

JURUSAN ILMU KOMPUTER/INFORMATIKA
FAKULTAS SAINS DAN MATEMATIKA
UNIVERSITAS DIPONEGORO

2013

**INTERACTIVE HOTSPOT BILLING SYSTEM MENGGUNAKAN METODE
PROTOYPING BERBA DAN SMS GATEWAY
STUDI KASUS FAKUL S DAN MATEMATIKA
UNIVERSITAS DIPONEGORO**

Moch. Amri Wildan¹, Eko Adi Sarwoko², dan Helmie Arif Wibawa³

Jurusan Ilmu Komputer/Informatika
Fakultas Sains dan Matematika
Universitas Diponegoro
amri@undip.ac.id

ABSTRACT

The number of internet users academics Faculty of Science and Mathematics University of Diponegoro via wireless is increasing, therefore it is necessary to Hotspot Billing can help network administrators to manage. Hotspot Billing is interactive, so users do not have contact hotspot with the administrator directly. Users can manage their account, obtain statistical information, and can complain to an administrator with easy access via the web and sms. This Software was built using a programming language Hypertext Preprocessor (PHP) and MySQL database management system that is integrated with the Remote Authentication Dial-In User Service (RADIUS) Server, and supported by the SMS Gateway.

Keyword : *RADIUS, SMS Gateway, billing hotspot*

1. Pendahuluan

Fakultas Sains dan Matematika Universitas Diponegoro (FSM UNDIP) adalah salah satu fakultas yang sangat memperhatikan perkembangan teknologi informasi, berbagai inovasi terus ditingkatkan untuk meningkatkan layanan kepada segenap civitas akademika. Salah satu layanan yang sangat penting bagi civitas akademika adalah internet, dengan adanya fasilitas internet yang baik maka diharapkan kegiatan belajar mengajar akan menjadi lebih baik lagi terutama untuk menambah *literature* selain dari buku perpustakaan.

Tuntutan mobilitas yang tinggi saat ini bagi civitas akademika menjadikan teknologi *wireless/nirkabel* menjadi

kebutuhan utama, sehingga pengguna bisa dengan mudah terkoneksi dengan jaringan internet kampus dengan berbagai *device* seperti *Notebook, Smartphone* dan PC tablet, sehingga akan sangat memudahkan dalam mengakses internet.

Jumlah pengguna internet nirkabel yang semakin bertambah tentunya akan menyulitkan bagi administrator jaringan dalam hal pengelolaannya. Mengingat bahwa pengguna *hotspot* saat ini bukan hanya civitas akademika FSM UNDIP saja, namun orang di luar UNDIP dapat dengan bebas mengakses hotspot, disebabkan *coverage/jangkauan* akses yang tersedia tidak hanya di dalam gedung dan ruang kuliah saja, akan tetapi terjangkau pula di taman kampus sekitar FSM UNDIP,

sehingga dibutuhkan suatu *policy* pembatasan akses dengan metode otentikasi pengguna. Dibutuhkan suatu perangkat lunak untuk mengelola pengguna *wireless/hotspot* di FSM UNDIP yaitu sebuah *Billing Hotspot*. Hal inilah yang mendorong penulis untuk membuat sebuah aplikasi *Billing Hotspot* berbasis Web yang interaktif sehingga memudahkan pengguna dalam mengelola akun masing-masing dengan tidak sepenuhnya administrator jaringan yang harus menangani hal seperti ketika pengguna lupa password dan juga ingin mengetahui sudah berapa *traffic* data yang mereka gunakan. Sistem yang penulis kembangkan adalah merujuk dari yang sudah pernah ada yaitu *Billing Hotspot* dengan nama *easyhotspot* yang bisa diunduh di situs <http://easyhotspot.inov.asia>, hanya saja sistem tersebut belum mendukung sisi interaktif dengan pengguna. Selain itu sistem ini nantinya akan dapat membantu fakultas untuk mendapatkan laporan data statistik pengguna *hotspot*.

2. Tinjauan Pustaka

Tinjauan pustaka menjelaskan beberapa literatur mengenai Jaringan *Wireless*, Sistem *Billing*, *Remote Authentication Dial-In User Service* (RADIUS), *SMS Gateway*, *PHP*, *MySQL*, dan *Prototype Model* dalam pengembangan *Interactive Billing Hotspot* di FSM Universitas Diponegoro.

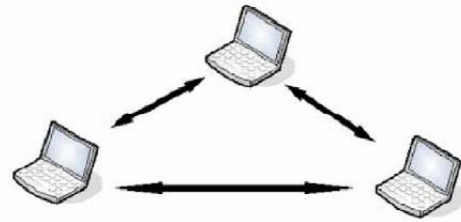
5.1. Jaringan Wireless

Jaringan WLAN (*Wireless Local Area Network*) merupakan salah satu bentuk jaringan *wireless*. Jaringan WLAN adalah jaringan yang memungkinkan dua mesin atau lebih untuk berkomunikasi menggunakan protokol jaringan standar tetapi tanpa menggunakan media transmisi kabel. Media transmisi yang digunakan komunikasi pada jaringan WLAN adalah gelombang elektromagnetik yang dapat berupa sinar infra-merah (*infrared, IR*), gelombang mikro (*microwave*) atau gelombang radio (*radio frequency, RF*) [2].

