuan ukuran	Derajat
Presisi	4 angka di belakang koma
Karakteristik	-

Tabel pic_station

Tabel 3.22. Elemen data "id" pada tabel "pic_station"

pic_station: id	
Nama elemen data	Id
Tipe data	<i>Unsigned integer</i>
Definisi	ID Media Gateway pada <i>database</i>
Fungsi	Digunakan untuk mencari data Media Gateway baik yang disimpan di <i>database</i> atau di <i>file</i> sistem
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	Diisi otomatis oleh <i>database</i>

Tabel 3.23. Elemen data "id_user" pada tabel "pic_station"

pic_station: id_user	
Nama elemen data	id_user
Tipe data	<i>Unsigned integer</i>
Definisi	ID pengguna yang mempunyai Media Gateway bersangkutan
Fungsi	Untuk mencari pemilik dari sebuah Media Gateway
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel 3.24. Elemen data “name” pada tabel “pic_station”

pic_station: name	
Nama elemen data	Name
Tipe data	Varchar
Definisi	Nama Media Gateway pada <i>database</i>
Fungsi	Digunakan untuk mendapatkan nama Media Gateway
Besar data	45 karakter
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	-

Tabel 3.25. Elemen data “location” pada tabel “pic_station”

pic_station: location	
Nama elemen data	Name
Tipe data	OGC <i>Point</i>
Definisi	Koordinat bumi dimana Media Gateway berada
Fungsi	Untuk mencari dimana Media Gateway berada
Besar data	-
Format	-
Satuan ukuran	-
Presisi	-
Karakteristik	Data berbasis geospasial

3.1.5.4. Detail File-File Multimedia

Setiap komponen penyiaran sistem menciptakan file multimedia masing-masing, komponen PiCo dan komponen TrackOn menciptakan file multimedia bertipe gambar sedangkan komponen LiveReport menciptakan file multimedia bertipe video.

TrackOn

Tipe data: gambar

Format gambar: JPEG

Rasio kualitas: 80%

PiCo

Tipe data: gambar

Format gambar: JPEG

Rasio kualitas: 80%

LiveReport

Tipe data: *audio/video*

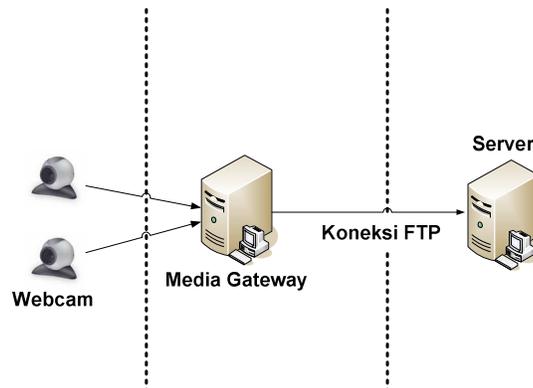
Format video: FLV

Audio codec: AMR 800 Hz 1,6 kb/s

Video codec: LIBJPEG 1-20 KB/frame dan 15 FPS || FFMPEG 0,3-5 KB/frame dan 5-10 FPS

3.1.5. Komponen PiCo

PiCo berbentuk sebuah aplikasi desktop yang dipasangkan pada komputer yang bertindak sebagai Media Gateway. PiCo mengirimkan data dari *webcam* secara terus menerus ke *server*. Data yang telah dikirimkan dari PiCo digunakan untuk menampilkan gambar dari suatu lokasi dengan posisi pengambilan gambar yang relatif tetap. Gambar yang dihasilkan dari komponen ini telah diproses terlebih dahulu seperti pengaturan pencahayaan, besar gambar, dan lain-lain supaya dapat diterima oleh pengguna dengan kualitas gambar yang memadai.



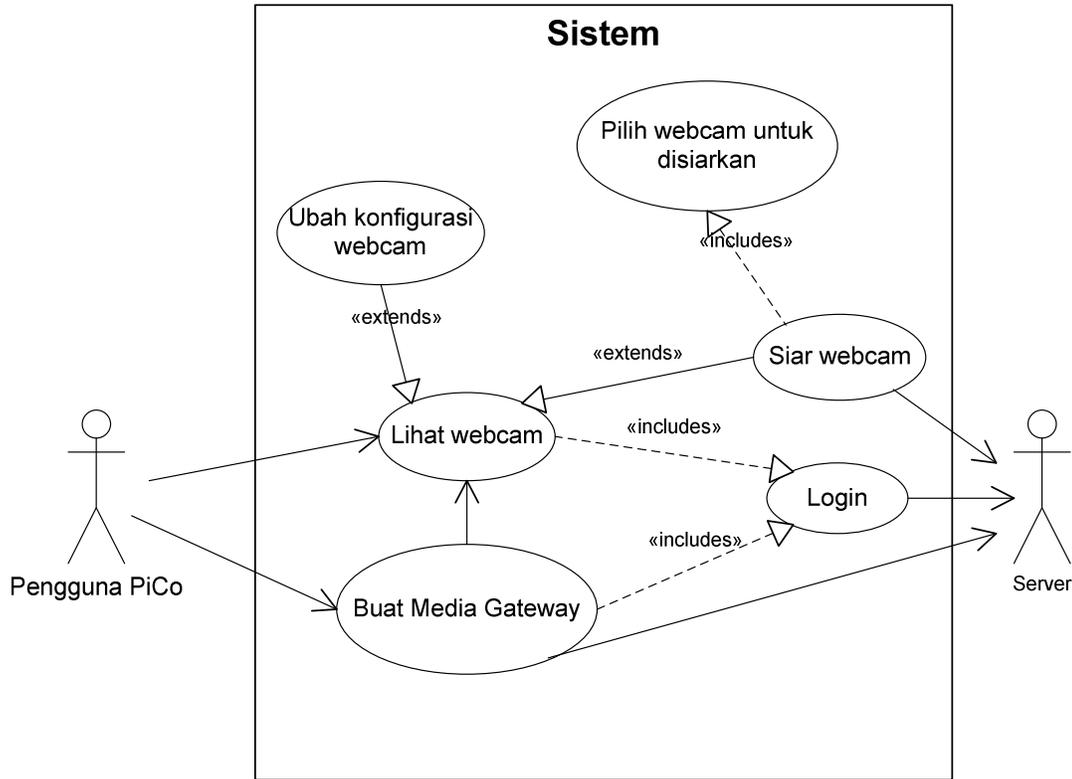
Gambar 3.6. Diagram alur kerja PiCo

Secara detail alur dari proses PiCo berdasarkan diagram di atas adalah:

1. *Webcam* mentransmisikan gambar yang ia dapat ke Media Gateway.
2. Transmisi gambar dari *webcam* ditangkap oleh Media Gateway dengan menggunakan Microsoft DirectShow untuk mengakses perangkat *webcam* yang terhubung dengan komputer sekaligus memanipulasi gambar yang didapat sebelum dikirimkan ke *server*.

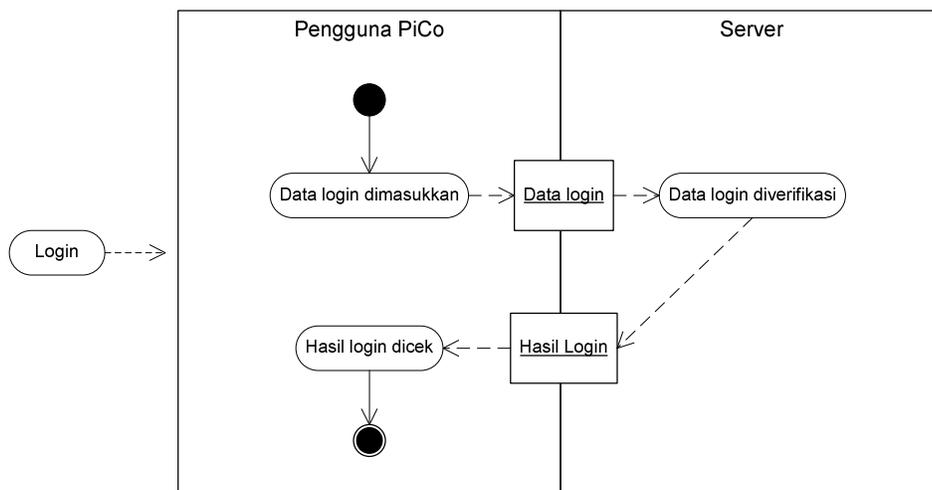
- Gambar yang telah diproses lalu dikirim ke *server* melalui protokol FTP beserta dengan informasi-informasi tambahan dari gambar tersebut dan kemudian data disimpan pada *server* dengan mengikuti aturan penempatan data di *server*.

