

CHAPTER 4

ANALYSIS AND DESIGN

4.1 Analysis

The purpose of this project is to build an Interactive Voice Response simulation using Asterisk as server and Web Browser as user. To access Interactive Voice Response from web browser, then use API sipml5. This sipml5 can access IVR on web browser for desktop or smartphone, this project is using Mozilla Firefox.

Asterisk is using phpAGI to run a php script, in this php script include a logic about how this program can be run. First, Asterisk will provide welcome message and will prompt the user to input a year class of the user, then Asterisk will temporaryli save the input from user, next system will prompt the user to input one last number on majors code, system will temporaryli save again a input from user. Last, system will prompt the user to input four number in a NIM, then system combines all user input.

Next Asterisk provide a menu in the form of voice, if user touch or press button, client will send the signal as DTMF info. After the server receive a request from user, the server will check in database about data request from user, if data exists or not in database, Asterisk will respond to the user in voice. If the user incorrectly inputs data up to 3 times, Asterisk automaticly hangup the call without approval from user. In this step, it takes several attempts to get the program running correctly.

4.2 Design

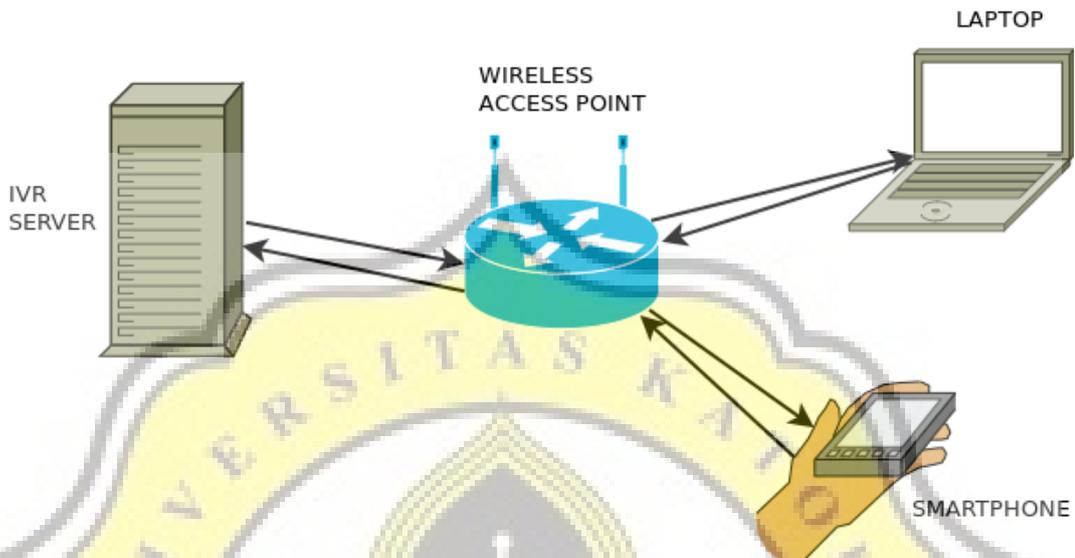


Illustration 4.2.1: Jaringan

This project are using TP-LINK TL-WR840N as a tool to share network from the IVR server, so communication between the IVR server with the client must go through the Access Point in order to connect in one network.

T MATAKULIAH		T HSLSTUDI	
•KDMK_PUS	varchar(8)	•THNAJAR	varchar(4)
•KDMK_JUR	varchar(12)	•KD_MSUJI	char(2)
•NM_MKULIAH	varchar(60)	•NIM	varchar(10)
•SKS	int(2)	•KDMK_PUS	varchar(12)
		•NILAI	char(3)
		•KELAS	char(2)
		•NO_ARSIP	varchar(16)
		•OPERATOR	varchar(15)
		•TANGGAL	datetime
		•KD_JUR	char(2)
		•DOSEN	varchar(20)

Illustration 4.2.2: Database Table

In the database contain 1 primary key on T_MATAKULIAH, while on T_HSLSTUDI contain 4 primary key which is usefull as a deterrnet to the same input/duplicte entry, in this case it is not that the same 2 KDMK_PUS taken by the

same NIM at the same KD_MSUJI and in the same THNAJAR, it can lead to incorrect value calculations, either the value becomes higher in value or becomes lower, this causes chaos in the database.