Mode Jaringan WLAN, antara lain [2]:

1. Mode Ad-Hoc

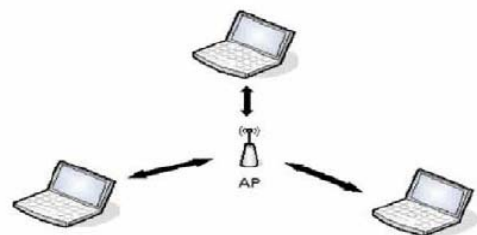
Mode Ad-hoc sering disebut sebagai jaringan *peer to peer* atau disebut juga jaringan *point to point*. *Mode Ad-hoc* memungkinkan hubungan antar komputer pada jaringan WLAN tanpa melalui suatu *access point* seperti pada gambar 2.1.



Gambar 2.1 Mode Ad-Hoc [2]

2. Mode Infrastruktur

Untuk menghubungkan banyak komputer jaringan WLAN harus dijalankan menggunakan mode Infrastruktur. Pada mode Infrastruktur diperlukan peralatan tambahan berupa WAP (*wireless access point*) atau disebut secara singkat dengan *access point*. *Access point* berlaku seperti *hub* atau *switch* pada jaringan kabel, sehingga *access point* akan menjadi pusat dari jaringan WLAN seperti pada gambar 2.2



Gambar 2.2 Mode Infrastruktur [2]

5.2. Sistem Billing

Billing berasal dari bahasa Inggris yaitu *bill* (noun), yang artinya bukti transaksi pembayaran. Maka *billing* (adv) dapat juga diartikan mengirimkan bukti transaksi, atau mengumumkan bukti transaksi [3].

Sistem billing merupakan sistem yang membantu para usahawan untuk mengatur dan mencatat segala transaksi yang terjadi. Contohnya bagi pengusaha warung internet, sistem billing digunakan untuk memonitor penggunaan dan pemasukan warnetnya. Sedangkan jika bagi usahawan di bidang rumah sakit sistem billing digunakan untuk mencatat proses pelayanan, mulai pasien datang sampai dengan pasien pulang. Menghitung biaya yang harus dibayar pasien secara otomatis, serta memberikan informasi sebagai analisa pengambilan keputusan secara cepat dan akurat [3].

5.3. *Remote Authentication Dial-In User Service (RADIUS)*

RADIUS adalah sebuah protokol keamanan komputer yang digunakan untuk melakukan otentikasi, otorisasi, dan pendaftaran akun pengguna secara terpusat untuk mengakses jaringan. RADIUS didefinisikan di dalam RFC 2865 dan RFC 2866, yang pada awalnya digunakan untuk melakukan otentikasi terhadap akses jaringan secara jarak jauh dengan menggunakan koneksi dial-up. RADIUS, kini telah diimplementasikan untuk melakukan otentikasi terhadap akses jaringan secara jarak jauh dengan menggunakan koneksi selain dial-up, antara lain seperti VPN (Virtual Private Networking), access point nirkabel, switch Ethernet [2].

Server RADIUS menyediakan mekanisme keamanan dengan menangani otentikasi dan otorisasi koneksi yang dilakukan pengguna. Pada saat computer client akan menghubungkan diri dengan jaringan, server RADIUS akan meminta identitas pengguna (username dan password) untuk kemudian dicocokkan dengan data yang ada dalam *database* server RADIUS untuk kemudian ditentukan apakah pengguna diijinkan untuk menggunakan layanan dalam jaringan komputer. Jika

proses otentikasi dan otorisasi berhasil maka proses pelaporan dilakukan, yakni dengan mencatat semua aktifitas koneksi pengguna, menghitung durasi waktu dan jumlah transfer data yang dilakukan oleh pengguna. Proses pelaporan yang dilakukan server RADIUS bisa dalam bentuk waktu (detik, menit, jam) maupun dalam bentuk besar transfer data (*Byte, KByte, Mbyte*) [2].

5.4. *SMS Gateway*

5.4.1. *SMS (Short Message Services)*

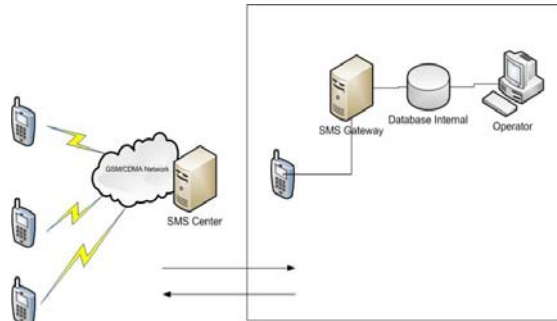
Layanan SMS merupakan sebuah layanan yang bersifat non *real time* dimana sebuah pesan pendek dapat dikirim ke suatu tujuan, tidak peduli apakah tujuan tersebut aktif atau tidak. Bila dideteksi bahwa tujuan tidak aktif, maka sistem akan menunda pengiriman ke tujuan hingga tujuan aktif kembali. Pada dasarnya sistem SMS akan menjamin pengiriman dari suatu pesan pendek hingga sampai ke tujuan.

Layanan SMS dibangun dari berbagai entitas yang saling terkait dan mempunyai fungsi dan tugas masing - masing. Tidak ada satu pun dalam sistem SMS yang dapat bekerja secara parsial. Entitas dalam jaringan SMS ini disebut juga elemen jaringan SMS.

Beberapa perusahaan menggunakan layanan SMS ini untuk meningkatkan proses bisnisnya terutama mengenai hubungan dengan customer, dan keperluan komunikasi. Karena panjang data dari SMS yang bersifat kecil, maka pengolahan data SMS tersebut dapat dilakukan secara cepat sehingga meningkatkan efisiensi pengolahan data [9].