3.1.5.5. Use Case Pico

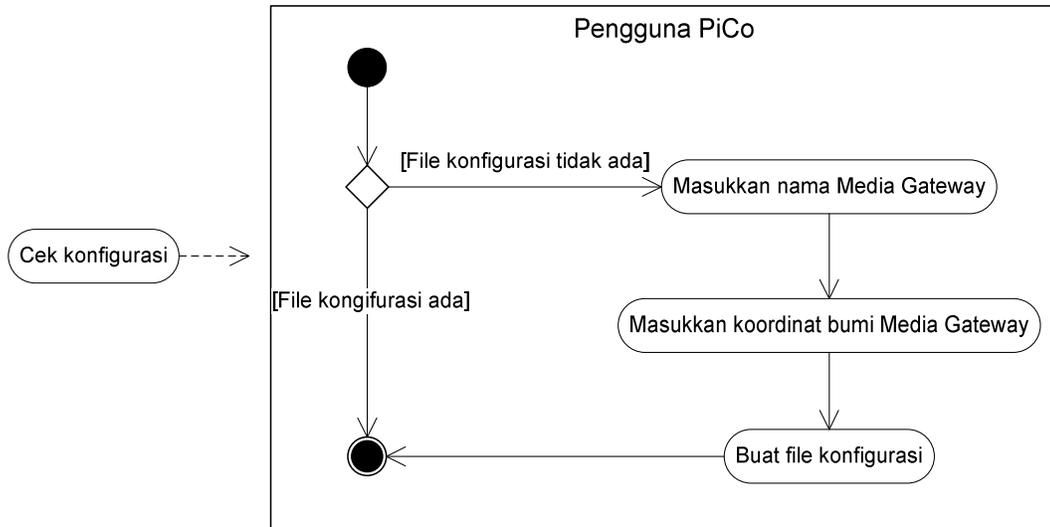


Gambar 3.7. Use Case PiCo

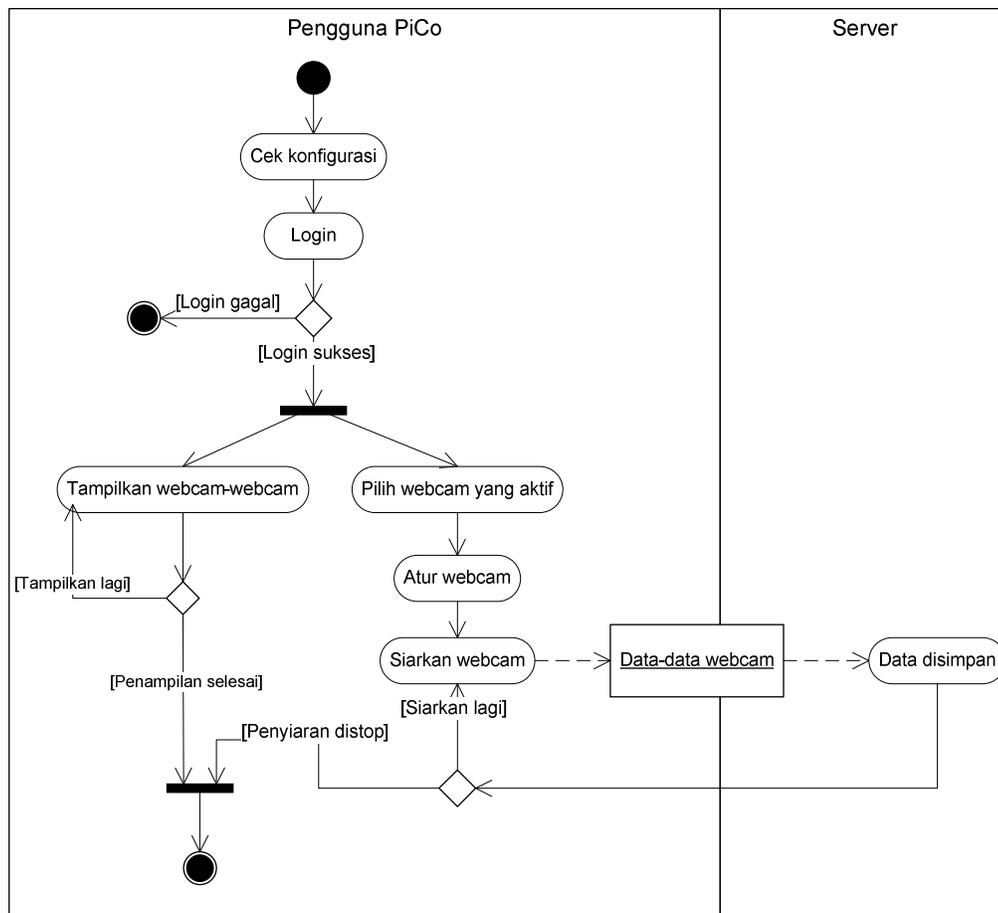
3.1.5.6. Activity Diagram Pico



Gambar 3.8. Activity diagram PiCo



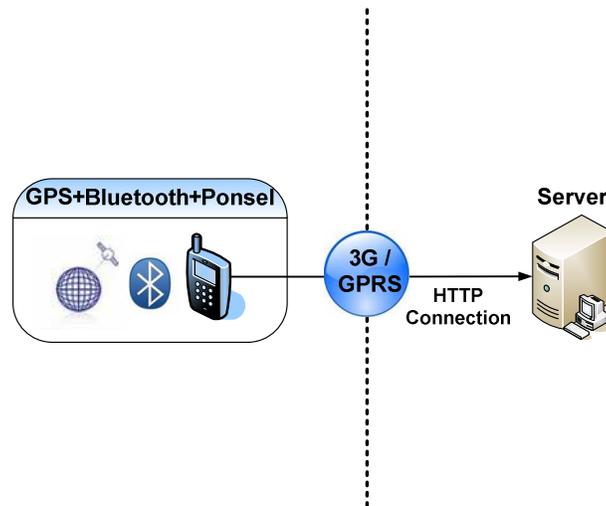
Gambar 3.9. Activity diagram pengecekan konfigurasi PiCo



Gambar 3.10. Activity Diagram Penyiaran Webcam PiCo

3.1.6. Komponen TrackOn

TrackOn merupakan sebuah aplikasi *handphone* yang dipasangkan pada *handphone* yang ingin dijadikan sebagai perangkat pengetrekan. TrackOn mengirimkan data dari kamera *handphone* serta GPS yang dilakukan terus menerus dengan jeda waktu tertentu tiap pengiriman data ke *server*. Data yang dikirimkan dari TrackOn digunakan untuk menampilkan trek perjalanan seseorang beserta dengan gambar-gambar yang diambil pada lokasi-lokasi tertentu selama perjalanan.

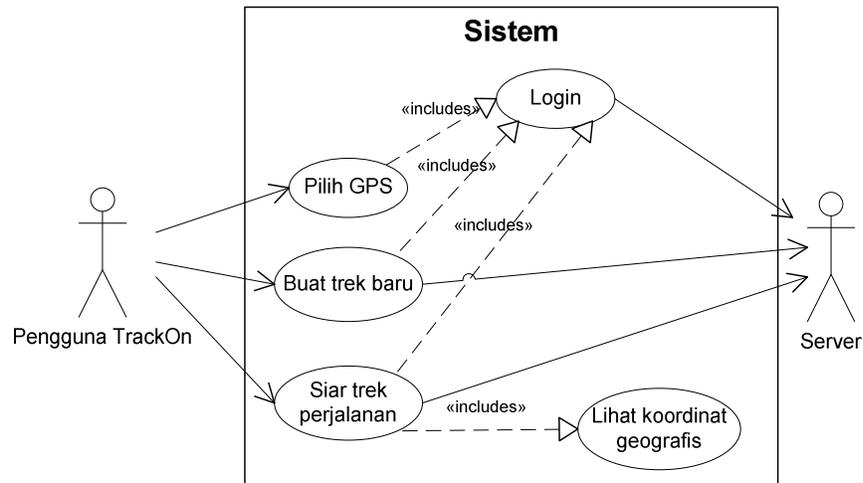


Gambar 3.11. Diagram Alur Kerja TrackOn

Secara detail alur dari proses TrackOn berdasarkan diagram di atas adalah:

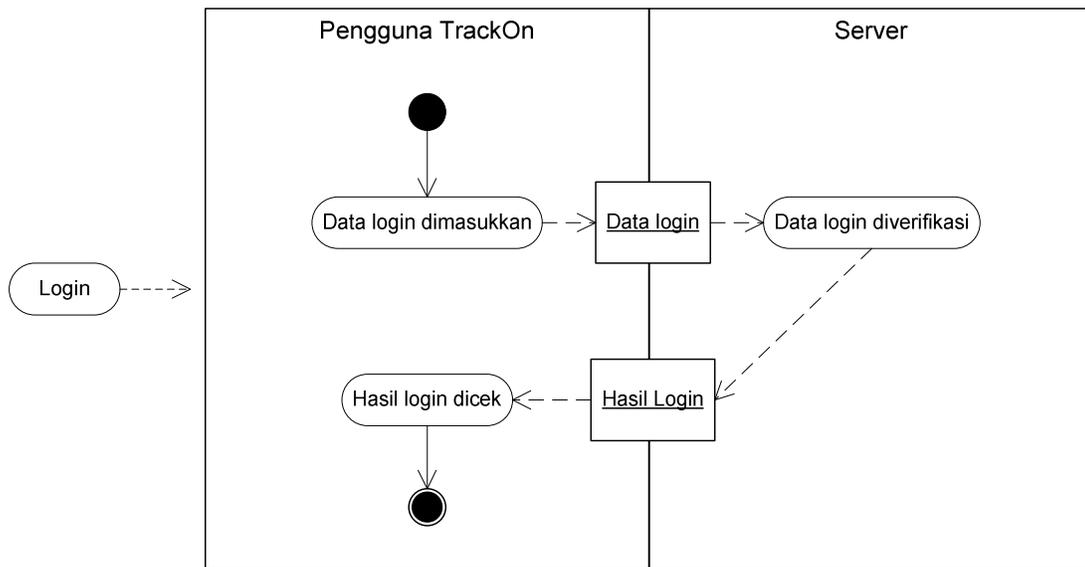
1. Ketika pengguna menggunakan TrackOn, secara otomatis program pada ponsel akan mengambil gambar dari lokasi di mana pengguna berada dan terus berulang dalam jangka waktu tertentu.
2. Ketika sebuah gambar diambil, TrackOn secara bersamaan akan mengambil posisi dimana gambar tersebut diambil menggunakan GPS dalam bentuk koordinat geografis.
3. Gambar dan data lokasi yang telah didapat dikirimkan ke *server* melalui jaringan internet pada *handphone* seperti GPRS atau 3G.

3.1.6.1. Use Case TrackOn

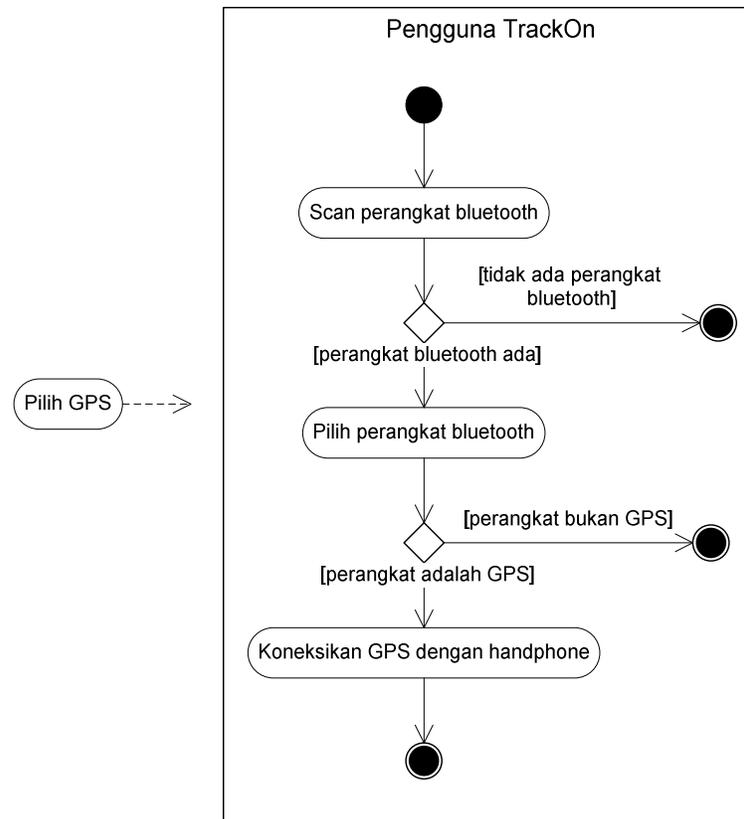


Gambar 3.12. Use Case TrackOn

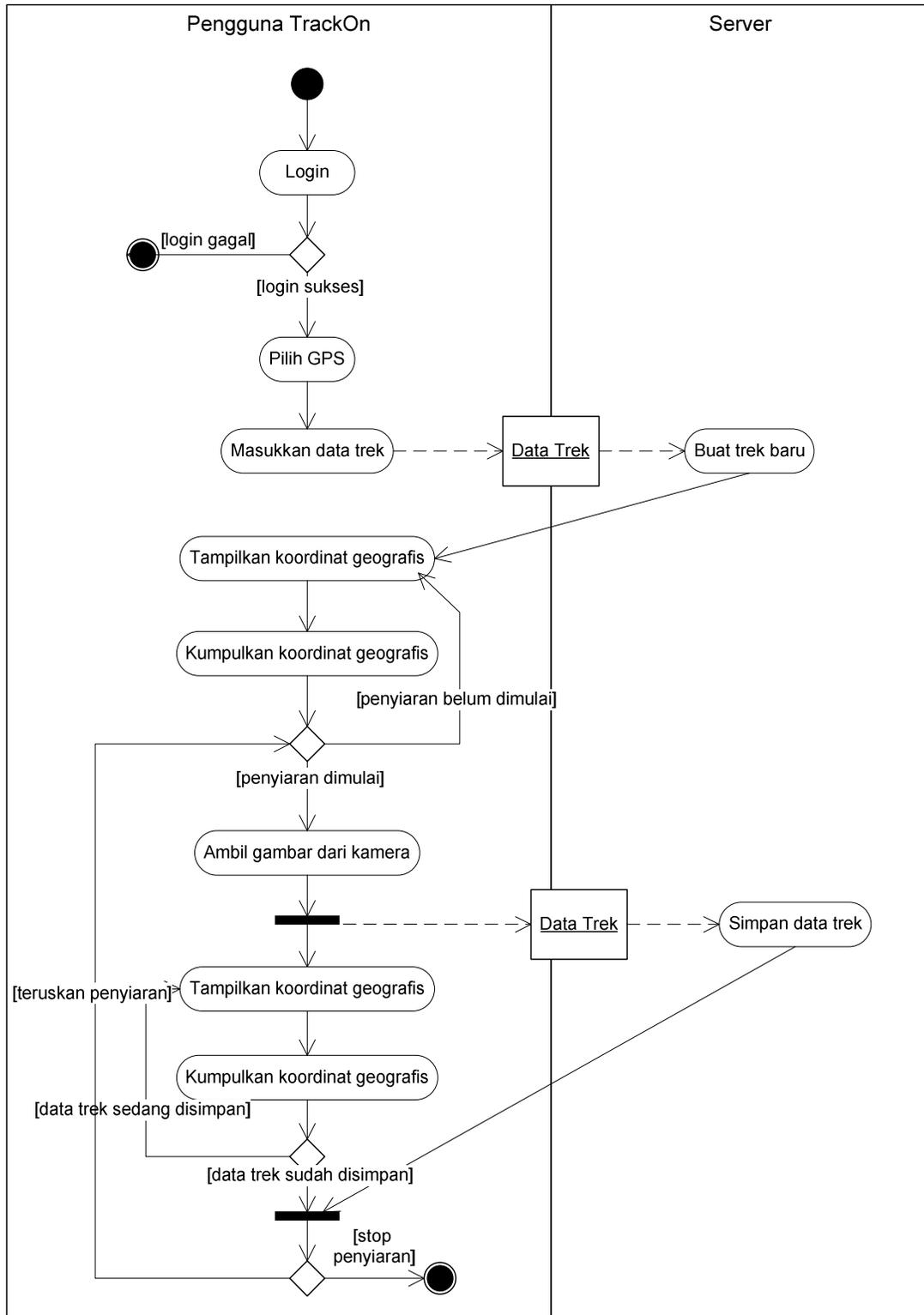
3.1.6.2. Activity Diagram TrackOn



Gambar 3.13. Activity diagram login pada TrackOn



Gambar 3.14. Activity diagram pemilihan GPS TrackOn



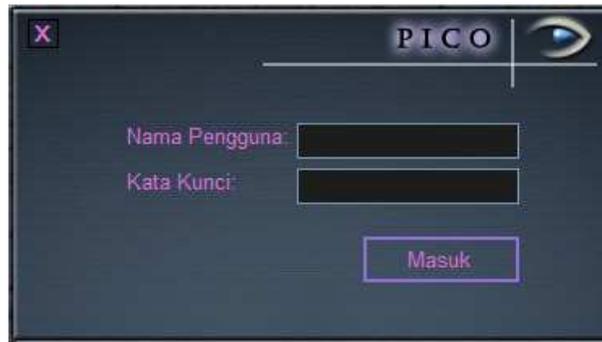
Gambar 3.15. Activity Diagram Penyiaran Trek TrackOn

3.2. Realisasi Sistem

3.2.1. Realisasi Komponen PiCo

PiCo dibuat dengan menggunakan bahasa pemrograman C# dan menggunakan pustaka DirectShow .Net yang berbasis pada Microsoft DirectShow untuk pengaturan multimedia pada komputer *desktop*. Fungsi utama dari komponen PiCo yang dibuat adalah untuk memproses data yang didapat dari *webcam* untuk kemudian mengirimkannya ke *server*.

