

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

Bab ini akan membahas detail dari pembuatan, hasil pengujian dan analisis sistem. Analisis dilakukan untuk mengetahui kelebihan dan kekurangan dari sistem manajemen presensi siswa menggunakan OpenWrt dan RFID yang telah dibuat.

#### **4.1 Pembuatan Sistem**

Pada tahap pembuatan sistem akan dijelaskan bagaimana perancangan dan pembuatan sistem manajemen presensi siswa menggunakan RFID dan OpenWrt.

##### **4.1.1 Perangkat Lunak yang Digunakan**

Pada pembuatan sistem manajemen presensi siswa menggunakan RFID dan OpenWrt memerlukan beberapa perangkat lunak sebagai berikut:

1. Sistem Operasi Windows 7

Proses pembuatan sistem manajemen presensi siswa mulai dari instalasi OpenWrt pada router sampai dengan pembuatan antarmuka berupa sistem informasi berbasis *web* dilakukan melalui Windows 7.

2. Putty

Putty digunakan untuk melakukan melakukan *remote*, penulisan program, pemindahan data, dan pengaturan-pengaturan tertentu pada sistem operasi OpenWrt yang terdapat pada router melalui sistem operasi Windows 7.

3. Sistem Operasi OpenWrt

Sistem operasi OpenWrt digunakan sebagai wadah dan pengendali dari sistem manajemen presensi siswa.

##### **4.1.2 Perangkat Keras yang Digunakan**

Pada pembuatan dan uji coba sistem ini menggunakan sebuah laptop. Adapun spesifikasi dari laptop yang digunakan dalam pembuatan sistem ini adalah sebagai berikut:

1. Prosesor Core i3.
2. RAM DDR3 2 GB.
3. Kartu Grafis 2 GB.
4. Hardisk 500 GB.

Selain laptop dengan spesifikasi yang sudah dijelaskan di atas, perangkat keras lain yang digunakan adalah sebagai berikut:

1. Router Wireless Access Point HUAWEI Echolife HG553
2. Kabel UTP
3. Modem 3g HUAWEI E1750

#### **4.1.3 Persiapan Pembuatan Sistem**

Persiapan pembuatan sistem dimulai dengan mempersiapkan komponen-komponen untuk membangun sistem. **Gambar 4.1** merupakan komponen - komponen digunakan dalam sistem sesuai dengan urutan sebagai berikut:

1. Modul RFID RDM6300
2. USB Hub
3. Router *wireless access point* HUAWEI Echolife HG553
4. Kartu RFID
5. PL2303 USB to TTL Converter
6. USB modem HUAWEI E1750
7. USB *flash drive* TOSIBHA 16 GB
8. Kabel *jumper*

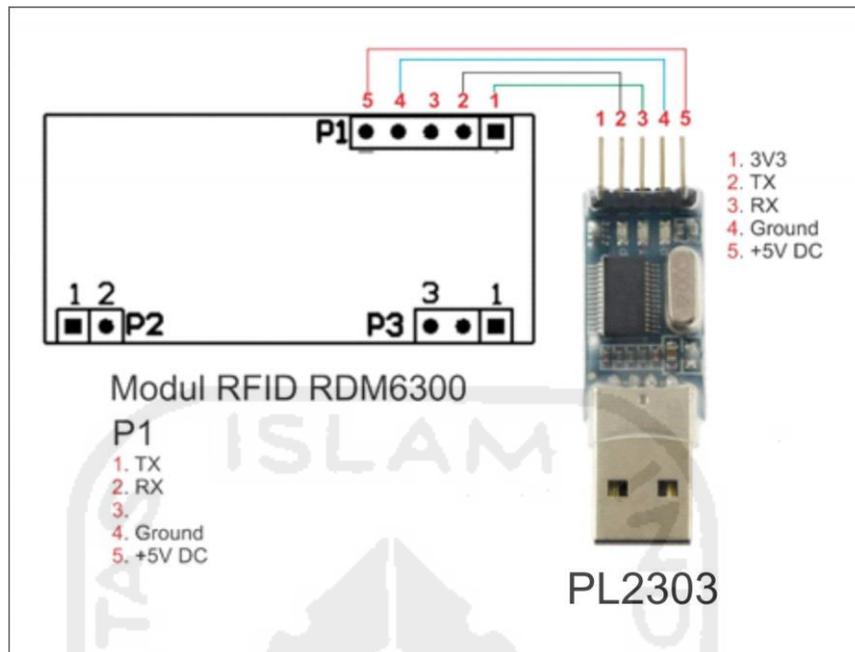
Persiapan selanjutnya adalah mempersiapkan perangkat lunak yang digunakan pada sistem yaitu Sistem operasi OpenWrt versi Barrier Breaker 14.04 dan paket-paket yang dibutuhkan pada sistem termasuk aplikasi Smstools3 yang digunakan sebagai pengiriman SMS.



**Gambar 4.1** Komponen-Komponen yang Digunakan dalam Sistem

#### 4.1.4 Perangkaian Komponen

Pada tahap ini, komponen-komponen yang sudah dijelaskan sebelumnya dirangkai menjadi satu. Proses perangkaian yang harus diperhatikan adalah menghubungkan antara modul RFID RDM6300 dengan PL2303 (USB 2.0 to TTL converter). Skema perangkaian antara dua komponen tersebut dapat dilihat pada **Gambar 4.2**.



**Gambar 4.2** Skema Perangkaian Modul RFID RDM6300 dengan PL2303

Dilanjutkan dengan menghubungkan keseluruhan komponen-komponen lainnya yaitu sebagai berikut:

1. RFID RDM6300 yang sudah mempunyai antarmuka USB 2.0 dipasang pada USB hub *port* 0.
2. Modem HUAWEI E1750 dipasang pada USB hub *port* 1.
3. USB hub di pasang pada *port* USB 2.0 router HUAWEI HG553 bagian bawah.
4. USB *flash drive* dipasang pada *port* USB 2.0 router HUAWEI HG553 bagian atas.

Hasil perangkaian komponen untuk sistem manajemen presensi siswa menggunakan RFID dan OpenWrt dapat dilihat pada **Gambar 4.3**.



Sebelum melakukan konfigurasi lebih jauh, terlebih dahulu harus diketahui *port* mana saja yang digunakan oleh *device* yang terhubung dengan router dengan menuliskan perintah `dmesg` melalui *command line* Putty. Hasil dari perintah `dmesg` dapat dilihat pada **Gambar 4.5**.

```
[ 22.384000] usbcore: registered new interface driver pl2303
[ 22.388000] usbserial: USB Serial support registered for pl2303
[ 22.396000] pl2303 1-2.5:1.0: pl2303 converter detected
[ 22.416000] usb 1-2.5: pl2303 converter now attached to ttyUSB0
[ 22.436000] PPP generic driver version 2.4.2
[ 22.448000] NET: Registered protocol family 24
[ 22.572000] b43-phy0: Broadcom 4318 WLAN found (core revision 9)
[ 22.612000] b43-phy0: Found PHY: Analog 3, Type 2 (G), Revision 7
[ 22.616000] b43-phy0: Found Radio: Manuf 0x17F, ID 0x2050, Revision 8, Version 0
[ 22.644000] Broadcom 43xx driver loaded [ Features: PNL ]
[ 22.672000] usbcore: registered new interface driver option
[ 22.680000] ieee80211 phy0: Selected rate control algorithm 'minstrel_ht'
[ 22.696000] usbserial: USB Serial support registered for GSM modem (1-port)
[ 22.704000] option 1-2.1:1.0: GSM modem (1-port) converter detected
[ 22.728000] usb 1-2.1: GSM modem (1-port) converter now attached to ttyUSB1
[ 22.736000] option 1-2.1:1.3: GSM modem (1-port) converter detected
[ 22.748000] usb 1-2.1: GSM modem (1-port) converter now attached to ttyUSB2
[ 22.756000] option 1-2.1:1.4: GSM modem (1-port) converter detected
[ 22.772000] usb 1-2.1: GSM modem (1-port) converter now attached to ttyUSB3
```

**Gambar 4.5** Deteksi Port

Pada gambar tersebut dapat dilihat bahwa PL2303 yang merupakan penghubung antara RFID dengan router terdapat pada `ttyUSB0`, sedangkan *modem* 3g yang digunakan terdapat pada `ttyUSB1`, `ttyUSB2`, dan `ttyUSB3`. Selanjutnya adalah proses konfigurasi, konfigurasi-konfigurasi yang dilakukan beserta dengan fungsinya adalah sebagai berikut:

### 1. Koneksi Internet

Koneksi internet didapatkan dengan melakukan konfigurasi *dial up* melalui antarmuka *web* Luci. Konfigurasi koneksi *dial up* dapat dilihat pada **Gambar 4.6**. Dilanjutkan dengan melakukan konfigurasi untuk *wireless hotspot* agar sistem dapat diakses melalui jaringan nirkabel.