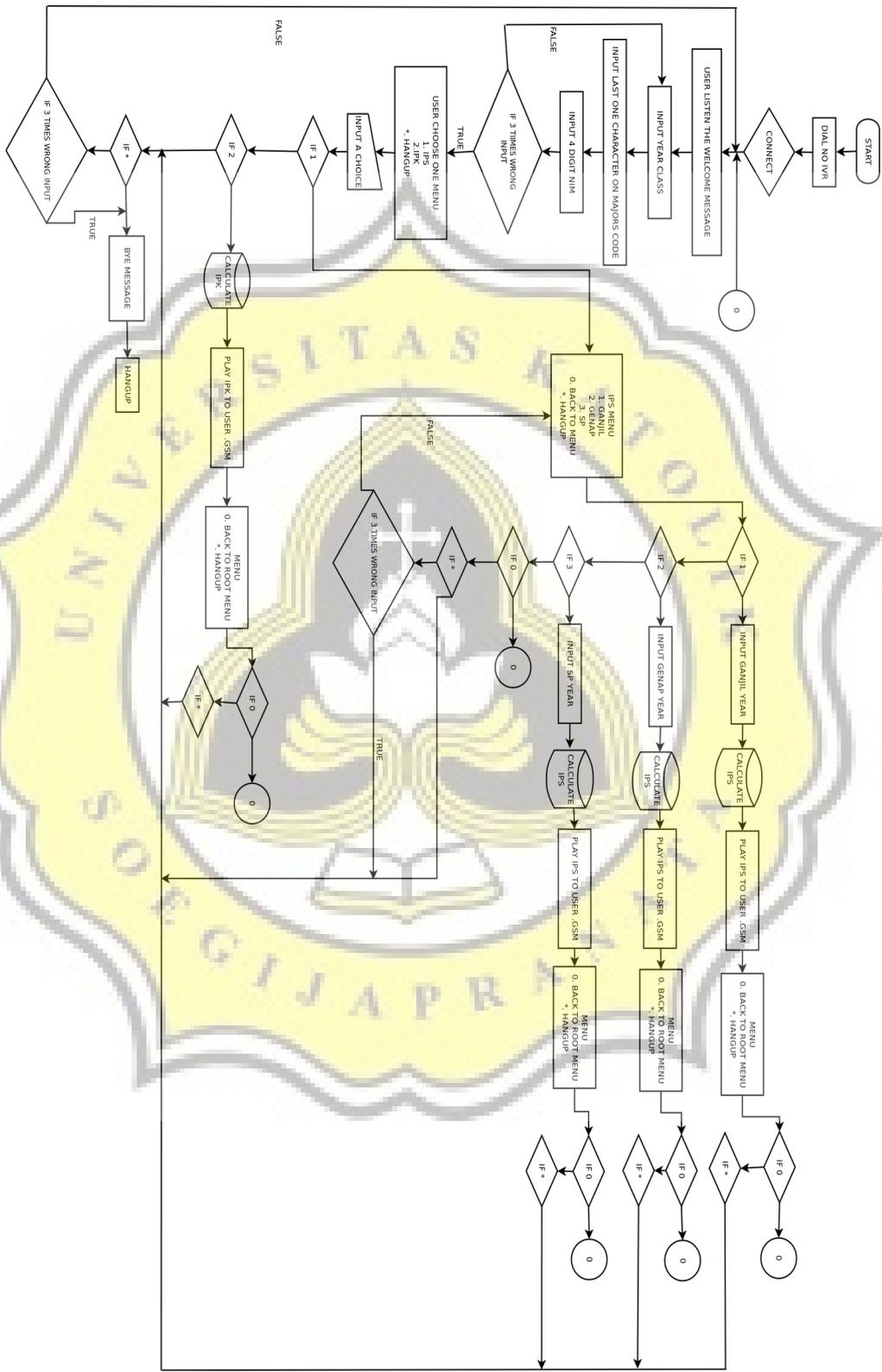


Illustration 4.2.3: IVR Flowchart

From the flowchart above, the logic of the program is when the user dials the IVR dial number, the Asterisk server will receive the call and play the welcome message and server will ask the user to enter the 2 digits of the year that exists in the first two digits of the user NIM, then after the user input is received, the server will ask for 1 digit of the department code, and the last server will ask the user to input the last 4 digits of NIM. After all input is received, the server will combine all the characters that have been entered by the user and check to the database whether the data entered by the user is true or not. If the user provided parameter is incorrect or missing in the database, the server will give the user the chance to input the correct data up to 3 times, if the user still does not input the correct data, the server will automatically hangup the call from the user. If the data entered user is correct, then iVR server will give 2 menu to user, that is menu to check IP Semester or Cumulative IP, if user press number 1, that is Semester IP menu, then user will be given back 3 menu, that is Odd semester, even semester, or other semester, after selecting one of the 3 menus, the server will ask the user to re-enter the semester that has been selected by the user, then the server will check to the database with the user's NIM parameters, odd semester information, even or other, and the year the semester is taken. If the server finds the data in accordance with the parameters already entered by the user, then the server will calculate the existing data on the database and in accordance with the parameters entered by the user into one output of the IP semester, then the output data will be provided to the user in the form of voice. But if the parameters given by the user is wrong or not in the database, the server will give the user the chance to input the correct data up to 3 times, if the user still does not input the exact data, the server will automatically hangup the call from the user. Once the user gets the information that has been requested, the server will give back the option to continue the service back to the main menu or choose to end the call. Then move to the Cumulative IP menu in the IVR main menu, If user press number 2, then the server will check on