Form login adalah *form* utama ketika menjalankan aplikasi. Proses login dilakukan dengan cara mencocokkan data pengguna yang dimasukkan oleh pengguna dengan data yang ada di *server* melalui koneksi *database* MySQL. Jika sukses maka PiCo akan mendeteksi konfigurasi PiCo terlebih dahulu.



Gambar 3.16. Form Login PiCo

Proses login dilakukan dengan kodingan seperti di bawah ini:

```

DataSet dbResult = new DataSet();
MySqlDataAdapter dbAdapter = new MySqlDataAdapter("SELECT id FROM
acc_user WHERE name = \" + txtUser.Text + \" AND password = \" +
txtPassword.Text + \"", dbCon);
dbAdapter.Fill(dbResult, "login");

if (dbResult.Tables["login"].Rows.Count == 0)
{
    MessageBox.Show(this, "Login gagal...\r\nSilahkan dicoba
lagi", "Error");
}
else
{
    //Proses setelah login
}

```

PiCo membutuhkan 2 file xml yang berisi konfigurasi perangkat *webcam* yang terpasang (*conf_cam.xml*) dan identitas Media Gateway pada komputer di mana PiCo

digunakan agar nanti dapat melakukan sinkronisasi data dengan server (conf.xml). Jika salah satu file tersebut tidak terdeteksi, maka PiCo akan membuatnya secara otomatis. Jika file conf.xml tidak terdeteksi, maka selain membuat file xml baru mengenai konfigurasi identitas computer, PiCo juga akan membuat Media Gateway yang baru di *database*. File XML yang dibuat oleh PiCo menggunakan metode penulisan data teks pada C# seperti di bawah ini:

```
using (StreamWriter sw = new StreamWriter("conf.xml"))
{
    sw.WriteLine("<?xml version=\"1.0\" encoding=\"UTF-8\"?>");
    sw.WriteLine("<ApplicationSettings>");
    sw.WriteLine("<Section Name=\"app_setting\">");
    sw.WriteLine("<Key Name=\"machine_name\" Value=\"\" />");
    sw.WriteLine("<Key Name=\"machine_id\" Value=\"\" />");
    sw.WriteLine("</Section>");
    sw.WriteLine("</ApplicationSettings>");
    sw.Close();
}
```

Pada *form* pembuatan Media Gateway baru terdapat *text box* untuk mengisikan nama Media Gateway beserta koordinat dimana Media Gateway tersebut terletak. Pemasukan data koordinat bumi dilakukan secara manual dengan bantuan program pendeteksi lokasi koordinat bumi menggunakan *handphone* dan GPS Bluetooth.

Gambar 3.17. Form penamaan Media Gateway dan pengisian koordinat

Pembuatan Media Gateway baru dilakukan dengan cara memanggil file php di server melalui protokol HTTP. PiCo mengirimkan data Media Gateway yang baru dengan menggunakan *hashtable* untuk dimasukkan ke dalam query POST HTML. Penggalan kode berikut adalah kode pemrograman C# yang digunakan untuk membuat Media Gateway melalui protokol HTTP beserta pemasukan koordinat lokasinya:

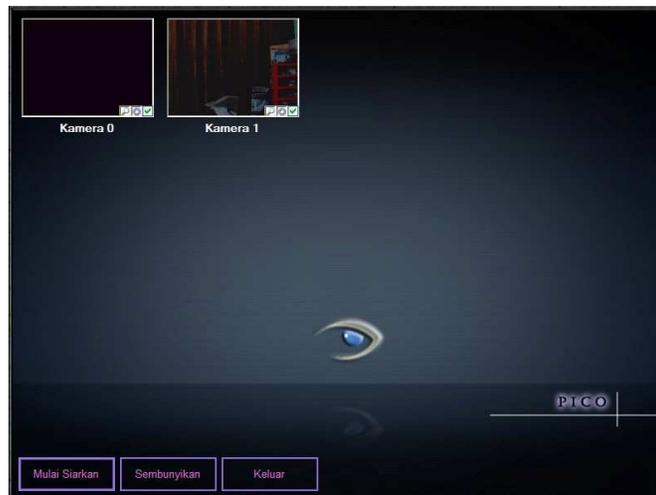
```
MySqlCommand com = new MySqlCommand("insert pic_station
(id_user,name,location) values ('" + AppCom.getUserID() + "\",\"" +
txtMachine.Text + "\", PointFromText('POINT(" + txtLatitude.Text + " " +
txtLongitude.Text + ")')\"", dbCon);
com.ExecuteNonQuery();
```

```

PostSubmitter post = new PostSubmitter();
post.Url = "http://" + AppCom.JED_SERVER +
"/jed/pico_proc/create_gateway.php";
post.PostItems.Add("user", AppCom.getUserName());
post.PostItems.Add("password", AppCom.getUserPass());
post.PostItems.Add("machine_id", AppCom.getMachineID());
post.Type = PostSubmitter.PostTypeEnum.Post;
string result = post.Post();

```

Form pengaturan *webcam* adalah *form* yang bertugas untuk mengatur penyiaran komponen PiCo. *Form* pengaturan *webcam* akan secara otomatis memperlihatkan hasil penangkapan video dari *webcam-webcam* yang terpasang secara tabular dengan 4 kolom dan 2 baris sehingga *form* dapat menampilkan 8 kamera sekaligus dalam satu *form*. Jika kamera lebih dari 8 maka akan muncul *scroller* untuk melihat *webcam* lainnya. Setiap *webcam* juga disisipi kontrol untuk mengatur konfigurasi *webcam*, melihat hasil penangkapan *webcam* dengan ukuran lebih besar, dan penanda apakah *webcam* ingin dimasukkan ke dalam daftar untuk disiarkan ke server.



Gambar 3.18. *Form* pengaturan penyiaran *webcam* PiCo

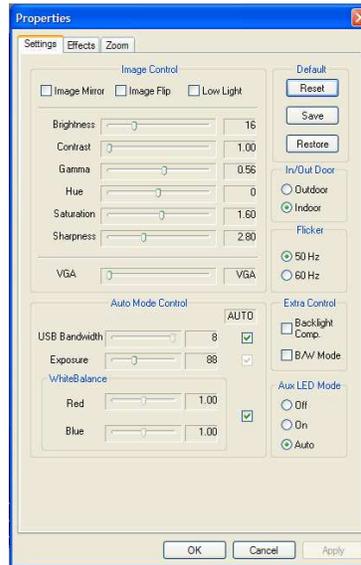
Pendaftaran *webcam* dilakukan dengan menggunakan fungsi-fungsi yang terdapat pada library DirectShowNet seperti pada di bawah ini:

```

Try
{
    tempCam = new Capture(VIDEODEVICE, VIDEOWIDTH, VIDEOHEIGHT,
VIDEOBITSPIXEL, tempCapture);
    arrCam.Add(tempCam);
    Console.WriteLine("Adding camera " + VIDEODEVICE);
}
catch (Exception exc)
{
    Console.WriteLine(exc.StackTrace); }

```

Konfigurasi *webcam* pada PiCo menggunakan konfigurasi yang disediakan oleh driver *webcam* tersebut. Sehingga jika merek atau seri *webcam* berbeda kemungkinan besar antarmuka dan fitur-fitur penyetingan kamera akan berbeda tiap merek atau seri *webcam*.