**Gambar 4.6** Konfigurasi Koneksi *Dial Up*

## 2. Nano Text Editor

Nano merupakan aplikasi yang digunakan untuk penulisan kode program. Instalasi yang dilakukan adalah dengan mengetikkan perintah sebagai berikut:

```
root@OpenWrt:~# opkg install nano
```

## 3. Melakukan Exroot

Exroot adalah menambah kapasitas *storage* pada sistem operasi OpenWrt. Secara *default* kapasitas *storage* pada router HUAWEI HG553 hanya 16MB, maka dari itu dilakukan proses *Exroot* agar kapasitas *storage* mencukupi untuk kebutuhan sistem. Tahapan untuk melakukan proses *exroot* adalah sebagai berikut:

1. Melakukan instalasi paket yang dibutuhkan yaitu block-mount, kmod-fs-ext4, kmod-usb-storage, kmod-usb-ohci kemudian reboot.

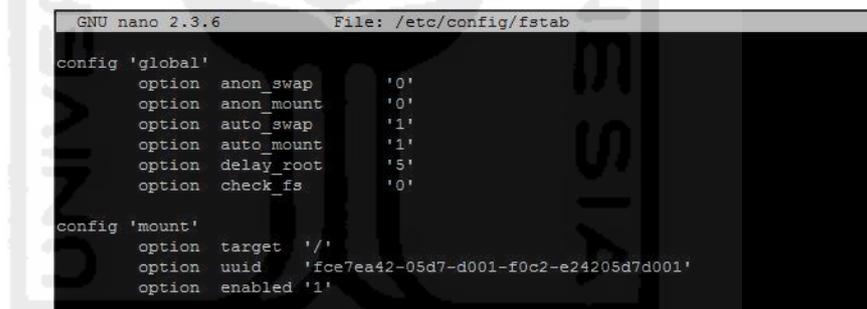
```
root@OpenWrt:~# opkg install block-mount kmod-
fs-ext4 kmod-usb-storage kmod-usb-ohci
root@OpenWrt:~# reboot
```

2. Melakukan pengecekan *pendrive* dari USB *flash drive* yang digunakan dengan mengetik perintah `block info`. Hasil yang didapat dari pengecekan adalah USB *flash drive* terdapat pada *pendrive* `sda1` dengan UUID `fce7ea42-05d7-d001-f0c2-e24205d7d001`.
3. Melakukan *copy* sistem OpenWrt pada USB *flash drive* dengan tujuan menjadikan USB *flash drive* sebagai *storage* utama sistem.

```
root@OpenWrt:~# mount /dev/sda1 /mnt
root@OpenWrt:~# mkdir /tmp/openwrt
root@OpenWrt:~# mount --bind / /tmp/openwrt
root@OpenWrt:~# tar -C /tmp/openwrt -cvf - . |
tar -C /mnt -xvf -
root@OpenWrt:~# sync ; umount /mnt
root@OpenWrt:~# umount /tmp/openwrt
root@OpenWrt:~# block detect > /etc/config/fstab
```

4. Mengubah konfigurasi pada `fstab` dengan menulis perintah:

```
root@OpenWrt:~# nano /etc/config/fstab
```



```
GNU nano 2.3.6 File: /etc/config/fstab

config 'global'
  option anon_swap '0'
  option anon_mount '0'
  option auto_swap '1'
  option auto_mount '1'
  option delay_root '5'
  option check_fs '0'

config 'mount'
  option target '/'
  option uuid 'fce7ea42-05d7-d001-f0c2-e24205d7d001'
  option enabled '1'
```

**Gambar 4.7** Konfigurasi `fstab`

Pada kode `option uuid` harus sesuai dengan *pendrive* yang digunakan ketika dilakukan pengecekan pada tahap pertama.

5. Tahap terakhir adalah menjalankan `fstab` dan melakukan pengaturan agar `fstab` berjalan setiap sistem melakukan *booting*.

```
root@OpenWrt:~# /ect/init.d/fstab enable
root@OpenWrt:~# /etc/init.d/fstab start
root@OpenWrt:~# reboot
```

6. Melakukan pengecekan apakah proses `exroot` berhasil dengan mengetikkan perintah:

```
root@OpenWrt:~# df -h
```

Filesystem	Size	Used	Available	Use%	Mounted on
rootfs	10.7G	346.9M	9.8G	3%	/
/dev/root	2.3M	2.3M	0	100%	/rom
tmpfs	30.0M	656.0K	29.4M	2%	/tmp
/dev/sda1	10.7G	346.9M	9.8G	3%	/
tmpfs	512.0K	0	512.0K	0%	/dev

**Gambar 4.8** Hasil *Exroot*

Pada **Gambar 4.8** dapat dilihat bahwa *storage* sistem yaitu `rootfs` sekarang mempunyai kapasitas sebesar 10.7 GB.

#### 4. Konfigurasi *driver* dan *baudrate* untuk RFID RDM6300

Konfigurasi *baudrate* harus sesuai dengan spesifikasi RFID RDM6300 yang digunakan agar komunikasi antara sistem OpenWrt dengan RFID RDM6300 yang dihubungkan melalui PL2303 USB 2.0 to TTL *converter* dapat berjalan, tahapannya adalah sebagai berikut:

1. Melakukan instalasi paket `kmod-usb-serial-pl2303` sebagai *driver* untuk *device* PL2303 USB 2.0 to TTL *converter* dan paket `coreutils-stty` untuk konfigurasi *baudrate* RFID RDM6300 dengan menulis perintah:

```
root@OpenWrt:~# opkg install kmod-usb-serial-pl2303
root@OpenWrt:~# opkg install coreutils-stty
```

2. Konfigurasi *baudrate* sesuai dengan spesifikasi RFID RDM6300 dengan menulis perintah:

```
root@OpenWrt:~# stty -F /dev/ttyUSB0 9600 clocal
cread cs8 -cstopb -parenb
```

#### 5. Instalasi dan Konfigurasi PHP5

PHP5 merupakan bahasa pemrograman yang digunakan dalam pembuatan sistem informasi berbasis *web* pada sistem manajemen presensi siswa. Tahapan instalasi dan konfigurasi PHP5 adalah sebagai berikut:

1. Melakukan instalasi paket yang dibutuhkan yaitu php5, php5-cgi dan php5-mod-mysql dengan menulis perintah:

```
root@OpenWrt:~# opkg install php5 php5-cgi php5-mod-mysql
```

2. Melakukan konfigurasi file *webserver* yaitu uhttpd, dengan menulis perintah:

```
root@OpenWrt:~# nano /etc/config/uhttpd
```

Kemudian menghilangkan tanda # di depan baris kode:

```
list interpreter ".php/etc/init.d/uhttpd
```

3. Restart *webserver* uhttpd dengan perintah:

```
root@OpenWrt:~# /etc/init.d/uhttpd restart
```

4. Melakukan pengecekan melalui *browser* dengan mengakses alamat 192.168.1.1/phpinfo.php. Apabila tidak terdapat pesan *error* berarti konfigurasi untuk PHP5 berhasil dilakukan.