the database, and calculate the data requested by the user then provide a response to the user in the form of voice. Then the server will give 2 more options to return to the main menu or immediately end the call.

4.3 Installation

- First install Asterisk by typing command 'apt-get install asterisk' on the terminal(in this project using Debian linux operating system).



Illustration 4.3.1: Install Asterisk

- Then also installize php, apache, mariadb and phpmyadmin which will be useful to run phpAGI by typing in terminal 'apt-get install php apache2 mariadb-server phpmyadmin'.

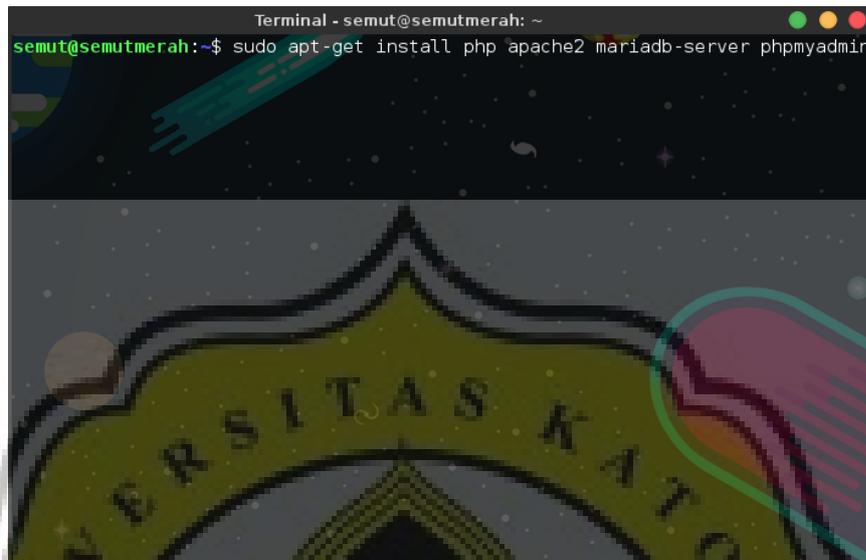


Illustration 4.3.2: Install PHP, Apache, MariaDB, PHPMyAdmin

- First configure 000-default.conf file in '/etc/apache2/sites-available/' directory, edit the contents of 000-default.conf file. Customize the existing directory in the above configuration with the existing agi-bin folder in the directory where Asterisk is installed on a Linux system.

```
<VirtualHost *:80>
```

```
ServerAdmin webmaster@localhost
```

```
DocumentRoot /usr/share/asterisk/agi-bin/
```

```
<Directory /usr/share/asterisk/agi-bin/>
```

```
Options +Indexes +FollowSymLinks
```

```
AllowOverride All
```

```
Order allow, deny
```

```
allow from all
```

```
</Directory>
```

```
</VirtualHost>
```

```

Terminal - semut@semutmerah: ~
7 # value is not decisive as it is used as a last resort host regard
less.
8 # However, you must set it for any further virtual host explicitly
9
10 #ServerName www.example.com
11
12 ServerAdmin webmaster@localhost
13 DocumentRoot /usr/share/asterisk/agi-bin
14
15 <Directory /usr/share/asterisk/agi-bin>
16     Options +Indexes +FollowSymLinks
17     AllowOverride All
18     Order allow,deny
19     allow from all
20 </Directory>
21
22 # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
23 # error, crit, alert, emerg.
24 # It is also possible to configure the loglevel for particular
25 # modules, e.g.
26 #LogLevel info ssl:warn