Gambar 3.19. Form konfigurasi webcam PiCo

Konfigurasi *webcam* dilakukan dengan membuat *filter* menggunakan fungsi yang ada pada *library* DirectShowNet dan memasukkan *path* dari webcam yang terpasang pada Media Gateway ke dalam *filter* tersebut. Hal tersebut bias dilihat pada potongan kodingan di bawah ini:

```

if (theDevice != null)
{
    Marshal.ReleaseComObject(theDevice);
    theDevice = null;
}
theDevice = CreateFilter(FilterCategory.VideoInputDevice, camPath);
DisplayPropertyPage(theDevice);

```

Pengaktifan penyiaran *webcam-webcam* yang terpasang pada Media Gateway dapat dilakukan dengan mengklik tombol penyiaran pada *form* pengaturan penyiaran *webcam*. Ketika fungsi penyiaran diaktifkan, PiCo secara otomatis akan membuat file konfigurasi *webcam* pada Media Gateway yang digunakan untuk dikirim ke server. PiCo juga akan mengirimkan gambar-gambar yang diambil dari setiap *webcam* yang aktif ke

server secara terus-menerus dengan interval pengiriman gambar selama 10 detik. Pengiriman gambar dan data konfigurasi *webcam* Media Gateway ke *server* dilakukan melalui protokol FTP. Pengiriman data-data tersebut ke *server* dilakukan pada *thread* terpisah dengan *thread* utama sehingga pengguna tetap dapat menggunakan aplikasi ketika proses pengiriman data berlangsung. Pembuatan *thread* baru untuk transfer data ke server adalah seperti di bawah ini:

```
if (thrBroadcast == null || thrBroadcast.IsAlive == false)
{
    brc = new Broadcaster(this);
    thrBroadcast = new Thread(new ThreadStart(brc.broadcastCams));
    thrBroadcast.Start();
}
```

3.2.2. Realisasi Komponen TrackOn

TrackOn dibuat dengan menggunakan bahasa pemrograman Java yang dibangun di atas spesifikasi CLDC 1.1 dan MIDP 2.0. TrackOn dibuat agar dapat mencatat trek perjalanan seseorang serta mengambil gambar dari lingkungan yang dilewati selama perjalanan lalu mengirimkannya ke *server*.

Pada form Login terdapat *text box* untuk memasukkan nama pengguna dan kata sandinya agar dapat menggunakan aplikasi TrackOn. Proses *login* pada TrackOn dilakukan dengan cara membuka koneksi HTTP pada *handphone* dan melakukan permintaan ke *file* PHP di *server* menggunakan metode POST agar *handphone* dapat berkomunikasi dengan *server*.



Gambar 3.20. Antarmuka login pada TrackOn

Form login dibuat dengan komponen TextField dan Command pada JME, berikut adalah potongan kodingannya:

```

txtUser = new TextField("Nama:", preferences.get(preUser), 45,
TextField.ANY);
txtPass = new TextField("Kata sandi:", preferences.get(prePassword), 45,
TextField.PASSWORD);
txtServer = new TextField("Server:", preferences.get(preServer), 45,
TextField.ANY);

this.append("Login terlebih dahulu...");
this.append(txtUser);
this.append(txtPass);
this.append(txtServer);

cmdExit = new Command("Keluar", Command.EXIT, 1);
cmdStart = new Command("Login", Command.OK, 1);
this.addCommand(cmdExit);
this.addCommand(cmdStart);

this.setCommandListener(this);

```

Pada *form* pendeteksian perangkat *Bluetooth*, *TrackOn* akan secara otomatis mendaftarkan segala perangkat *Bluetooth* yang terdeteksi di sekitar *handphone* dengan radius sekitar 10 meter. Proses pendeteksian perangkat *Bluetooth* dilakukan melalui protokol BTSP dengan pengimplementasian *DiscoverListener* menggunakan paket *javax.bluetooth* (JSR 82) yang disediakan oleh CLDC 1.1. Daftar perangkat-perangkat *Bluetooth* yang ada salah satu dapat dipilih agar *TrackOn* dapat mencoba untuk membuat koneksi dengan perangkat tersebut.



Gambar 3.21. Antarmuka pemilihan perangkat GPS pada *TrackOn*

Fungsi untuk mendeteksi perangkat-perangkat *Bluetooth* yang berada di sekitar *handphone* adalah sebagai berikut:

```

private void scanDevice() {
    try {
        remoteDevices = discoverBluetoothDevices();

        if (remoteDevices != null || remoteDevices.length > 0) {

            choDevice = new ChoiceGroup("Pilih perangkat GPS",
                Choice.EXCLUSIVE);

            for (int i = 0; i < remoteDevices.length; i++) {
                choDevice.append(remoteDevices[i][0], null);
            }

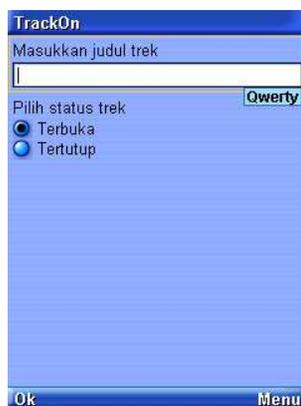
            tracker.deleteAll();
            tracker.append(choDevice);

            cmdOk = new Command("Ok", Command.OK, 1);
            tracker.addCommand(cmdOk);
        } else {
            tracker.append("Tidak ditemukan perangkat
bluetooth...");
        }

        cmdCancel = new Command("Batal", Command.CANCEL, 1);
        tracker.addCommand(cmdCancel);
    } catch (Exception e) {
        trackOn.handleException(e);
    }
}

```

Pada *form* pembuatan trek, tersedia *text box* untuk memasukkan judul trek dan pilihan untuk menentukan status trek tersebut apakah trek bersifat privat atau publik. Data yang dimasukkan pada *text box* tersebut dan pilihan status trek dikirimkan ke *server* melalui koneksi HTTP. Jika data trek baru telah berhasil diterima oleh server, server akan menyimpankannya ke *database* dan TrackOn akan mulai melakukan transaksi data dengan perangkat GPS yang telah dipilih.



Gambar 3.22. Antarmuka penampilan informasi geografis pada TrackOn

```

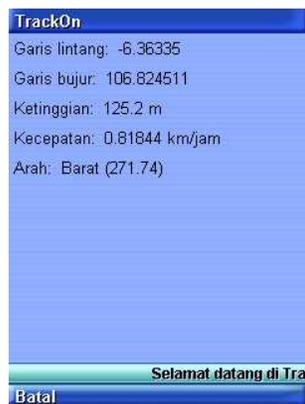
txtTitle = new TextField("Masukkan judul trek", "", 45, TextField.ANY);

choStatus = new ChoiceGroup("Pilih status trek", Choice.EXCLUSIVE);
choStatus.append("Terbuka", null);
choStatus.append("Tertutup", null);
choStatus.setSelectedIndex(0, true);

this.append(txtTitle);
this.append(choStatus);

```

Form penampilan informasi geografis akan menampilkan informasi berupa posisi koordinat bumi dari pengguna, ketinggian pengguna dari permukaan laut, kecepatan gerak pengguna, dan arah gerak pengguna. Informasi-informasi yang disajikan pada *form* tersebut diperbaharui terus menerus dengan interval waktu 2,5 detik. Pengulangan proses dilakukan dengan menggunakan *thread* terpisah.



Gambar 3.23. Antarmuka penampilan informasi geografis pada TrackOn

```

while (IsRunning) {
    strLat.setText(tempLat);
    strLong.setText(tempLong);
    strAlt.setText(tempAlt + " m");
    strSpd.setText(tempSpd + " km/jam");
    if (intHead < 23 || intHead >= 338)
        strHead.setText("Utara (" + tempHead + "°");
    else if (intHead >= 23 && intHead < 68)
        strHead.setText("Timur Laut (" + tempHead + "°");
    else if (intHead >= 68 && intHead < 113)
        strHead.setText("Timur (" + tempHead + "°");
    else if (intHead >= 113 && intHead < 158)
        strHead.setText("Tenggara (" + tempHead + "°");
    else if (intHead >= 158 && intHead < 203)
        strHead.setText("Selatan (" + tempHead + "°");
    else if (intHead >= 203 && intHead < 248)
        strHead.setText("Barat Daya (" + tempHead + "°");
    else if (intHead >= 248 && intHead < 293)
        strHead.setText("Barat (" + tempHead + "°");
    else if (intHead >= 293 && intHead < 338)
        strHead.setText("Barat Laut (" + tempHead + "°");
    thread.sleep(2500);}

```

Kesemua data tersebut didapat dari hasil pemrosesan teks NMEA yang didapat dari GPS menggunakan prosesor teks NMEA pada TrackOn. Tipe data NMEA yang diproses oleh TrackOn mencakup tipe-tipe data berikut:

Tabel 3.26. Tipe data NMEA yang diproses TrackOn

Tipe Data NMEA	Digunakan Untuk Mengambil Data
GPGSV	Jumlah satelit navigasi yang terdeteksi
GPGLL	Titik garis bujur dan garis lintang
GPRMC	Kecepatan dan arah
GPGGA	Ketinggian
GPGSA	Keseluruhan data-data mengenai satelit

Informasi geografis yang ditampilkan ke pengguna diperbaharui terus menerus pada interval waktu sebanyak 2,5 detik. Setiap terjadi pembaharuan informasi geografis, TrackOn akan menggabungkan data informasi geografis yang lama dengan yang baru pada satu kumpulan objek. Setiap 12 iterasi dari pembaharuan informasi geografis yang ditampilkan (30 detik), TrackOn akan memulai untuk mengambil gambar lingkungan sekitar pengguna melalui kamera *handphone*. Hal ini dilakukan dengan menggunakan kelas *VideoControl* pada paket *javax.microedition.media* di JME. Gambar yang didapat dari kamera diproses dengan menggunakan JPEG Encoding, Berikut adalah penggalan kode pemrograman untuk penghitungan interaksi pembaharuan informasi geografis:

```
intCounter++;
if (intCounter >= 12 && isBroadcasting == false) {
isBroadcasting = true;
new Broadcaster(tempLat, tempLong, tempAlt, tempSpd, tempHead,
bufLineString.toString());
bufLineString.setLength(0);
}
```

TrackOn kemudian akan langsung mengirimkan gambar yang telah didapat dan kumpulan objek yang berisi data-data informasi geografis ke server melalui koneksi HTTP. Ketika proses ini berjalan pengguna akan diberitahu bahwa TrackOn sedang mengirimkan data ke server melalui penggunaan *Ticker* di JME. Setelah proses pengiriman data trek dikirimkan, TrackOn akan mereset data trek pada *handphone* untuk menghemat penggunaan memori dan kemudian melanjutkan pengisian data trek dari awal.