## 6. Instalasi dan Konfigurasi MySQL

MySQL digunakan sebagai *database* pada sistem ini. Semua data pada sistem tersimpan pada *database* MySQL. Tahapan proses instalasi dan konfigurasi MySQL adalah sebagai berikut:

1. Instalasi paket yang dibutuhkan dengan menuliskan perintah:

```
root@OpenWrt:~# opkg install mysql-server
```

2. Membuat folder di mana data disimpan, pada sistem ini data akan disimpan pada folder *mysql* yang terdapat dalam folder *srv*. Perintah yang dibutuhkan untuk pembuatan folder tersebut adalah:

```
root@OpenWrt:~# mkdir -p /srv/mysql
```

Kemudian melakukan pengaturan pada file *my.conf* melalui *text editor* nano dengan menuliskan perintah:

```
root@OpenWrt:~# nano /etc/my.conf
```

**Gambar 4.9** merupakan penulisan kode untuk konfigurasi yang ditambahkan pada file `my.conf`.

```
GNU nano 2.3.6 File: /etc/my.cnf
datadir      = /srv/mysql/
tmpdir       = /tmp/
```

**Gambar 4.9** Konfigurasi File `my.conf`

3. Inisialisasi perintah database, kemudian menjalankan `mysqld` (aplikasi MySQL) dan memastikan `mysqld` berjalan setiap sistem melakukan *booting* dengan menuliskan perintah:

```
root@OpenWrt:~# mysql_install_db -force
root@OpenWrt:~# /etc/init.d/mysqld enable
root@OpenWrt:~# /etc/init.d/mysqld start
```

4. Tahap terakhir adalah membuat *username* dan *password* untuk MySQL. Kemudian melakukan login dengan *username* dan *password* tersebut agar dapat melakukan akses pada *database*. Perintah yang dibutuhkan adalah sebagai berikut:

```
root@OpenWrt:~# mysqladmin -u root password
admin123 //membuat username dan password
root@OpenWrt:~# mysql -u root -p admin123
//akses ke database
```

## 7. Instalasi Python Pyserial

Python merupakan bahasa pemrograman utama yang digunakan sistem dari mulai proses membaca kode *tag*, penentuan waktu dan keterangan presensi, sampai dengan penyimpanan data presensi dalam *database*. Instalasi Python dilakukan dengan menuliskan perintah:

```
root@OpenWrt:~# opkg install python pyserial
```

## 8. Konfigurasi Smstools3

Smstools3 merupakan aplikasi yang digunakan sistem untuk proses pengiriman SMS. Smstools3 adalah aplikasi berbasis linux, kemudian dilakukan *compile* sehingga dapat digunakan untuk sistem operasi OpenWrt. Tahap yang harus dilakukan untuk instalasi dan konfigurasi adalah sebagai berikut:

1. Mengunduh file `smstools3_3.1.15-2_brcm63xx.ipk`, kemudian menaruh file tersebut ke dalam folder `/tmp` OpenWrt. Setelah itu dilakukan proses instalasi dengan menuliskan perintah:

```
root@OpenWrt:~# opkg install *ipk
```

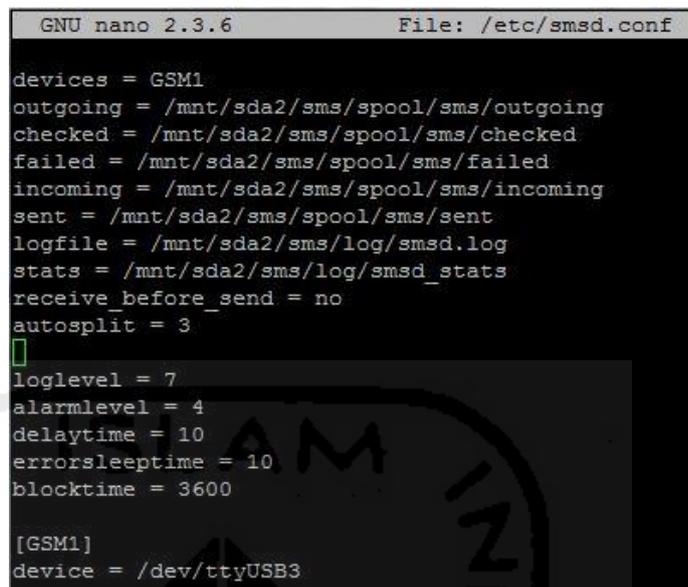
2. Membuat folder-folder yang dibutuhkan untuk smstools3. Pada sistem ini folder-folder untuk keperluan pengiriman SMS diletakkan pada `/mnt/sda2/sms/spool/sms/`, folder-folder tersebut adalah:
  - a. `outgoing`. Tempat di mana file.txt dikirimkan secara otomatis melalui SMS.
  - b. `checked`. File.txt dari folder `outgoing` dengan format yang sesuai dan akan langsung dikirimkan melalui SMS.
  - c. `failed`. Tempat untuk pengiriman SMS yang gagal.
  - d. `incoming`. Sama halnya seperti *inbox*.
  - e. `send`. Tempat untuk file SMS yang sudah berhasil dikirim.

Kemudian membuat file untuk *log* dan *stat* yaitu `smsd.log` dan `smsd_stats` yang ditempatkan pada folder `/mnt/sda2/sms/log/`.

3. Melakukan konfigurasi pada file `smsd.conf` melalui *text editor* nano dengan menuliskan perintah:

```
root@OpenWrt:~# nano /etc/smsd.conf
```

Konfigurasi pada file `smsd.conf` dapat dilihat pada **Gambar 4.10**.



```

GNU nano 2.3.6 File: /etc/smsd.conf

devices = GSM1
outgoing = /mnt/sda2/sms/spool/sms/outgoing
checked = /mnt/sda2/sms/spool/sms/checked
failed = /mnt/sda2/sms/spool/sms/failed
incoming = /mnt/sda2/sms/spool/sms/incoming
sent = /mnt/sda2/sms/spool/sms/sent
logfile = /mnt/sda2/sms/log/smsd.log
stats = /mnt/sda2/sms/log/smsd_stats
receive_before_send = no
autosplit = 3
loglevel = 7
alarmlevel = 4
delaytime = 10
errorsleeptime = 10
blocktime = 3600

[GSM1]
device = /dev/ttyUSB3

```

**Gambar 4.10** Konfigurasi File smsd.conf

4. Memastikan program smstools3 berjalan setiap *booting* dengan melakukan konfigurasi pada file rc.local.

```
root@OpenWrt:~# nano /etc/rc.local smsd -C
```

Pada file rc.local ditambahkan kode program sebagai berikut:

```
smsd -c /etc/smsd.conf &
```

5. Aplikasi smstools3 siap untuk digunakan. File.txt berisi nomor tujuan beserta pesan di dalamnya, akan dikirimkan secara otomatis setelah file.txt tersebut masuk ke dalam folder outgoing dengan format yang telah ditentukan.

#### 4.1.6 Penulisan Kode Program

Kode program ditulis dengan menggunakan *software* nano 2.3.6 yang telah dipasang pada tahap sebelumnya. Kode program pada sistem manajemen presensi siswa menggunakan RFID dan OpenWrt menggunakan bahasa pemrograman Python. Kode program pada sistem ini ditulis pada satu file yang diberi nama presensi.py.