```

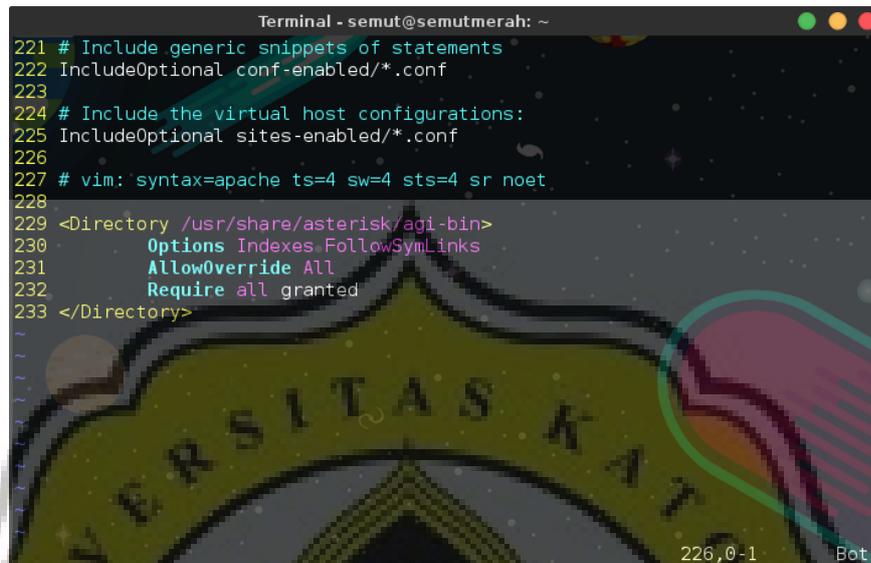
Illustration 4.3.3: Edit 000-default.conf

- Also edit the `apache2` file, `conf` in the directory `/etc/apache2/apache2.conf`.

```

<Directory /usr/share/asterisk/agi-bin/>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>

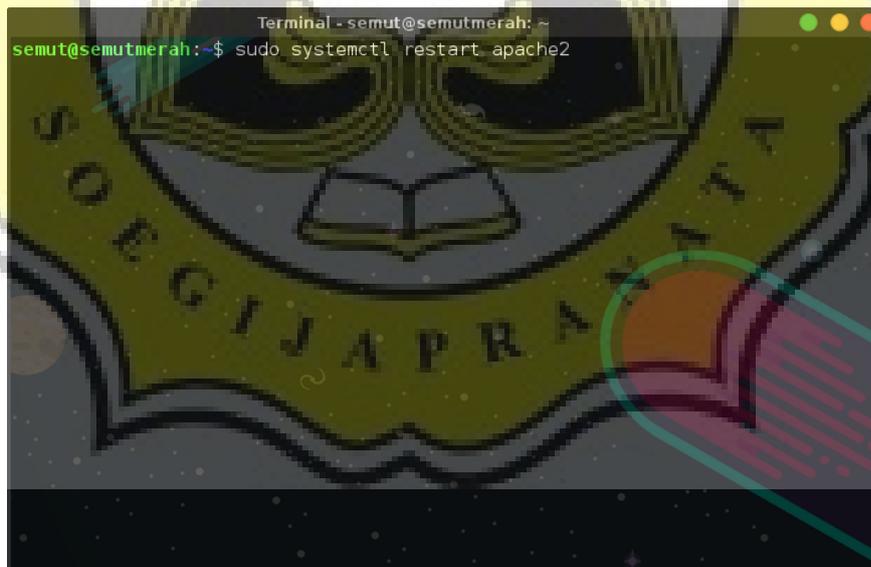
```



```
Terminal - semut@semutmerah: ~
221 # Include generic snippets of statements
222 IncludeOptional conf-enabled/*.conf
223
224 # Include the virtual host configurations:
225 IncludeOptional sites-enabled/*.conf
226
227 # vim: syntax=apache ts=4 sw=4 sts=4 sr noet
228
229 <Directory /usr/share/asterisk/agi-bin>
230     Options Indexes FollowSymLinks
231     AllowOverride All
232     Require all granted
233 </Directory>
```