BAB IV

PENGUJIAN DAN ANALISIS DATA

4.1. Pengujian Perangkat Lunak

4.1.1. Deskripsi Pengujian

a. Deskripsi Pengujian Komponen PiCo

Untuk memastikan bahwa program pada komponen PiCo telah berjalan dengan baik, maka harus dilakukan pengujian terhadap program. Pengujian komponen dilakukan dengan menjalankan aplikasi PiCo pada komputer yang ingin dijadikan sebagai Media Gateway lalu menyiarkan *webcam-webcam* yang terpasang pada Media Gateway ke *server* untuk kemudian dilihat melalui *website demo*.

- Target Pengujian

Pada pengujian komponen PiCo, hasil pengiriman data dari Media Gateway ke *server* diharapkan dapat menampilkan hasil pengambilan gambar oleh *webcam-webcam* yang terpasang pada Media Gateway.

- Data Lingkungan

- Lokasi : Kantin Fakultas Teknik Universitas Indonesia
- Tanggal Pelaksanaan : Jumat, 31 Juli 2009
- Pelaksana : Firman Nugraha
- Instruktur : Mera Kartika Delimayanti, Ssi., MT.

- Peralatan Pengujian



Gambar 4.1. Peralatan pengujian komponen PiCo

- Laptop
- USB webcam 2 buah

b. Deskripsi Pengujian Komponen TrackOn

Untuk memastikan bahwa program pada komponen TrackOn telah berjalan dengan baik, maka harus dilakukan pengujian terhadap program. Pengujian komponen dilakukan dengan mengaktifkan program TrackOn yang telah terpasang pada *handphone* lalu menjalankan transmisi data ke *server* dari *handphone* lalu melihat hasil akhirnya menggunakan *website demo*.

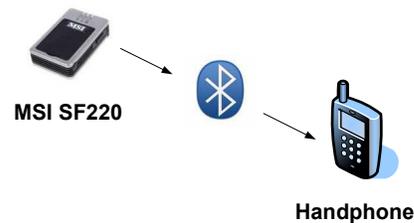
- Target Pengujian

Pada pengujian komponen TrackOn, hasil transmisi data dari *handphone* ke *server* diharapkan dapat menghasilkan trek perjalanan sesuai dengan trek perjalanan yang dilakukan dan dapat menampilkan gambar-gambar yang didapat selama perjalanan menggunakan *website demo*.

- Data Lingkungan

- Lokasi : Lingkungan Fakultas Teknik dan Fakultas Ekonomi Universitas Indonesia
- Tanggal Pelaksanaan : Jumat, 31 Juli 2009
- Pelaksana : Firman Nugraha
- Instruktur : Mera Kartika Delimayanti, Ssi., MT.

- Peralatan Pengujian



Gambar 4.2. Peralatan pengujian komponen TrackOn

- Handphone Nokia 3110c (koneksi internet yang didukung GPRS dan EDGE)
- Kartu SIM Indosat Mentari
- GPS Bluetooth MSI SF220

4.1.2. Prosedur Pengujian

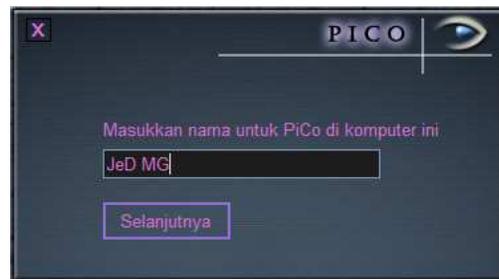
a. Prosedur Pengujian Komponen PiCo

1. Pasangkan 2 USB *webcam* slot USB yang tersedia pada *laptop*
2. Jalankan aplikasi PiCo
3. Masukkan data nama pengguna “jed” dan kata kunci “password” pada form login



Gambar 4.3. Pemasukan data pengguna pada form login PiCo

4. Masukkan nama Media Gateway “JeD MG”



Gambar 4.4. Pemasukan nama Media Gateway pada form pembuatan Media Gateway PiCo

5. Masukkan koordinat bumi dimana Media Gateway berada dengan garis lintang - 6.5639114379883 dan garis bujur 106.79278564453



Gambar 4.5. Pemasukan lokasi Media Gateway pada form lokasi Media Gateway PiCo

6. Pilih *webcam* yang ingin disiarkan ke *server* dengan mencentang di box yang tersedia di setiap *webcam* pada form
7. Jalankan fungsi penyiaran *webcam*

8. Lihat hasil penyiaran *webcam* dari PiCo ke server menggunakan website demo

b. Prosedur Pengujian Komponen TrackOn

1. Nyalakan GPS di luar ruangan dan diamkan selama 30 detik
2. Pastikan koneksi Bluetooth *handphone* aktif dan set *visibility handphone* ke *hidden*
3. Pastikan koneksi internet pada *handphone* aktif
4. Set aplikasi TrackOn pada *handphone* agar dapat memiliki otoritas untuk mengakses kamera, Bluetooth, dan GPRS
5. Jalankan aplikasi TrackOn pada *handphone*
6. Masukkan nama “jed” dan kata sandi “password” agar dapat *login* ke sistem

Gambar 4.6. Pemasukan data pengguna pada form login TrackOn

7. Pilih perangkat GPS pada daftar

Gambar 4.7. Pemilihan perangkat GPS pada form daftar perangkat TrackOn

8. Masukkan judul trek “Tes TrackOn” dan set status trek ke public



Gambar 4.8. Pengisian nama trek pada form pembuatan trek TrackOn

9. Lakukan perjalanan dengan berjalan kaki di sekeliling Fakultas Teknik dan Fakultas Ekonomi UI dengan rute perjalanan sebagai berikut:
 - a. Kantin FTUI
 - b. Lantai 1 Gedung Engineering Center FTUI
 - c. Lantai 3 Gedung Engineering Center FTUI (menggunakan lift dari lantai 1)
 - d. Kantin FEUI
 - e. Aula FEUI
 - f. Lapangan parkir motor FTUI (melalui jalan raya Universitas Indonesia)
 - g. Aula FTUI
 - h. Kantin FTUI
10. Amati hasil pencatatan trek beserta gambar-gambar yang diambil selama perjalanan dengan menggunakan website demo
11. Klik pada masing-masing pin berwarna biru untuk melihat hasil pengambilan gambar selama perjalanan
12. Ulangi langkah 1 sampai 7
13. Masukkan judul trek
14. Lakukan pengujian di 3 tempat berikut:
 - a. Belakang Lab Telkom PNJ
 - b. Samping Lab Telkom PNJ
 - c. Depan Lab Telkom PNJ
15. Catat tipe koneksi jaringan yang ada pada *handphone* setiap 1 menit sekali sebanyak 5 kali
16. Catat total data yang dikirimkan ke server pada menit ke 5
17. Ulangi langkah 12 sampai 16 sebanyak 2 kali

4.1.3. Data Hasil Pengujian

a. Data Hasil Pengujian Komponen PiCo

Ketika aplikasi PiCo dijalankan ia langsung mengecek file konfigurasi Media Gateway yang ada. Berhubung PiCo baru pertama kali dijalankan file tersebut belum ada dan PiCo secara otomatis membuat file tersebut.