Penulisan program dimulai dengan mengikutsertakan modul-modul yang dibutuhkan, yaitu sebagai berikut:

```
#!/usr/bin/python

import MySQLdb

from time import sleep

from os import popen

import serial

import time
```

Dilanjutkan dengan menulis kode program yang digunakan untuk koneksi ke *database* sistem, yaitu sebagai berikut:

```
host = "127.0.0.1"          #host database

user = "root"              #username database

passw = "password"        #password database

base = "presensi_rfid"    #nama database

db = MySQLdb.connect(host,user,passw,base) #perintah untuk
membuka koneksi ke database diberikan pada variable db

cur = db.cursor()         #cursor digunakan untuk melakukan eksekusi
query database. Diberikan pada variable cur
```

Kemudian dilanjutkan dengan penulisan kode program untuk konfigurasi koneksi serial. Konfigurasi harus sesuai dengan port yang digunakan dan spesifikasi dari tipe RFID yang digunakan yaitu RDM6300, yaitu sebagai berikut:

```

ser = serial.Serial()

ser.port = "/dev/ttyUSB0"           #port yang
digunakan

ser.baudrate = 9600                 #baudrate
RFID RDM6300

ser.bytesize = serial.EIGHTBITS    #jumlah bit per
byte

ser.parity = serial.PARITY_NONE     #set parity check: tanpa
parity

ser.stopbits = serial.STOPBITS_ONE  #jumlah stop bits

ser.timeout = 1                    #non-block yang terbaca

ser.xonxoff = False

```

Tahap selanjutnya adalah penulisan kode program untuk membaca *tag* RFID. Pada kode program di bawah ini terdapat variabel *rfid* yang berisikan `ser.read(14)`. Variabel *ser* merupakan `serial.Serial()` sedangkan `read` adalah kode program untuk membaca *tag* RFID dari serial. Kemudian (14) merupakan batasan *byte* yang akan terbaca, disesuaikan dengan jumlah karakter kode *tag* yang tertera pada kartu RFID yang digunakan. Dengan kata lain kode *tag* dengan 14 karakter yang didapatkan dari komunikasi serial ditempatkan pada variabel *rfid*.

```

try:
    ser.open() #membuka serial
except Exception, e:
    print "error open serial port: " + str(e) #pesan kesalahan
    exit() #program tertutup
if ser.isOpen():
    ser.flushInput()
    ser.flushOutput()
    while 1:
        rfidtag = ser.read(14)

```

Kemudian menulis kode program untuk mendapatkan format waktu yang dibutuhkan sistem dengan cara konversi, yaitu sebagai berikut:

```

now = time.localtime(time.time()) #waktu sekarang pada
sistem
f = '%Y-%m-%d' #format tanggal 1999-09-30
t = time.strftime(f, now) konversi waktu now menjadi
format f
g = '%H:%M:%S' #format waktu 00:00:00
j = time.strftime(g, now) #konversi waktu now menjadi
format g
z = '%d %B %Y' #format tanggal 30 September 1970
zx = time.strftime(z, now) #konversi waktu now
menjadi format z

```

Tahapan berikutnya adalah melakukan pengambilan data berupa waktu dari kolom start dan stop yang terdapat pada tabel presensi di mana keterangan dari data waktu tersebut adalah HADIR, TERLAMBAT, atau ABSEN. Kemudian melakukan konversi data tersebut menjadi format waktu yang digunakan sistem.

```

        cur.execute("SELECT start, stop from presensi WHERE
keterangan = 'HADIR'") #query data dari database
        result = cur.fetchall() #data menjadi matriks
        for row in result:
            stt_h = row[0] #baris ke 0 yaitu waktu start
            starts_h = time.strptime(stt_h, g) #konversi
            start_h = time.strftime(g, starts_h)
#konversi
            stp_h = row[1] #baris ke 1 yaitu waktu stop
            stops_h = time.strptime(stp_h, g) #konversi
            stop_h = time.strftime(g, stops_h) #konversi
        cur.execute("SELECT start, stop from presensi WHERE
keterangan = 'TERLAMBAT'")
        result = cur.fetchall()
        for row in result:
            stt_t = row[0]
            starts_t = time.strptime(stt_t, g)
            start_t = time.strftime(g, starts_t)
            stp_t = row[1]
            stops_t = time.strptime(stp_t, g)
            stop_t = time.strftime(g, stops_t)
        cur.execute("SELECT start, stop from presensi WHERE
keterangan = 'ABSEN'")
        result = cur.fetchall()
        for row in result:
            stt_a = row[0]
            starts_a = time.strptime(stt_a, g)
            start_a = time.strftime(g, starts_a)
            stp_a = row[1]
            stops_a = time.strptime(stp_a, g)

```

Dari kode program di atas, `start_h`, `stop_h`, `start_t`, `stop_t`, `start_a` dan `stop_a` adalah variabel-variabel yang digunakan untuk menentukan periode waktu untuk proses presensi.

Tahap selanjutnya adalah penulisan kode program untuk keseluruhan proses presensi, dimulai ketika terjadi proses *scan* kartu RFID (proses presensi) dan waktu pada sistem berada pada periode presensi yaitu antara periode HADIR dan TERLAMBAT. Sistem terlebih dahulu melakukan pengecekan pada *database* apakah *tag* tersebut sudah melakukan proses presensi pada tanggal di mana proses *scan* terjadi. Penulisan kode program adalah sebagai berikut:

```

        if start_h <= j <= stop_t: # jika waktu pada sistem
berada pada periode presensi

            if len(rfidtag)==14: # jika panjang byte dari kode
tag yang terbaca adalah 14 karakter

                query = """SELECT
presensi_siswa.id_presensi_siswa
                FROM presensi_siswa INNER JOIN siswa
                ON presensi_siswa.id_siswa = siswa.id_siswa
                WHERE presensi_siswa.tanggal = %s
                AND siswa.rfid = %s""" #pengecekan
                result = (t, rfidtag) #t untuk %s pertama
adalah waktu sistem ketika scan tag, rfidtag untuk %s kedua
adalah hasil scan kode tag rfid.
                cur.execute(query, (result)) #query dari
variable query dan result

                yet = len(cur.fetchall()) #hasil dari query
yang merupakan pengecekan data diberikan pada variable yet.

```

Proses penyimpanan data akan dilakukan apabila hasil pengecekan pada *database* menghasilkan nilai 0. Siswa akan tercatat dengan keterangan HADIR atau TERLAMBAT sesuai dengan periode ketika proses *scan* terjadi. Berikut ini merupakan kode program ketika proses *scan* kartu RFID pada periode hadir:

```

        if yet == 0: #jika pengecekan database
menghasilkan nilai 0
            if j <= stop_h: #jika waktu sistem
berada pada periode hadir
                ket = '1' # 1 adalah id pada
tabel presensi untuk keterangan hadir.
                query = """INSERT INTO
presensi_siswa(id_presensi, id_siswa, tanggal)
                SELECT %s, siswa.id_siswa, %s
                FROM siswa
                WHERE siswa.rfid = %s"""
                result = (ket, t, rfidtag)
#hadir untuk %s pertama berupa 1, t untuk %s kedua adalah
tanggal, rfidtag untuk %s ketiga adalah kode tag rfid.
                presensi = cur.execute(query,
(result)) #hasil query variable query dan result diberikan pada
varabel presensi
                db.commit()

```

Perbedaan periode hadir dengan terlambat terletak pada kode program:

```

if j <= stop_t:
ket = '2'

```

Pada kode di atas dapat dijelaskan bahwa ketika *j* (waktu sistem) kurang dari atau sama dengan waktu ditutupnya *stop\_t* (selesainya periode terlambat), dan '2' adalah id dari tabel presensi (*database*) dengan keterangan TERLAMBAT, maka hasil *scan* pada waktu tersebut akan menghasilkan keterangan TERLAMBAT.