5.4.2. *SMS Gateway*

SMS Gateway adalah suatu *platform* yang menyediakan mekanisme untuk mengirim dan menerima SMS dari peralatan *mobile* (*HP, PDA phone*, dan lain-lain) [9]. Arsitektur sistem SMS Gateway bisa dilihat pada gambar 2.3.



Gambar 2.3 Arsitektur SMS Gateway [9]

5.4.3. Gammu

Gammu adalah nama proyek serta nama utilitas baris perintah yang dapat Anda gunakan untuk mengontrol telepon Anda. Hal ini ditulis dalam C dan dibangun di atas *LibGammu* [1].

Utilitas *Command Line Gammu* menyediakan akses ke berbagai fitur telepon, namun tingkat dukungan berbeda dari ponsel ke ponsel [1].

Umumnya fitur berikut ini didukung [1]:

- Daftar panggil, memulai dan penanganan
- SMS* pengambilan, backup dan pengiriman
- MMS* pengambilan
- Daftar buku telepon, ekspor dan impor (juga dari format standar seperti *vCard*)
- Kalender dan daftar tugas, ekspor dan impor (juga dari format standar seperti *vCalendar* atau *iCalendar*)
- Pengambilan informasi telepon dan jaringan
- Akses ke sistem telepon file (catatan bahwa beberapa ponsel bekerja juga sebagai perangkat penyimpanan *USB* dan mereka tidak dapat diakses melalui *Gammu*)

Proyek ini awalnya bercabang dari *Gnokii* dan sampai versi 0,58 telah bernama *MyGnokii2*. Kemudian perlu untuk nama yang lebih baik tiba dan *Gammu* terpilih sebagai akronim untuk *Gammu* Semua

Utilitas Manajemen *Mobile*, tanpa mengetahui bahwa *Gammu* dari "*Heretics dari Dune*" buku yang ditulis oleh *Frank Herbert* ada [1].

Paket *Gammu* tidak termasuk hanya biner ini, tetapi juga *Gammu SMS Daemon*, *Gammu* perpustakaan dan *Python Binding* yang dapat digunakan untuk mengembangkan aplikasi sendiri mengakses ponsel [1].

5.5. PHP

PHP adalah bahasa *scripting* yang menjadi satu dengan HTML yang dijalankan pada *serverside*, artinya adalah semua perintah yang diberikan akan secara penuh dijalankan pada *server* sedangkan yang dikirimkan ke *client (browser)* hanya berupa hasilnya saja. Kebanyakan sintaks pada PHP mengadopsi sintaks bahasa pemrograman C, JAVA, PERL dengan pasangan fitur khusus untuk PHP yang bersifat unik [4].

5.6. MySQL

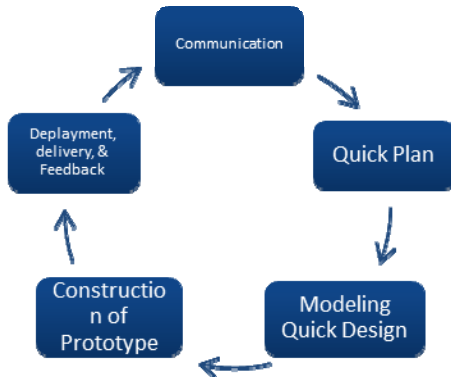
MySQL adalah Relational Database Management System (RDBMS) yang didistribusikan secara gratis di bawah lisensi General Public License (GPL), sehingga setiap orang bebas untuk menggunakan MySQL. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu Structured Query Language (SQL). SQL adalah sebagai bahasa query yang tidak hanya terbatas digunakan untuk query ke dalam basis data, tetapi dapat digunakan untuk mendefinisikan struktur data, memodifikasi data, dan menspesifikasikan security constraint [10].

MySQL mendukung hampir semua bahasa pemrograman populer saat ini seperti C, C++, Java, Perl, PHP, Python, dan lain-lain [7].

5.7. Prototype Model

Model proses perangkat lunak yang digunakan sebagai pedoman untuk membangun sistem adalah *prototype* model. Pengembangan *prototype* yang

cepat penting dilakukan untuk membangun sistem agar biaya terkontrol dan pengguna dapat bereksperimen dengan *prototype* pada tahap awal proses perangkat lunak [6]. Siklus hidup perangkat lunak digambarkan pada gambar 2.1



Gambar 2.4. Siklus Hidup Perangkat Lunak Prototype Model

2.7.1. Communication

Proses mengumpulkan informasi kebutuhan sistem/perangkat lunak melalui komunikasi dengan pengguna sistem. Tahap ini mendefinisikan secara rinci mengenai fungsi-fungsi, batasan dan tujuan dari perangkat lunak sebagai spesifikasi sistem yang akan dibuat [8].

2.7.2. Quick plan and Modeling Quick Design

Tahap *quick plan design* adalah tahap pembuatan ERD (*Entity Relationship Diagram*) dan DFD (*Data Flow Diagram*). ERD untuk menggambarkan entitas-entitas dan relasi antar entitas tersebut dalam model basis data relasional, sedangkan DFD memberi gambaran bagaimana data berubah sejalan dengan alirannya dalam sistem dan menggambarkan fungsi-fungsi yang mengubah data-data tersebut. Tahap *modeling quick design* adalah tahap perancangan perangkat lunak terdiri dari empat atribut yaitu struktur data, arsitektur perangkat lunak, antarmuka pengguna, dan detail

prosedural. Perancangan perangkat lunak melibatkan identifikasi dan deskripsi abstrak sistem perangkat lunak yang mendasar dan hubungan-hubungannya [8].