Gambar 4.9. Pembuatan file konfigurasi Media Gateway pada PiCo

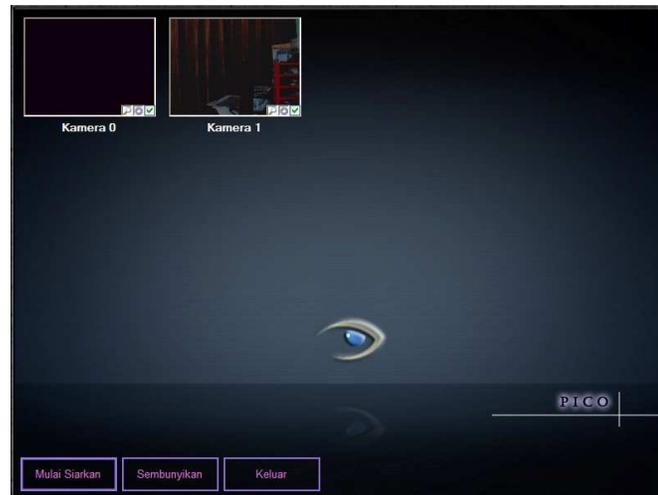
Ketika aplikasi PiCo dijalankan dengan mengikuti langkah pengujian yang telah disebutkan sebelumnya, PiCo berhasil mengirimkan data dari Media Gateway ke server secara terus menerus selama status penyiaran masih aktif.

Setelah itu data login dimasukkan dan PiCo langsung meminta pengguna untuk memasukkan nama Media Gateway dan lalu koordinat bumi dari Media Gateway tersebut. Data yang dimasukkan oleh pengguna langsung dimasukkan ke *database* di *server*.

id	id_user	name	.astext(location)
2	1	Duduk	POINT(-6.5639100074768 106.79276275635)
3	1	Testing	POINT(1 2)
4	3	JeD MG	POINT(-6.5639114379883 106.79278564453)

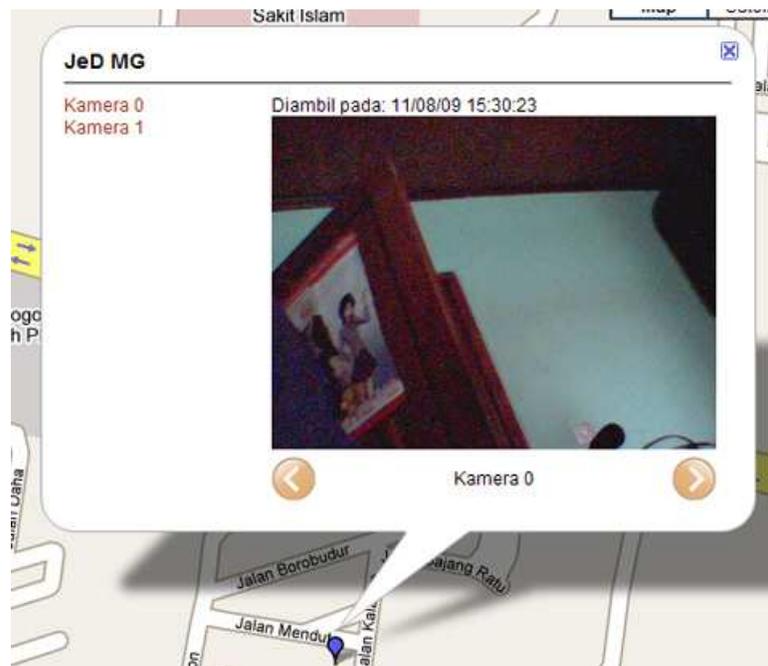
Gambar 4.10. Data Media Gateway baru pada PiCo

Setelah proses otentikasi dan pembuatan Media Gateway, PiCo langsung mendeteksi *webcam-webcam* yang terpasang dan lalu menampilkan hasil pengambilan gambar dari masing-masing *webcam* tersebut.



Gambar 4.11. Pendaftaran webcam-webcam pada PiCo

Setelah interval yang telah ditentukan, PiCo mengambil gambar dari semua kamera *webcam* yang dimasukkan ke dalam daftar penyiaran ke server. Hasil pengiriman data dari Media Gateway ke server dilihat melalui web dengan mencari posisi Media Gateway tersebut pada peta.

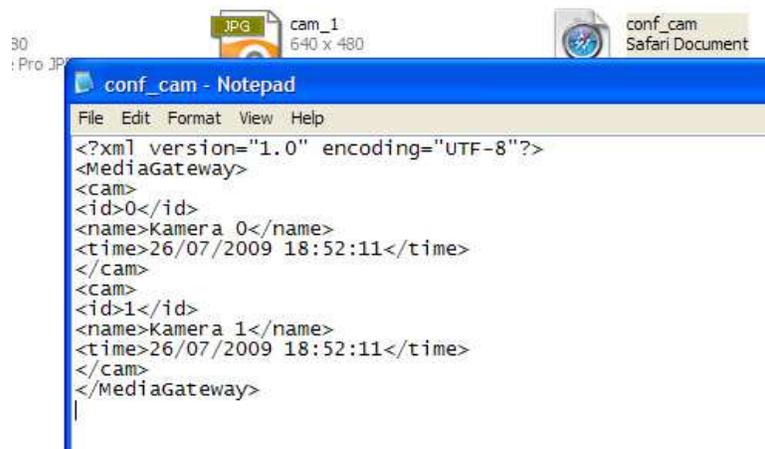


Gambar 4.12. Hasil penyiaran webcam oleh PiCo pada website demo

File-file yang dikirimkan dari PiCo disimpan pada server dengan hasil sebagai berikut:



Gambar 4.13. File-file PiCo di server



Gambar 4.14. File *conf_cam.xml* Media Gateway PiCo di server

Besar data dari masing-masing gambar adalah 36,8 dan 36,2 kilobyte.

b. Data Hasil Pengujian Komponen TrackOn

Ketika aplikasi TrackOn dijalankan dengan mengikuti langkah pengujian yang telah disebutkan sebelumnya, TrackOn berhasil mendapatkan data posisi koordinat bumi dari pengguna secara cukup tepat. Pada saat pengguna dalam kondisi tidak bergerak, TrackOn menampilkan informasi yang tidak akurat mengenai kecepatan dan arah perjalanan pengguna.

Setelah aplikasi TrackOn berjalan untuk beberapa waktu, ia mulai mengambil gambar dari lingkungan yang berada di lingkup penglihatan kamera *handphone* dan memberikan bunyi yang umumnya terdengar ketika seseorang memotret menggunakan kamera *handphone*. Bunyi tersebut juga menandakan bahwa TrackOn akan memulai proses pengiriman data-data informasi geografis yang telah dikumpulkan beserta gambar yang ditangkap ke server.

Selama pengujian berlangsung, tidak terjadi *error* pada aplikasi TrackOn dan penggunaan memori *handphone* paling besar yang tercatat adalah sebesar 776428 KB. Dari pengujian yang telah dilakukan, berikut adalah hasil akhir data yang didapat dari TrackOn beserta tampilannya pada website demo:

- Jarak total perjalanan : 1,832 km
- Lama perjalanan : 0 jam, 16 menit, 13 detik
- Kecepatan rata-rata : 0,678 km/jam
- Waktu mulai perjalanan : 2009-07-31 10:18:07
- Waktu akhir perjalanan : 2009-07-31 10:34:20



Gambar 4.15. Antarmuka penampikan informasi geografis pada TrackOn

Setiap pin biru pada peta berisi gambar yang diambil oleh TrackOn pada titik tersebut, berikut adalah contoh gambar yang diambil oleh TrackOn pada pengujian ini:



Gambar 4.16. Hasil pengambilan gambar selama perjalanan

Di pengujian tahap selanjutnya di 3 tempat yang telah disebutkan sebelumnya, didapat data-data sebagai berikut:

Lokasi Pengetesan : Belakang Lab Telkom PNJ

Tabel 4.1. Hasil pengujian TrackOn terhadap kondisi jaringan pada lokasi pertama

Iterasi ke	Kekuatan Sinyal Jaringan Internet Menit ke					Jumlah Data
	1	2	3	4	5	
1	3G Full	3G Full	3G Full	3G Full	3G Full	7
2	3G Full	3G Full	3G Full	3G Full	3G Full	7
3	3G Full	3G Full	3G Full	3G Full	3G Full	7

Lokasi Pengetesan : Depan Lab Telkom PNJ

Tabel 4.2. Hasil pengujian TrackOn terhadap kondisi jaringan lokasi kedua

Iterasi ke	Kekuatan Sinyal Jaringan Internet Menit ke					Jumlah Data
	1	2	3	4	5	
1	EDGE Full	EDGE Full	3G Full	3G Full	3G Full	7
2	3G Full	GPRS Full	GPRS Full	GPRS Full	GPRS Full	7
3	3G Full	3G Full	3G Full	3G Full	3G Full	6

Lokasi Pengetesan : Samping Lab Telkom PNJ

Tabel 4.3. Hasil pengujian TrackOn terhadap kondisi jaringan lokasi ketiga

Iterasi ke	Kekuatan Sinyal Jaringan Internet Menit ke					Jumlah Data
	1	2	3	4	5	
1	3G Full	3G Full	3G Full	3G Full	3G Full	7
2	3G Full	3G Full	3G Full	3G Full	3G Full	7
3	3G Full	3G Full	3G Full	3G Full	3G Full	6

Tes yang dilakukan dengan menjalankan aplikasi trackOn selama lima menit yang di tiap menitnya dilakukan pencatatan data aplikasi dalam satu menit tersebut. Pengujian ini dilakukan sebanyak tiga kali untuk menyakinkan bahwa data valid dan juga untuk pengujian ketahanan aplikasi jika dipakai dalam periode waktu tertentu. Kemudian dilakukan pengetesan di beberapa tempat yang berbeda untuk mendapatkan referensi lain untuk pengujian data.