Setelah tersimpan, data presensi tersebut diolah untuk dijadikan notifikasi berupa informasi presensi yang ditampilkan melalui antarmuka *web* kepada siswa yang telah melakukan proses presensi (*scan* kartu RFID). Dimulai dengan

mengambil data presensi dari *database* kemudian menuliskannya ke dalam sebuah file.txt. Kode programnya adalah sebagai berikut:

```

        if presensi:
            sms = """SELECT
presensi_siswa.id_presensi_siswa, presensi_siswa.id_siswa,
                                presensi_siswa.tanggal,
siswa.nama, siswa.kelas, siswa.nohp,
                                presensi.keterangan,
presensi.keterangan_sms
                                FROM presensi_siswa
                                INNER JOIN siswa
                                ON presensi_siswa.id_siswa
= siswa.id_siswa
                                INNER JOIN presensi on
presensi_siswa.id_presensi = presensi.id_presensi
                                WHERE tanggal = %s and rfid
= %s"""

            result1 = (t, rfidtag)
            cur.execute(sms, (result1))
            kirimsms = cur.fetchone()
            nmsms = kirimsms[0]
            ids = kirimsms[1]
            tgl = kirimsms[2]
            nm = kirimsms[3]
            kls = kirimsms[4]
            nohp = kirimsms[5]
            ket = kirimsms[6]
            ket_sms =

```

Pada kode program di atas, pertama-tama apabila variabel presensi (hasil query dari kode program sebelumnya untuk menyimpan data) bernilai benar, dilakukan pengambilan data dari *database* berupa:

1. id presensi siswa diberikan pada variabel nmsms.
2. id siswa diberikan pada variabel ids.
3. tanggal diberikan pada variabel tgl.

4. nama siswa diberikan pada variabel nm.
5. kelas siswa diberikan pada variabel kls.
6. nomor hp orang tua siswa diberikan pada variabel nohp.
7. keterangan presensi diberikan pada variabel ket.
8. keterangan sms diberikan pada variabel ket\_sms.

Di mana tanggal adalah ketika siswa melakukan proses *scan* RFID, dan rfid (kode *tag*) adalah yang digunakan di saat proses *scan* tersebut. Variabel-variabel tersebut digunakan untuk melakukan penulisan pada sebuah file dengan nama record.txt. Kode programnya adalah sebagai berikut:

```

                                f =
open('/www/presensi/siswa/record.txt', 'a') #membuka file
record.txt dari folder /www/presensi/siswa
                                f.write("NIS: %s\nNama:
%s\nTanggal: %s\nStatus: %s\n" % (ids, nm, tgl, ket)) #menulis
pada file record.txt tersebut dengan mengambil data dari
variabel ids, nm, tgl, dan ket
                                f.close()#menutup file
record.txt

```

Kode program di atas menghasilkan file record.txt yang berisi informasi presensi berupa NIS siswa, nama siswa, tanggal, dan keterangan (HADIR atau TERLAMBAT). File record.txt ini akan di-*request* oleh halaman *web* setiap detiknya. Jadi setiap proses presensi yang berhasil dilakukan, halaman *web* akan menampilkan informasi presensi tersebut. Berikut ini merupakan potongan dari kode program dari sisi *web* untuk melakukan *request* file record.txt tersebut dan menampilkannya menjadi sebuah informasi presensi siswa:

```

<div class="panel-body">
    <div id="record" style="font-size:40px"></div> //id record
    adalah tempat di mana informasi presensi siswa ditampilkan
</div>
<script type="text/javascript">
function loadXMLDoc()
{
var xmlhttp;
if (window.XMLHttpRequest)
    { // code for IE7+, Firefox, Chrome, Opera, Safari
    xmlhttp=new XMLHttpRequest();
    }
else
    { // code for IE6, IE5
    xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
xmlhttp.onreadystatechange=function()
{
    if (xmlhttp.readyState==4 && xmlhttp.status==200)
        {
        document.getElementById("record").innerHTML+xmlhttp.responseText.split('\n').slice(-5).join('<br/>'); //menempatkan elemen-
        } //yang didapat ke dalam id record.
        }
xmlhttp.open("GET","siswa/record.txt",true); //membuka file-
xmlhttp.send(); //record.txt kemudian mengirimkannya.
}
var myVar=setInterval(function(){myTimer();},1000); //request-
function myTimer() { //setiap satu detik
    loadXMLDoc();
}
</script>

```

Tidak selesai sampai di situ, hasil pengambilan data presensi tersebut juga digunakan untuk pengiriman SMS. Data yang diambil dari *database* adalah sebagai berikut:

1. Id presensi siswa sebagai nama dari file.txt yang dibuat.
2. Nomor HP sebagai tujuan pengiriman SMS.
3. NIS (Nomor Induk Siswa).
4. Nama siswa.
5. Kelas.
6. Keterangan, baik HADIR maupun TERLAMBAT.
7. Tanggal.
8. Keterangan SMS, di tulis pada akhir kalimat SMS.

Data-data tersebut akan ditulis pada file dengan nama yang dinamis sesuai dengan id presensi siswa, kemudian di letakkan pada folder outgoing sehingga dapat terkirim melalui SMS secara otomatis. Kode program nya adalah sebagai berikut:

```

f =
open('/mnt/sda2/sms/spool/sms/outgoing/%s.txt' % nmsms, 'a')
#membuka dan membuat file.txt dan meletakkanya ke folder
outgoing. Dengan nama sesuai dengan id presensi siswa.
f.write("To: %s\n\nAnak
Anda dengan NIS: %s, Nama: %s, Kelas: %s, Status: %s Pada
tanggal: %s\n%s." % (nohp, ids, nm, kls, ket, tgl,
ket_sms)) #menulis SMS sesuai dengan format.
f.close() #menutup file.txt

```

Pada **Gambar 4.11** dapat dilihat contoh file.txt yang dihasilkan secara otomatis oleh kode program di atas.

```
To: 089608023046
NIS: 1006, Nama: Firman, Kelas: 9B, Status: TERLAMBAT Pada tanggal: 2015-09-20.
Tingkatkan kedisiplinan.
```

**Gambar 4.11** File.txt untuk pengiriman SMS

Pada periode berikutnya yaitu periode absen, siswa yang tidak melakukan proses presensi (*scan* kartu RFID) sampai dengan periode absen dimulai akan tercatat dengan keterangan ABSEN. Tidak ada notifikasi berupa informasi presensi yang ditampilkan melalui antarmuka *web* oleh sistem seperti periode sebelumnya. Tetapi pengiriman informasi presensi berupa SMS yang dikirimkan kepada orang tua siswa tetap dilaksanakan.

Di luar periode presensi, terdapat periode di mana pendaftaran siswa baru atau perubahan data berupa kode *tag* RFID siswa dapat dilakukan. Kode program pada periode ini memiliki kesamaan dengan kode program sebelumnya yaitu kode program untuk membaca *tag* RFID, menuliskannya ke dalam sebuah file.txt kemudian halaman *web* akan melakukan *request* untuk mendapatkan isi dari file.txt tersebut. Perbedaannya adalah tidak ada proses penyimpanan data ke *database*.

Pada tahap terakhir, penulisan kode program untuk sistem informasi berbasis *web* dilakukan. Sistem informasi tersebut menggunakan bahasa pemrograman PHP. Semua data tentang informasi presensi siswa dapat diakses dan diolah oleh pihak sekolah maupun siswa melalui antarmuka *web*.

## 4.2 Pengujian Sistem

### 4.2.1 Pengujian Presensi Menggunakan RFID

Pada proses presensi menggunakan RFID, terdapat pengujian sistem yang harus dilakukan. Pengujian pertama adalah melakukan *scan* kartu RFID dengan mendekatkan kartu tersebut dengan RFID *reader* RDM6300 kemudian kode *tag* dari kartu RFID tersebut akan ditampilkan pada halaman *web*. Pengujian ini dapat

dilakukan ketika sistem diluar periode presensi. Maka dari itu mengubah pengaturan periode presensi melalui antarmuka *web* harus dilakukan terlebih dahulu. **Gambar 4.12**, **Gambar 4.13**, dan **Gambar 4.14** merupakan pengaturan periode presensi yang dilakukan melalui antarmuka *web* untuk tujuan pengujian.