2.7.3. Construction of Prototype

Pada tahap ini, perancangan perangkat lunak direalisasikan dengan dibuatnya serangkaian *prototype* program.

2.7.4. Deployment, Delivery and Feedback

Pada tahap ini, *prototype* yang sudah dibuat diserahkan dan dievaluasi oleh pengguna. Pada tahap ini juga terjadi umpan balik dari pengguna apabila ada penambahan kebutuhan. Di sinilah terjadi suatu proses awal untuk iterasi berikutnya hingga user merasa telah terpenuhi semua kebutuhannya [8].

5.8. Black-box Testing

Black-box testing, bisa juga disebut *behavioral testing*, berfokus pada *functional requirement* sebuah perangkat lunak [8]. *Black-box testing* merupakan pendekatan komplementer yang memungkinkan untuk mengungkap beberapa kesalahan.

3. Analisis dan Perancangan

Tahap ini menjelaskan proses pembangunan perangkat lunak yaitu analisis dan perancangan. Kedua tahapan tersebut merupakan bagian dari pengembangan perangkat lunak dengan model *prototype*.

3.1. Communication

3.1.1. Gambaran Umum

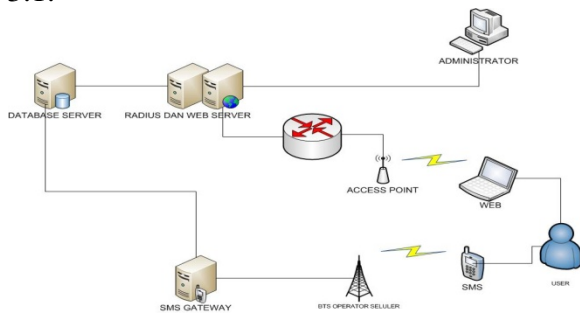
Sistem ini bertujuan untuk memudahkan pengelolaan *hotspot* di Fakultas Sains dan Matematika Universitas Diponegoro yang menekankan interaktifitas antara *administrator* jaringan dengan pengguna *hotspot*. Dengan *Billing Hotspot* aktifitas pengguna *hotspot* akan diatur, sehingga hanya pengguna yang mempunyai akun yang dapat terkoneksi dengan jaringan internet. Aktifitas pengguna juga tercatat dalam database, yaitu *bandwidth usage*,

waktu akses, dan berapa jumlah pengguna yang terkoneksi.

Pengguna dari sistem ini adalah:

- 1) Admin merupakan pihak yang dapat mengelola data user *hotspot*, melakukan pemeliharaan sistem, melihat laporan statistik secara detail dan mereview keluhan yang disampaikan user melalui sistem.
- 2) User merupakan pihak yang dapat mengubah data diri, meminta *reset password*, melihat data statistik penggunaan dirinya sendiri, dan dapat menyampaikan keluhan melalui sistem.

Dengan sistem ini diharapkan interaksi berlebih antara user dengan administrator jaringan menjadi berkurang. Interaksi user dan administrator akan jauh berkurang dengan adanya sistem yang akan dikembangkan. Arsitektur sistem yang akan dikembangkan ditunjukkan pada gambar 3.1.



Gambar 3.1 Arsitektur Sistem Baru

3.1.2. Spesifikasi Kebutuhan Perangkat Lunak

Tahap ini melakukan komunikasi dengan Unit Pengelolaan dan Pelayanan Teknologi Informasi Fakultas Sains dan Matematika Universitas Diponegoro terkait dengan batasan permasalahan dan spesifikasi perangkat lunak. Spesifikasi kebutuhan perangkat lunak dijelaskan pada tabel 3.1.

NO	SRS ID	Deskripsi
1.	SRS-IBS-F01	Melakukan otentikasi admin

2.	SRS-IBS-F02	Melakukan otentikasi user
3.	SRS-IBS-F03	Mengelola data user
4.	SRS-IBS-F04	Menampilkan laporan data statistik
5.	SRS-IBS-F05	Mengelola keluhan user
6.	SRS-IBS-F06	Melakukan reset password user

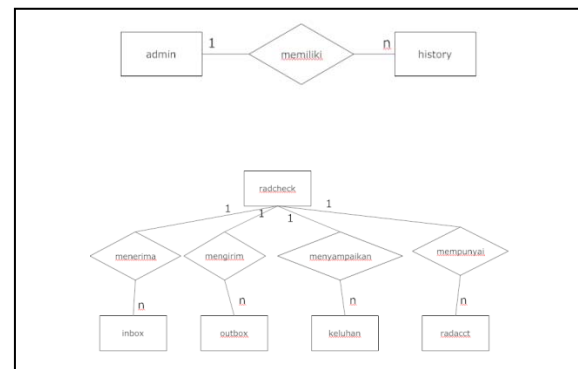
Tabel 3.1. Spesifikasi Kebutuhan Perangkat Lunak

3.2. Quick plan and modeling quick design

Tahap ini menjelaskan pemodelan data, pemodelan fungsi dan perancangan sistem.