4.2. Analisis Data Hasil Pengujian

a. Analisis Data Hasil Pengujian Komponen PiCo

Dari data hasil pengujian PiCo yang telah dipaparkan sebelumnya, dapat dilihat bahwa data-data yang berada di server sesuai dengan yang telah dimasukkan. Data-data yang dimasukkan juga sesuai dengan standar spesifikasi data keluaran sistem yang telah dijelaskan di bagian perancangan. Seluruh proses dapat dijalankan tanpa *error* mulai dari login, pembuatan konfigurasi Media Gateway, pembuatan Media Gateway baru, hingga penyiaran *webcam*.

b. Analisis Data Hasil Pengujian Komponen TrackOn

Inisialisasi GPS Bluetooth dilakukan diluar ruangan dan memerlukan pengguna untuk diam beberapa waktu hal ini dikarenakan GPS membutuhkan ruang yang luas dengan interferensi minimal seperti gedung-gedung dan membutuhkan beberapa waktu untuk kalibrasi data yang di dapat dari sinyal satelit GPS agar sesuai dengan lingkungan di mana pengguna berada.

TrackOn bekerja dengan menggunakan *handphone* sebagai pusat pemrosesan data yang ditangkap dari GPS Bluetooth dan kamera *handphone* lalu data-data tersebut dikirimkan ke server melalui protokol HTTP. Transfer data ke server dilakukan melalui protokol HTTP. Protokol HTTP digunakan karena didukung oleh kebanyakan *handphone* sedangkan untuk GPS dipilih yang model Bluetooth karena GPS tipe ini paling umum digunakan berbarengan dengan *handphone*.

Pemrosesan data yang didapat dari GPS Bluetooth dan kamera *handphone* dilakukan oleh program yang dibuat dengan menggunakan bahasa Java pada platform J2ME.

Data yang didapat server dari TrackOn tidak selalu akurat terutama jika pengguna ketika menjalankan aplikasi trackon untuk membuat trek melewati jalan-jalan yang memiliki banyak halangan seperti gedung-gedung. Pada gambar hasil keluaran TrackOn pada website demo yang disajikan sebelumnya, bisa dilihat bahwa ada beberapa bagian trek yang tidak normal.



Gambar 4.17. Antarmuka penampikan informasi geografis pada TrackOn

2 bagian dari trek hasil pengetesan menunjukkan jalur yang abnormal, hal ini dikarenakan pada bagian 1 pengetes berjalan masuk ke gedung Engineering Center FTUI dan menggunakan lift untuk naik ke lantai 3 gedung Engineering Center. Pada kondisi tersebut sinyal GPS akan melemah terutama ketika memasuki lift. Pada bagian 2 sinyal GPS melemah karena banyaknya interferensi dari sinyal-sinyal wi-fi yang ada pada aula FTUI ditambah dengan pengetes saat itu berada di dalam aula sehingga sinyal GPS terhalang. Pada kedua bagian tersebut sinyal GPS menjadi lemah sehingga menyebabkan GPS Bluetooth yang digunakan tidak dapat membaca posisi pengetes secara tepat. Pin-pin biru yang ada pada gambar menandakan letak dimana proses pengiriman data dari TrackOn ke server dilakukan dan juga posisi dimana gambar lingkungan pengetes diambil oleh TrackOn melalui kamera *handphone*.

Pengetesan tahap selanjutnya yang dilakukan ini merupakan tes ketahanan dari aplikasi TrackOn dengan kondisi jaringan internet yang ada pada di 3 lokasi yang telah disebutkan. Pengujian ini perlu dilakukan untuk melihat kelayakan pakai dari aplikasi ini dengan pengujian di beberapa kondisi tertentu dan juga dalam periode waktu yang telah ditentukan.

Hasil dari pengujian ini akan memperlihatkan tingkat keberhasilan dan juga tingkat kegagalan dari aplikasi TrackOn pada implementasi nyata. Sehingga dapat diambil kesimpulan apakah aplikasi ini layak dipakai untuk implementasi nyata atau tidak. Berikut adalah kesimpulan dari data yang telah didapat dalam pengujian ke dua ini:

TrackOn berjalan optimal pada jaringan 3G, EDGE dan GPRS

TrackOn dikatakan optimal karena jika dilihat dari hasil pengujian trackOn dapat mengirimkan rata - rata tujuh data secara konsisten di setiap iterasinya. Ini membuktikan bahwa dengan apapun jenis jaringannya asalkan dengan sinyal yang baik maka pengguna akan dapat memakai TrackOn untuk waktu yang lama dan dengan pengiriman data yang lancar.

Transfer data TrackOn lancar dan konsisten pada jaringan 3G, EDGE dan GPRS

Transfer data pada TrackOn berjaan lancar dan juga konsisten karena rata-rata pengiriman gambar sebanyak tujuh buah gambar di berbagai macam jaringan dengan sinyal yang bagus. Ada beberapa poin yang mendukung hal ini diantara lain adanya jeda waktu di setiap pengiriman data di trackon sehingga data dapat terkirim dengan konsisten, ukuran data yang dikirimkan tidak terlalu besar sehingga transfer data lancar. Jenis data yang dikirimkan berupa gambar sehingga mudah metode pengiriman datanya dan tidak memakan tempat *bandwidth* dari jaringan yang sedang dipakai.

Perubahan jaringan antara 3G, EDGE dan GPRS tidak mempengaruhi transfer data TrackOn

Dengan adanya perubahan-perubahan jaringan yang digunakan saat TrackOn dijalankan tidak mempengaruhi kelancaran dan juga konsistensi dari TrackOn itu sendiri, karena dapat dilihat data tetap terkirim sebanyak 7 buah walaupun ada dua kali percobaan yang hanya menghasilkan 6 data terkirim.

BAB V

PENUTUP

5.1. Kesimpulan

Berikut adalah kesimpulan-kesimpulan yang didapat:

1. Komponen PiCo dapat mengirimkan data ke *server* secara terus menerus tanpa menghabiskan banyak *bandwith* internet.
2. TrackOn berjalan optimal pada jaringan 3G, EDGE dan GPRS.
3. Transfer data TrackOn lancar dan konsisten pada jaringan 3G, EDGE dan GPRS.
4. Perubahan jaringan antara 3G, EDGE dan GPRS tidak mempengaruhi transfer data TrackOn.

5.2. Saran

Berikut ini adalah saran-saran yang dapat diberikan untuk implementasi dan pengembangan lebih lanjut dari sistem ini:

1. Tingkat akurasi data GPS dari TrackOn akan semakin tinggi bila menggunakan GPS *Bluetooth* yang radius ketepatannya tinggi.
2. Kualitas gambar yang dimiliki oleh TrackOn akan semakin tajam seiring dengan perkembangan dari kamera *handphone*, yang sekarang memiliki resolusi yang cukup besar.
3. Dengan munculnya *handphone* yang sudah memiliki GPS didalamnya maka TrackOn akan semakin mudah dan praktis digunakan.
4. Dengan teknologi *webcam* yang terus berkembang, diharapkan dapat membuat *webcam* tersebut bekerja secara dinamis (dapat digerakan secara otomatis) dan memiliki resolusi gambar yang lebih baik.

DAFTAR PUSTAKA

- Anonim. 5 September 2009, Apa Itu Java?, <http://www.zahroel.co.cc/2008/12/apa-itu-java.html>
- Katriena, Flory. 5 September 2009. Apa itu Linux?. <http://pemula.linux.or.id/intro/apa.html>
- MySQL AB. 5 September 2009. MySQL 5.1 Reference Manual. <http://dev.mysql.com/doc/refman/5.1/en/>
- Supono. 5 September 2009. Apa Itu PHP?. <http://supono.wordpress.com/2006/09/14/apa-itu-php/>
- Telecom Education. 5 September 2009. Apa Itu GPS?. <http://telecomeducation.blogspot.com/2009/06/apa-itu-gps.html>
- Wikipedia. 5 September 2009. .NET Framework. http://id.wikipedia.org/wiki/.NET_Framework
- Wikipedia. 5 September 2009. Apache HTTP Server. http://en.wikipedia.org/wiki/Apache_HTTP_Server
- Wikipedia. 5 September 2009. Google Maps. <http://en.wikipedia.org/wiki/Google>