Periode HADIR

**Status**  
HADIR

**Keterangan SMS**  
Pertahankan Kedisiplinan Anak Anda

**Waktu Mulai**  
03:00:00

**Waktu Selesai**  
03:05:00

**Gambar 4.12** Pengaturan Presensi Periode Hadir

Periode TERLAMBAT

**Status**  
TERLAMBAT

**Keterangan SMS**  
Tingkatkan Kedisiplinan Anak Anda

**Waktu Mulai**  
03:05:00

**Waktu Selesai**  
03:10:00

**Gambar 4.13** Pengaturan Presensi Periode Terlambat

Periode ABSEN	
<b>Status</b>	ABSEN
<b>Keterangan SMS</b>	Tingkatkan Pengawasan ke Anak Anda
<b>Waktu Mulai</b>	03:11:00 <input type="button" value="⌚"/>
<b>Waktu Selesai</b>	03:20:00 <input type="button" value="⌚"/>

**Gambar 4.14** Pengaturan Presensi Periode Absen

Dari tiga pengaturan periode presensi yang dapat dilihat pada **Gambar 4.12**, **Gambar 4.13**, dan **Gambar 4.14** dapat diketahui bahwa periode yang dapat digunakan untuk melakukan pengujian pertama adalah diluar periode presensi, yaitu pukul 03:00:00 sampai dengan pukul 03:20:00. Pengujian dilakukan dengan mendekatkan kartu RFID dengan RFID reader (**Gambar 4.15**).



**Gambar 4.15** Proses *Scan* Kartu RFID

Halaman *web* Tambah Siswa digunakan untuk melakukan pengujian apakah kode *tag* dari kartu RFID dapat ditampilkan. Pada **Gambar 4.16** dapat dilihat bahwa kode *tag* ditampilkan dalam *form* tambah siswa, yaitu pada kolom RFID-Tag, setelah proses *scan* kartu RFID dilakukan. Waktu pada sistem juga berada diluar periode presensi yaitu sekitar pukul 03:40:00.

**Tambah Siswa**

Peraturan Tambah Siswa Baru. Mohon Dibaca!

- Scan RFID-Tag untuk tambah siswa baru hanya bisa dilakukan diluar periode presensi
- Yaitu di luar pukul 03:00:00 sampai dengan pukul 03:20:00.

**RFID-Tag**  
6E008DD7FC4  
Scan RFID Card ke RFID Reader

**Nama**

**Kelas**  
Pilih Kelas

**Jenis Kelamin**  
Pilih Jenis Kelamin

**Tanggal Lahir**  
  
Format: YYYY-MM-DD

Friday, December 18, 2015  
December, 2015  
Su Mo Tu We Th Fr Sa  
29 30 1 2 3 4 5  
6 7 8 9 10 11 12  
13 14 15 16 17 18 19  
20 21 22 23 24 25 26  
27 28 29 30 31 1 2  
3 4 5 6 7 8 9  
3:39:12 AM  
Change date and time settings...

**Gambar 4.16** Kode *Tag* Ditampilkan pada *Form* Tambah Siswa

Pengujian berikutnya adalah mendapatkan notifikasi berupa informasi presensi pada periode hadir melalui halaman *web* Real Time Presensi dan melalui SMS. Informasi presensi didapatkan setelah proses *scan* kartu RFID dilakukan pada periode hadir, yaitu pukul 03:00:00 sampai dengan pukul 03:05:00. *Output* berupa informasi presensi melalui halaman *web* Real Time Presensi dapat dilihat pada **Gambar 4.17**. Informasi yang ditampilkan adalah NIS, Nama, Tanggal, dan keterangan HADIR, sesuai dengan waktu ketika proses *scan* kartu RFID dilakukan yaitu sekitar pukul 03:00:20.

Sedangkan SMS yang diterima melalui perangkat telepon genggam setelah proses *scan* kartu RFID dilakukan dapat dilihat pada **Gambar 4.18**. NIS, Nama,

Tanggal dan keterangan sesuai dengan informasi presensi yang ditampilkan pada halaman *web* Real Time Presensi.



**Gambar 4.17** Informasi Presensi dengan Keterangan HADIR



**Gambar 4.18** Informasi Presensi melalui SMS dengan keterangan HADIR

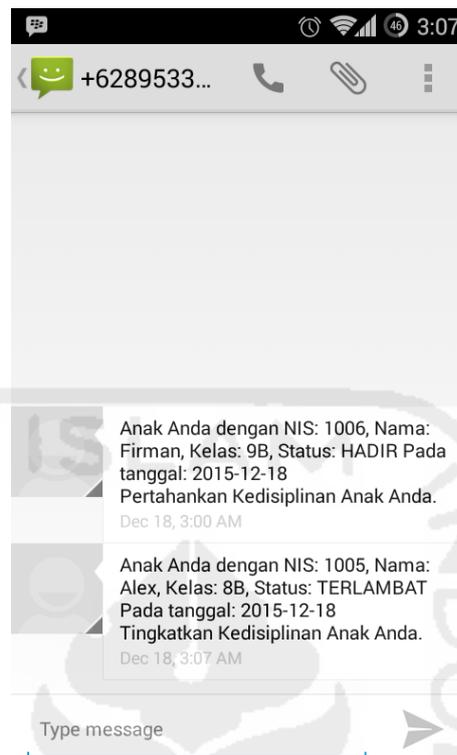
Pengujian berikutnya adalah mendapatkan notifikasi berupa informasi presensi pada periode terlambat melalui halaman *web* Real Time Presensi dan melalui SMS. Notifikasi didapatkan setelah proses *scan* kartu RFID dilakukan pada periode terlambat, yaitu pukul 03:05:00 sampai dengan pukul 03:10:00. *Output* berupa informasi presensi siswa melalui halaman *web* Real Time Presensi dapat dilihat pada **Gambar 4.19**. Informasi yang ditampilkan adalah NIS, Nama, Tanggal, dan keterangan HADIR, sesuai dengan waktu ketika proses *scan* kartu RFID dilakukan yaitu sekitar pukul 03:07:00.



**Gambar 4.19** Informasi Presensi dengan keterangan TERLAMBAT

Sedangkan SMS yang diterima melalui perangkat telepon genggam setelah proses *scan* kartu RFID dilakukan dapat dilihat pada **Gambar 4.20**. NIS, Nama, Tanggal dan keterangan sesuai dengan notifikasi yang ditampilkan pada halaman *web* Real Time Presensi.

Pengujian berikutnya adalah mendapatkan notifikasi melalui halaman *web* bahwa presensi sudah ditutup. Notifikasi tersebut ditampilkan ketika periode terlambat sudah selesai yaitu pada pukul 03:10:00. Tampilan dari Notifikasi tersebut dapat dilihat pada **Gambar 4.21**.



**Gambar 4.20** Informasi Presensi melalui SMS dengan keterangan HADIR



**Gambar 4.21** Notifikasi Bahwa Presensi Sudah Ditutup

Pengujian selanjutnya adalah mendapatkan SMS berupa informasi presensi siswa secara otomatis ketika waktu pada sistem memasuki periode absen, yaitu pukul 03:11:00 sampai dengan pukul 03:20:00. Semua siswa yang tidak melakukan presensi RFID sampai dengan pukul 03:11:00 otomatis akan tersimpan dalam data presensi dengan keterangan ABSEN kemudian informasi tersebut akan dikirimkan melalui SMS. Pada **Gambar 4.22** dapat dilihat bahwa SMS berupa informasi presensi siswa dengan keterangan ABSEN dikirim secara otomatis setelah pukul 03:11:00. Sedangkan **Gambar 4.23** adalah tampilan data presensi di hari aktif pada halaman *web* setelah proses presensi selesai dilakukan.



**Gambar 4.22** Informasi Presensi melalui SMS dengan keterangan ABSEN

Presensi Hari Ini

Ubah Data Presensi dengan Mencari NIS

Cari NIS ...

#	NIS	Nama	Kelas	Status	Tanggal (YYYY-MM-DD)	Edit
1	1006	Firman	9B	HADIR	2015-12-18	
2	1005	Alex	8B	TERLAMBAT	2015-12-18	
3	1003	Huda	7B	ABSEN	2015-12-18	
4	1004	Boby	7B	ABSEN	2015-12-18	



**Gambar 4.23** Data Presensi Tanggal 18 Desember 2015

#### 4.2.2 Pengujian Presensi Manual

Proses presensi manual dilakukan ketika siswa tidak membawa kartu RFID dan hanya dapat dilaksanakan ketika sistem berada pada periode hadir dan terlambat. Pada tahapan ini, pengujian yang dilakukan adalah untuk memastikan bahwa data presensi yang dihasilkan sesuai dengan waktu presensi manual dilakukan dan mendapatkan notifikasi SMS berupa informasi presensi setelah proses presensi manual dilakukan. Terlebih dahulu dilakukan pengubahan periode hadir dan terlambat melalui halaman *web* Ubah Pengaturan Periode Presensi, yang dapat dilihat pada **Gambar 4.24**.