3.2.1. Entity Relationship Diagram (ERD)

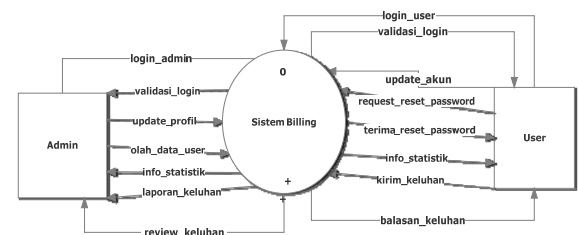
ERD adalah diagram yang menggambarkan hubungan antar himpunan entitas. Ada beberapa himpunan entitas yang diperlukan untuk kebutuhan data pada sistem ini, yaitu himpunan entitas *admin*, *history*, *radcheck*, *radacct*, *inbox*, *outbox*, dan *keluhan*. ERD pada system diilustrasikan pada gambar 3.2.



Gambar 3.2 ERD

3.2.2. Data Context Diagram (DCD)

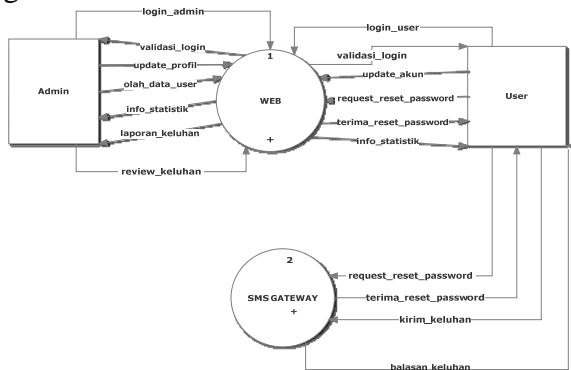
DCD atau DFD Level 0 sistem dapat dilihat pada gambar 3.2. Seluruh external entity akan mengakses satu Sistem Billing. External Entity tersebut adalah admin dan user.



Gambar 3.2 Data Context Diagram

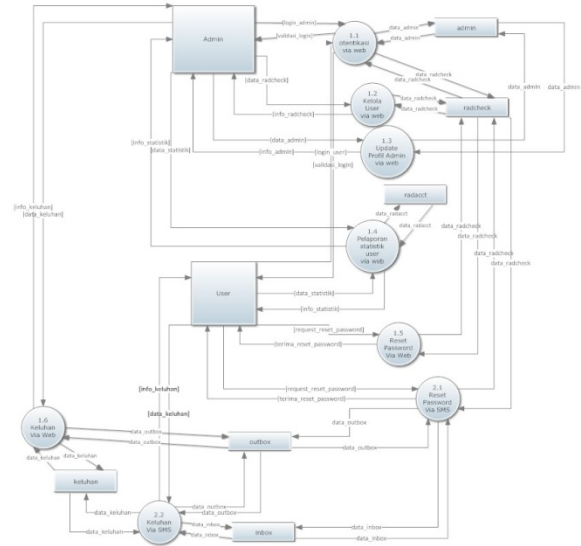
3.2.3. Data Flow Diagram (DFD)

DFD dapat dipecah menjadi sub sistem atau proses kecil. Dalam hal ini Sistem Billing mempunyai 2 sub sistem yang berjalan yaitu berbasis web dan SMS Gateway. DFD Level 1 dapat dilihat pada gambar 3.3.



Gambar 3.3 DFD Level 1

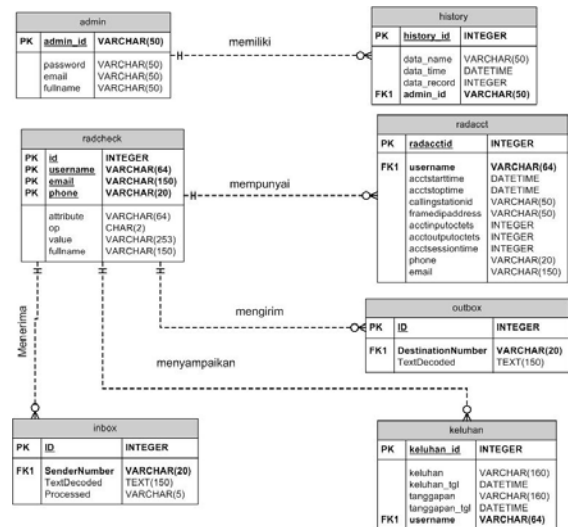
Pada DFD Level 2 Sub Sistem pada DFD Level 1 dapat dipecah menjadi beberapa proses kecil, guna menjelaskan fungsi-fungsi dan arus data yang terdapat pada sistem. DFD Level 2 dapat dilihat pada gambar 3.4. Pada DFD Level 2 terdapat 2 *external entity* yaitu admin dan user, 6 proses dari Sub Sistem Web, dan 2 proses dari Sub Sistem SMS Gateway.



Gambar 3.4 DFD Level 2

3.2.4. Conceptual Data Model (CDM)

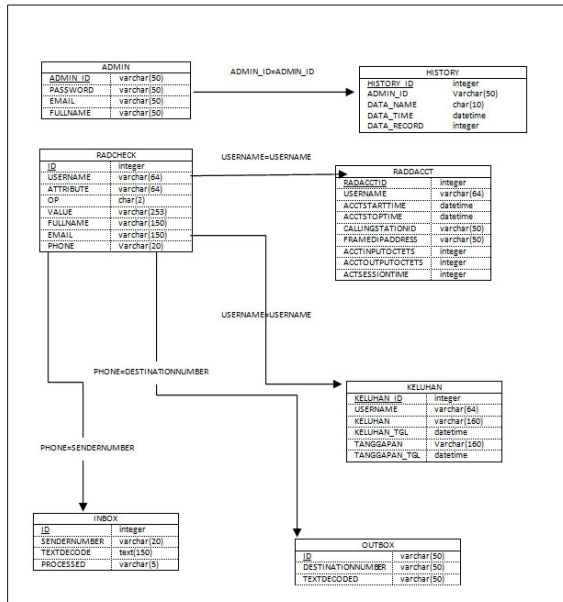
CDM merupakan perancangan basis data yang berdasarkan pengumpulan data dan analisis. CDM pada sistem terdiri atas 7 himpunan entitas. Himpunan entitas tersebut adalah admin, history, radcheck, radacct, inbox, outbox, dan keluhan. Rancangan data berupa CDM dapat dilihat pada gambar 3.5.



Gambar 3.5 CDM

3.2.5. Physical Data Model (PDM)

PDM merupakan model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antardata. PDM pada sistem terdiri atas 7 tabel. Tabel-tabel tersebut adalah admin, history, radcheck, radacct, inbox, outbox, dan keluhan. Rancangan data berupa PDM dapat dilihat pada Gambar 3.6.



Gambar 3.6 PDM

4. Implementasi dan Pengujian

Pada model proses *prototyping* tahap implementasi disebut *construction of prototype* dan tahap pengujian disebut *deployment, delivery and feedback*. Pengujian dilakukan dengan menguji spesifikasi perangkat lunak dengan metode *black box*.

4.1. Implementasi Perangkat Keras dan Perangkat Lunak

Spesifikasi perangkat keras yang digunakan dalam membangun IBS adalah sebagai berikut :

1) Untuk Komputer Server Radius

CPU : Intel 4 Core Xeon
 Memori : 2 GB
 Harddisk : 80 GB

2) Untuk Komputer SMS Gateway

CPU : Intel Core i3
 Memori : 2 GB
 Harddisk : 300 GB
 Modem USB : ZTE MF633+
 Kartu SIM : Three

Sedangkan Perangkat Lunak yang digunakan dalam membangun IBS sebagai berikut :

1) Untuk Komputer Server Radius

Sistem operasi : Ubuntu Server 12.04 64-bit
 Aplikasi : freeradius
 Webserver : Apache2, PHP5
 DBMS : MySQL

2) Untuk Komputer SMS Gateway

Sistem operasi : Ubuntu Desktop 12.04 32-bit
 Aplikasi : gammu
 Webserver : Apache2, PHP5
 DBMS : MySQL

Untuk konfigurasi Server sebagai berikut:

1) Konfigurasi Server Radius

a. Instal Apache, MySQL, PHP5, dan Freeradius

```

apt-get install freeradius
freeradius-mysql
apt-get install apache2
php5 mysql-server php5-mysql
    
```

b. Konfigurasi Freeradius

1. Masuk ke direktori `/etc/freeradius` dengan perintah
`cd /etc/freeradius`

2. Buat *database* baru untuk radius
`mysql -u root -p`
`mysql> create database radius;`
`mysql> quit`

3. *Import* database
`/etc/freeradius/sql/mysql/schema.sql`
`Mysql -u root -p radius`
`< sql/mysql/schema.sql`

4. *Edit* file `/etc/freeradius/sql.conf` seperti yang terlihat pada Kode 4.1
Vim `sql.conf`

```
sql {
    database = "mysql"
    driver = "rlm_sql_${database}"
    server = "localhost"
    login = "root"
    password = "gajahberdiri0k**//"
    radius_db = "radius"
    readclients = yes
}
```

Kode 4.1 Isi file `sql.conf`

5. *Edit* file `/etc/freeradius/clients.conf` seperti yang terlihat pada Kode 4.2
vim `clients.conf`

```
client localhost{
    ipaddr = 127.0.0.1
    secret = testing123
}
client 0.0.0.0/0 {
    secret      = gajahduduk
    shortname   = nas-mikrotik
}
```

Kode 4.2 Isi file `clients.conf`

6. *Edit* file `/etc/freeradius/radiusd.conf` dan menghilangkan tanda pagar (#) seperti yang terlihat pada Kode 4.3

```
$INCLUDE sql.conf
```

Kode 4.3 Isi file `radiusd.conf`

7. *Edit* file `/etc/freeradius/sites-enabled/default` hilangkan tanda pagar(#) pada line `#sql` seperti yang terlihat pada Kode 4.4

```
vim sites-enabled/default
```

```
Uncomment sql pada bagian authorize{}
    # See "Authorization Queries" in
    sql.conf
    sql
    Uncomment sql pada bagian
    accounting{}
    # See "Accounting queries" in
    sql.conf
    sql
    Uncomment sql pada bagian session{}
    # See "Simultaneous Use Checking
    Queries" in sql.conf
    sql
    Uncomment sql pada bagian post-
    auth{}
    # See "Authentication Logging
    Queries" in sql.conf
    sql
```

Kode 4.4 Isi File `default`

8. Kemudian *restart service radius* dengan perintah
`/etc/init.d/freeradius`
`restart`

9. Setelah konfigurasi selesai langkah selanjutnya adalah menguji radius server dengan perintah

```
radtest test 1234 127.0.0.1 1812
testing123
```