Pada periode hadir yaitu pukul 22:42:00 dilakukan pengujian presensi manual. **Gambar 4.25** merupakan proses dari presensi manual yang dilakukan melalui halaman *web* Presensi Manual. Dapat dilihat bahwa data presensi yang dihasilkan sesuai dengan kapan waktu presensi manual dilakukan.

**Periode HADIR**

**Status**  
HADIR

**Keterangan SMS**  
Pertahankan Kedisiplinan Anak Anda

**Waktu Mulai**  
22:40:00

**Waktu Selesai**  
22:45:00

**Periode TERLAMBAT**

**Status**  
TERLAMBAT

**Keterangan SMS**  
Tingkatkan Kedisiplinan Anak Anda

**Waktu Mulai**  
22:45:00

**Waktu Selesai**  
22:50:00

**Gambar 4.24** Pengaturan Periode Presensi Hadir dan Terlambat

### Presensi Manual

**Tanggal**  
2015-12-18  
Format: YYYY-MM-DD

**NIS**  
1004 ✓

Friday, December 18, 2015



---

### Presensi Hari Ini

Presensi manual sukses

Ubah Data Presensi dengan Mencari NIS

Cari NIS ...

#	NIS	Nama	Kelas	Status	Tanggal (YYYY-MM-DD)	Edit
1	1004	Boby	7A	HADIR	2015-12-18	<input type="button" value="Edit"/>

Friday, December 18, 2015



**Gambar 4.25** Presensi Manual

Setelah tersimpan dalam *database*, data presensi tersebut secara otomatis akan terkirim melalui SMS. Informasi data presensi melalui SMS yang didapatkan setelah proses presensi manual dilakukan dapat dilihat pada **Gambar 4.26**. Dapat dilihat bahwa informasi presensi siswa yang diterima sesuai dengan data yang tersimpan dalam *database*.



**Gambar 4.26** Informasi Presensi Melalui SMS

#### 4.2.3 Pengujian Unduh Laporan

Pengujian unduh laporan dilakukan untuk memastikan bahwa data presensi dapat digunakan sebagai laporan harian. Pada halaman *web* Unduh Laporan (**Gambar 4.27**) terdapat tabel HTML berupa data presensi yang kemudian dapat diunduh sebagai laporan. Setelah proses unduh laporan dilakukan, dapat dilihat pada **Gambar 4.28** bahwa tabel HTML pada halaman Unduh Laporan sukses dikonversi menjadi file PDF yang digunakan sebagai laporan harian.

Unduh Laporan

Unduh Laporan Hari Ini

Semua Kelas

LAPORAN PRESENSI SEMUA KELAS Tanggal 01 October 2015

#	NIS	Nama	Kelas	Tanggal	Keterangan
1	1003	Huda	7B	01 October 2015	TERLAMBAT
2	1005	Alex	8B	01 October 2015	ABSEN
3	1004	Boby	7B	01 October 2015	IZIN
4	1006	Firman	9B	01 October 2015	IZIN
Jumlah Hadir					0
Jumlah Terlambat					1
Jumlah Absen					1
Jumlah Izin					2
Total Kehadiran					1

**Gambar 4.27** Tabel Data Presensi untuk di Unduh

LAPORAN PRESENSI SEMUA KELAS Tanggal 01 October 2015

#	NIS	Nama	Kelas	Tanggal	Keterangan
1	1003	Huda	7B	01 October 2015	TERLAMBAT
2	1005	Alex	8B	01 October 2015	ABSEN
3	1004	Boby	7B	01 October 2015	IZIN
4	1006	Firman	9B	01 October 2015	IZIN
Jumlah Hadir					0
Jumlah Terlambat					1
Jumlah Absen					1
Jumlah Izin					2
Total Kehadiran					1

**Gambar 4.28** Hasil Unduh Laporan Presensi dengan Format PDF

#### 4.2.4 Penanganan Kesalahan Sistem Informasi

Tahap ini menjelaskan mengenai penanganan kesalahan pada sistem informasi berbasis *web* pada sistem manajemen presensi siswa menggunakan RFID dan OpenWrt. Pengujian dilakukan secara menyeluruh untuk tingkat keberhasilan

sistem yang dibangun serta mengetahui apakah masih terjadi kesalahan-kesalahan pada sistem. Penanganan kesalahan dari halaman *web* sistem manajemen presensi siswa adalah sebagai berikut:

1. Penanganan Kesalahan Login

Penanganan kesalahan login ini dilakukan ketika administrator salah mengisikan username maupun password ketika login. Pesan *error* yang ditampilkan dapat dilihat pada **Gambar 4.29**.



**Gambar 4.29** Penanganan Kesalahan Login

2. Penanganan Kesalahan Ubah Data Presensi

Pesan kesalahan ditampilkan ketika status atau keterangan presensi belum dipilih. Penanganan kesalahan ketika mengubah data presensi dapat dilihat pada **Gambar 4.30**.

### Ubah Data Presensi (ID Presensi Siswa : 29606)

The screenshot shows a web form titled "Ubah Data Presensi (ID Presensi Siswa : 29606)". The form has several input fields:

- NIS:** 1006
- Nama:** Firman
- Kelas:** 9B
- Tanggal:** 2015-12-18 (with a "YYYY-MM-DD" format hint below it)
- Status Sebelumnya:** HADIR
- Ganti Status:** This field is highlighted with a red border and contains a red "x" icon. Below it is the error message: "Status harus diisi dan tidak boleh kosong".

At the bottom of the form is a blue "Simpan" button.

**Gambar 4.30** Penanganan Kesalahan Ubah Data Presensi

### 3. Penanganan Kesalahan Ubah Data Siswa

Pesan kesalahan ditampilkan ketika terdapat data yang belum diisi dan apabila *tag* RFID sudah dipakai oleh siswa lain. Penanganan kesalahan ini sama dengan proses tambah data siswa. *Tag* RFID yang digunakan harus unik, dan belum digunakan oleh siswa lain. Sebagai contoh pada **Gambar 4.31** kode *tag* 6E0009F3E074 sudah digunakan oleh siswa dengan nama Bobby. Kemudian dilakukan ubah data pada siswa dengan nama Huda (**Gambar 4.32**). Sistem akan menampilkan pesan kesalahan (**Gambar 4.33**) ketika siswa dengan nama Huda menggunakan *tag* 6E0009F3E074 yang sudah digunakan oleh siswa dengan nama Bobby. Proses ubah *tag* RFID tersebut dilakukan dengan cara melakukan *scan* kartu RFID.

## Database Siswa

Ubah atau Hapus Data Siswa dengan Mencari NIS

#	NIS	RFID	Nama	Kelas
1	1003	6E001092A448	Huda	7B
2	1004	6E0009F3E074	Boby	7B
3	1005	6E0008571726	Alex	8B
4	1006	6E0008DD7FC4	Fiman	9B

**Gambar 4.31** Kode *Tag* yang digunakan

### Ubah Data Siswa (NIS: 1003)

RFID-Tag

6E0009F3E074

Scan RFID Card baru ke RFID Reader atau tetap menggunakan yang lama

Nama

Huda

Kelas Sebelumnya

7B

7B

Jenis Kelamin Sebelumnya

Laki-Laki

Laki-Laki

Tanggal Lahir

1991-11-19

Format YYYY-MM-DD

Alamat

Jl Jakarta

Nomor HP Wali Siswa

089608023046

**Simpan**

**Gambar 4.32** Proses Ubah Data Siswa

## Database Siswa

RFID sudah terpakai, gunakan RFID lain

Ubah atau Hapus Data Siswa dengan Mencari NIS

Cari NIS ...