Hasilnya terlihat pada kode 4.5

```
Sending Access-Request of id 119 to
127.0.0.1 port 1812
  User-Name = "test"
  User-Password = "1234"
  NAS-IP-Address = 127.0.0.1
  NAS-Port = 1812
rad_recv: Access-Accept packet from
host 127.0.0.1 port 1812, id=119,
length=44
  Ascend-Xmit-Rate = 65535
  Ascend-Data-Rate = 393216
```

Kode 4.5 Hasil Tes Freeradius

2) Konfigurasi Server SMS Gateway

- a. Instal paket *gammu*, *gammu-smsd*, *Apache2*, *PHP5*, *MySQL*, dan *php5-mysql*

```
apt-get install gammu gammu-smsd
apache2 php5 mysql-server php5-mysql
```

- b. Konfigurasi *Gammu*

- 1) Terlebih dahulu dengan mengidentifikasi port *USB modem* dengan perintah

```
lsusb
```

Hasilnya harus menunjukkan pada kode 4.6

```
Bus 001 Device 002: ID 8087:0024
Intel Corp. Integrated Rate Matching Hub
Bus 002 Device 002: ID 8087:0024
Intel Corp. Integrated Rate Matching Hub
Bus 001 Device 001: ID 1d6b:0002
Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0002
Linux Foundation 2.0 root hub
Bus 002 Device 011: ID 19d2:0031 ZTE
WCDMA Technologies MSM MF110/MF627/MF636
```

Kode 4.6 Hasil identifikasi Modem USB

- 2) Edit file */etc/gammu-smsdrc* seperti pada kode 4.7

```
vim /etc/gammu-smsdrc
```

```
# Configuration file for Gammu SMS
Daemon

# Gammu library configuration, see
gammurc(5)
[gammu]
# Please configure this!
Port = /dev/ttyUSB1
connection = at115200
# Debugging
#logformat = textall

# SMSD configuration, see gammu-
smsdrc(5)
[smsd]
service = sql
runonreceive = /usr/bin/gammu-inbox-
daemon
user = root
password = gajahberdiri0k**//
pc = 182.255.2.43 #ip server radius
database = radius
driver = native_mysql
DeliveryReport = sms

logfile = /var/log/smsdlog
# Increase for debugging information
debuglevel = 1

# Paths where messages are stored
inboxpath = /var/spool/gammu/inbox/
outboxpath =
/var/spool/gammu/outbox/
sentsmspath = /var/spool/gammu/sent/
errorsmspath =
/var/spool/gammu/error/
```

Kode 4.7 Isi file *gammu-smsdrc*

3) *Import* database ke basis data server radius dengan perintah

```
mysql -u root -p radius <
mysql.sql
```

4) Buat *script bash*

```
/usr/bin/gammu-inbox-
daemon
vim /usr/bin/gammu-inbox-
daemon
```

Yang berisi *script* seperti pada kode 4.8

```
#!/bin/bash
/usr/bin/php /var/www/autoreply.php
```

Kode 4.8 Isi file *gammu-inbox-daemon*

5) Selanjutnya yaitu me-restart service *gammu* dengan perintah berikut

```
/etc/init.d/gammu-smsd restart
```

4.2. Implementasi Basis Data

Implementasi basis data merupakan transformasi data yang dihasilkan dari proses perancangan data menjadi suatu basis data dari IBS. Basis data ini terdiri atas 7 tabel yang meliputi tabel *admin*, *history*, *radcheck*, *radacct*, *inbox*, *outbox*, dan *keluhan*. 7 tabel tersebut dapat dilihat pada tabel 4.1.

Tabel 4.1 Implementasi Basis Data

NO	Nama Tabel	Field	Tipe Data	Deskripsi Isi
1.	admin	admin_id	varchar(50)	Berisi data admin
		password	varchar(50)	
		email	varchar(50)	
		fullname	varchar(50)	
2.	history	history_id	integer(5)	Berisi data history tambah data melalui excel
		data_name	varchar(150)	
		data_time	datetime	
		data_record	integer(11)	
		admin_id	varchar(50)	
3.	radcheck	id	integer(11)	Berisi data user
		username	varchar(64)	

		attribute	varchar(64)	
		op	char(2)	
		value	varchar(253)	
		fullname	varchar(150)	
		email	varchar(150)	
		phone	phone(50)	
4.	radacct	radacctid	bigint(21)	Berisi data session dari user
		username	varchar(64)	
		acctstarttime	datetime	
		acctstoptime	datetime	
		callingstationid	varchar(50)	
		framedipaddress	varchar(15)	
		acctinputoctets	bigint(20)	
		acctoutputoctets	bigint(20)	
5.	inbox	ID	integer(10)	Berisi data kotak masuk SMS
		SenderNumber	varchar(20)	
		TextDecoded	text	
		Processed	enum('false', 'true')	
6.	outbox	ID	integer(10)	Berisi data kotak keluar SMS
		DestinationNumber	varchar(20)	
		TextDecoed	text	
7.	keluhan	keluhan_id	integer(5)	Berisi data keluhan dari user
		keluhan	varchar(160)	
		keluhan_tgl	datetime	
		tanggapan	varchar(160)	
		tanggapan_tgl	Datetime	
		username	varchar(64)	

4.3. Implementasi Fungsi

Tabel implementasi fungsi dapat dilihat pada tabel 4.2.

Tabel 4.2 Implementasi Fungsi

NO	Fungsi	Nama File
1.	SRS-IBS-F01 Melakukan Otentikasi admin	../ibs/admin/index.php
2.	SRS-IBS-F02 Melakukan otentikasi user	../ibs/login.php
3.	SRS-IBS-F03.1 Menambah user	../ibs/admin_datauser.php
4.	SRS-IBS-F03.2 Mengubah data user	../ibs/admin_datauser.php
5.	SRS-IBS-F03.3 Menghapus data user	../ibs/admin_datauser.php
6.	SRS-IBS-F04 Menampilkan data statistik	../ibs/admin_datastatistik.php
7.	SRS-IBS-F05 Mengelola keluhan user	../ibs/admin_keluhan.php
8.	SRS-IBS-F06.1 Melakukan <i>reset</i>	../ibs/forgot.php

	password via web	
9.	SRS-IBS-F06.2 Melakukan reset password via SMS	../smsd/autoreply.php

4.4. Deployment, Delivery and Feedback

Tahap *deployment*, *delivery* and *feedback* disebut juga tahap pengujian. Pengujian adalah proses eksekusi suatu program dengan maksud untuk mengevaluasi fungsi-fungsi yang tidak benar atau hilang, kesalahan antarmuka, dan kesalahan dalam struktur data mungkin terjadi sehingga dapat diperbaiki sedini mungkin untuk menjamin bahwa aplikasi memenuhi spesifikasi yang telah ditentukan dari awal pembuatan aplikasi. Pengujian Sistem ini dilakukan pada persyaratan fungsional perangkat lunak (*black-box*) dengan memasukkan serangkaian kondisi input yang sepenuhnya menggunakan persyaratan fungsional program.

Berdasarkan tabel hasil uji dapat dilihat bahwa semua kelas uji sebagaimana pada Lampiran D sudah diuji dan telah sesuai dengan SRS, sehingga sistem telah memenuhi persyaratan perangkat lunak yang sudah didefinisikan. Dari hasil pengujian dapat diketahui bahwa sistem ini telah memenuhi untuk:

- Dapat melakukan otentikasi admin dan user
- Admin dapat mengelola data user
- User dapat meng-*update* akun/profil dan melihat data statistik diri sendiri
- Dapat menerima keluhan user untuk selanjutnya ditanggapi oleh admin
- User dapat meminta *reset password* jika tidak bisa melakukan otentikasi sehingga tidak perlu datang langsung ke administrator jaringan.

Dengan terpenuhinya hasil uji tersebut maka *prototype* sistem ini bisa diterapkan ke Fakultas Sains dan Matematika Universitas Diponegoro.

5. Penutup

5.1. Kesimpulan

Kesimpulan yang dapat diambil dalam pengerjaan tugas akhir ini adalah dihasilkannya sebuah *Hotspot Billing System* yang interaktif berbasis *web* dan SMS gateway dengan penjelasan sebagai berikut:

- Dengan adanya *Hotspot Billing System* akan memudahkan administrator jaringan dalam mengelola user hotspot
- Interaksi secara langsung antara user hotspot dengan administrator jaringan menjadi berkurang dengan dijumpai oleh sistem.
- Dengan diterapkannya *prototype process model*, dapat memperoleh *feedback* dari *client* lebih cepat sehingga mempercepat proses pengembangan sistem.

5.2. Saran

Untuk pengembangan lebih lanjut, sistem dapat dikembangkan agar semakin dapat disempurnakan dengan menambahkan beberapa fitur, sehingga memudahkan *user* dalam memanfaatkan jaringan *hotspot*.

DAFTAR PUSTAKA

- [1] "About Gammu", Diakses dari <http://wammu.eu/gammu/> pada tanggal 2 Maret 2013 Pukul 09.30 WIB.
- [2] Amak Yunus, M.Kom.2002. "Implementasi sistem otentikasi pada pengguna Jaringan hotspot di universitas kanjuruhan Malang guna meningkatkan keamanan Jaringan Komputer", Universitas Kanjuruhan, Malang.
- [3] Firmansyah, I Ketut Gede Suhartana, 2012, "Implementasi Mikrotik Automated Billing dengan Smtip (Simple

- Mail Transfer Protocol) Di PT. Rabik Bangun Pertiwi Blueline Broadband Internet*", Universitas Udayana, Bali.
- [4] Gutmans, Andi., dkk, 2005, "*PHP 5 Power Programming*", Prentice Hall.
- [5] Pohan, Husni Iskandar., dkk, 1997, "*Pengantar Perancangan Sistem*", Erlangga, Jakarta.
- [6] Pressman, Roger S., 2010, "*Software Engineering : A Practitioner's Approach Seventh Edition*", McGraw – Hill, New York.
- [7] Riyanto, Prilnali Eka Putra, dan Hendi Indelarko, 2009, "Pengembangan Aplikasi Sistem Informasi Geografis Berbasis Desktop dan Web", Gava Media, Yogyakarta.
- [8] Sommerville, Ian., 2003, "*Software Engineering (Rekayasa Perangkat Lunak)*", Erlangga, Jakarta.
- [9] Thoyib. 2010. "*Pelayanan Pelanggan Berbasis SMS Gateway*". Jurnal UMB. Yogyakarta
- [10] Widodo, Aris Puji, Djalal ER Riyanto, dan Beta Noranita, 2004, "*Buku Ajar Basis Data*", Jurusan Matematika Fakultas MIPA Universitas Diponegoro.
- [11] Winarko, Edi, 2006, "*Perancangan Database Dengan Power Designer 6.32*", Prestasi Pustaka, Jakarta.