#	NIS	RFID	Nama	Kelas	Gender	Birth	Alamat	No. HP Wali	Aksi
1	1003	6E001092A448	Huda	7B	Laki-Laki	19 November 1991	Jl Jakarta	089608023046	<input type="button" value="✎"/> <input type="button" value="✖"/>
2	1004	6E0009F3E074	Boby	7B	laki-laki	01 December 1991	Yogyakarta	089608023046	<input type="button" value="✎"/> <input type="button" value="✖"/>
3	1005	6E0008571726	Alex	8B	laki-laki	11 December 1993	Yogyakarta	089608023046	<input type="button" value="✎"/> <input type="button" value="✖"/>
4	1006	6E0008DD7FC4	Firman	9B	laki-laki	13 November 1993	Yogyakarta	089608023046	<input type="button" value="✎"/> <input type="button" value="✖"/>

1

**Gambar 4.33** Penanganan Kesalahan Ubah Data Siswa

#### 4. Penanganan Kesalahan Ubah Pengaturan Presensi

Pesan kesalahan akan ditampilkan ketika terdapat data yang belum diisi dan waktu yang dimasukkan pada *form* tidak sesuai dengan prosedur. Penanganan kesalahan ubah pengaturan presensi dapat dilihat pada **Gambar 4.34**.

## Ubah Pengaturan Presensi

Waktu Mulai Kolom Periode HADIR harus lebih awal dari Waktu Selesai Kolom Periode HADIR. ×

Peraturan Untuk mengubah Pengaturan Presensi. Mohon Dibaca!

- Waktu Mulai Kolom Periode HADIR harus lebih awal dari Waktu Selesai Kolom Periode HADIR.
- Waktu Selesai Kolom Periode HADIR harus lebih awal atau sama dengan Waktu Mulai Kolom Periode TERLAMBAT.
- Waktu Mulai Kolom Periode TERLAMBAT harus lebih awal dari Waktu Selesai Kolom Periode TERLAMBAT.
- Waktu Selesai Kolom Periode TERLAMBAT harus lebih awal dari Waktu Mulai Kolom Periode ABSEN.
- Jarak Antara Waktu Mulai Kolom Periode ABSEN dengan Waktu Selesai Kolom Periode TERLAMBAT harus 60 detik.
- Waktu Mulai Kolom Periode ABSEN harus lebih awal dari WAKTU Selesai kolom Periode ABSEN.

Periode HADIR

Status

HADIR

Keterangan SMS

Pertahankan Kedisiplinan Anak Anda

Waktu Mulai

03:00:00 ⌚

Waktu Selesai

02:59:00 ⌚

**Gambar 4.34** Penanganan Kesalahan Ubah Pengaturan Presensi

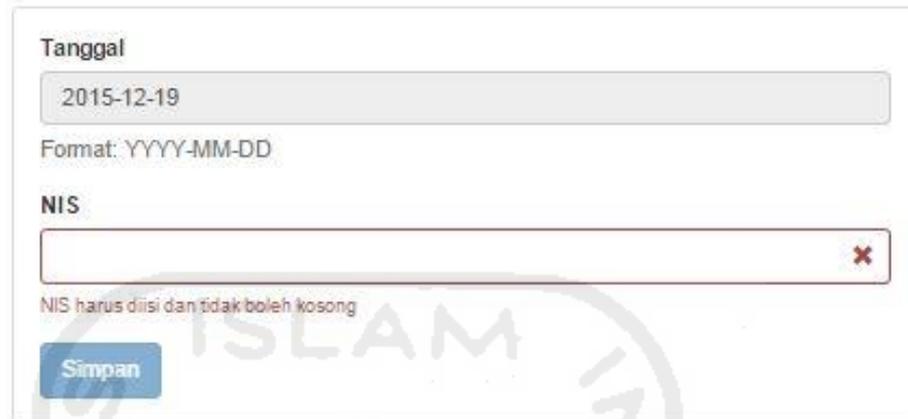
### 5. Penanganan Kesalahan Presensi Manual

Pesan kesalahan akan ditampilkan pada penanganan kesalahan presensi manual ketika :

- a. NIS tidak diisi
- b. Siswa dengan NIS yang telah diberikan sudah melakukan presensi pada hari aktif.
- c. NIS yang diberikan tidak terdaftar.
- d. Waktu sistem tidak berada pada periode presensi.

Tampilan penanganan kesalahan presensi manual dapat dilihat pada **Gambar 4.35**, **Gambar 4.36**, **Gambar 4.37**, dan **Gambar 4.38**.

## Presensi Manual



Tanggal

2015-12-19

Format: YYYY-MM-DD

NIS

NIS harus diisi dan tidak boleh kosong

Simpan

**Gambar 4.35** Penanganan Kesalahan Presensi Manual I

## Presensi Manual



NIS tidak terdaftar

Tanggal

2015-12-19

Format: YYYY-MM-DD

NIS

Simpan

**Gambar 4.36** Penanganan Kesalahan Presensi Manual II

## Presensi Manual

Bukan Periode Presensi ×

Tanggal

2015-12-19

Format: YYYY-MM-DD

NIS

Simpan

**Gambar 4.37** Penanganan Kesalahan Presensi Manual III

## Presensi Manual

Siswa dengan NIS tersebut sudah melakukan Presensi ×

Tanggal

2015-12-19

Format: YYYY-MM-DD

NIS

Simpan

**Gambar 4.38** Penanganan Kesalahan Presensi Manual IV

### 4.3 Analisis Kelebihan dan Kekurangan Sistem

Dari tahapan pengujian sebelumnya telah dilakukan beberapa tahapan bagaimana sistem berjalan. Dengan melihat bagaimana sistem berjalan maka dapat dianalisis kelebihan dan kekurangan dari sistem ini.

#### 4.3.1 Kelebihan Sistem

- a. Sistem ini tidak membutuhkan mikrokontroler sebagai pengolah *input* berupa *tag* RFID. *Input* diolah langsung oleh sistem operasi OpenWrt yang berperan sebagai otak sistem untuk menghasilkan informasi keterangan presensi.
- b. Laporan presensi dapat dihasilkan dan diunduh langsung melalui sistem, tidak dibutuhkan aplikasi lain untuk melakukan konversi tabel menjadi laporan yang dapat di cetak.
- c. Pengaturan waktu yang menentukan keterangan presensi yang didapatkan oleh siswa ketika proses presensi dilakukan bersifat dinamis. Pengaturan tersebut dapat diubah melalui antarmuka *web* sesuai dengan kebijakan sekolah.
- d. Selain dapat diakses oleh pihak sekolah (administrator) dan siswa, Informasi presensi juga dikirimkan kepada orang tua siswa melalui SMS.

#### 4.3.2 Kekurangan Sistem

- a. Dibutuhkan perangkat terpisah berupa laptop atau komputer yang terhubung dengan sistem melalui jaringan nirkabel maupun kabel untuk menampilkan notifikasi berupa informasi keterangan presensi setelah siswa melakukan presensi RFID melalui antarmuka *web*.
- b. Notifikasi yang ditampilkan setelah proses presensi RFID dilakukan kurang informatif. Notifikasi tersebut akan hilang dan diganti dengan notifikasi untuk siswa lain yang juga melakukan presensi RFID.
- c. Pada proses unduh laporan, tidak terdapat fitur untuk mengelompokkan laporan berdasarkan tanggal, bulan, dan tahun tertentu. Hanya laporan pada hari aktif yang dapat diunduh.

- d. Presensi manual dilakukan oleh administrator dengan memberikan *input* berupa NIS satu demi satu, hal tersebut akan memakan banyak waktu.
- e. Dibutuhkan koneksi internet agar waktu dalam sistem dapat tersinkronasi dengan waktu lokal.
- f. Tidak ada pemberitahuan perihal sukses atau gagalnya pengiriman SMS.