Illustration 4.3.4: Edit apache2.conf

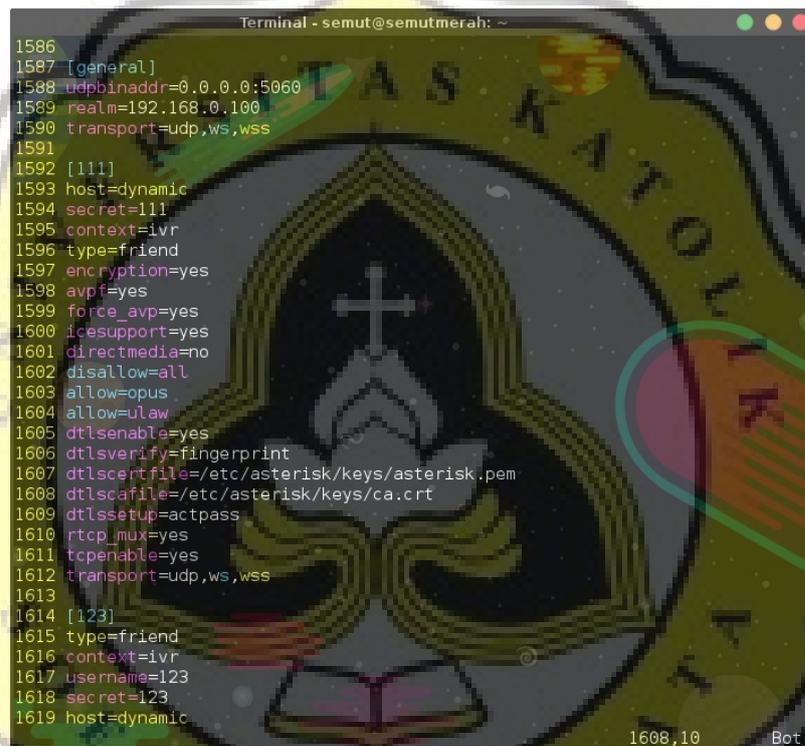
- Then restart apache by running the command ‘sudo systemctl restart apache2’ on the Linux terminal.



```
Terminal - semut@semutmerah: ~
semut@semutmerah:~$ sudo systemctl restart apache2
```

Illustration 4.3.5: Restart apache

- Download phpAGI at url 'https://sourceforge.net/projects/phpagi/files/phpagi/2.20/phpagi-2.20.tgz/download', then extract that file to directory '/usr/share/asterisk/agi-bin'.
- Configure existing sip.conf file in '/etc/asterisk/sip.conf' directory to add:



```

1586
1587 [general]
1588 udpbinding=0.0.0.0:5060
1589 realm=192.168.0.100
1590 transport=udp,ws,wss
1591
1592 [111]
1593 host=dynamic
1594 secret=111
1595 context=ivr
1596 type=friend
1597 encryption=yes
1598 avpf=yes
1599 force_avp=yes
1600 icesupport=yes
1601 directmedia=no
1602 disallow=all
1603 allow=opus
1604 allow=ulaw
1605 dtlsenable=yes
1606 dtlsverify=fingerprint
1607 dtlscertfile=/etc/asterisk/keys/asterisk.pem
1608 dtlscacfile=/etc/asterisk/keys/ca.crt
1609 dtlssetup=actpass
1610 rtcp_mux=yes
1611 tcpenable=yes
1612 transport=udp,ws,wss
1613
1614 [123]
1615 type=friend
1616 context=ivr
1617 username=123
1618 secret=123
1619 host=dynamic
1608,10 Bot

```

Illustration 4.3.6: Add SIP Account

- On line 1587-1590 declare a global variable name [general], with default port for sip connection 5060, then
 1. udpbinding = 0.0.0.0:5060 is used to listen to connections on all IP configured in the system by using port 5060 as the default port of the SIP connection.
 2. realm = 192.168.0.100 is use to authenticate the address use to access Asterisk.

3. transport = udp,wss,ws use to initialize the port to be used, here use is port udp, wss, ws.

- On line 1592-1612 is an account creation which will be user to login to the web socket.

1. [111] symbolizes the number to be used at login.

2. Host = dynamic symbolizes the number to be used at login.

3. context = ivr, used to call ivr context existing in extensions.conf located in the directory '/etc/asterisk/extensions.conf'.

4. Type = friend, used for this account to be dialed or made outbound calls.

5. Encryption = yes, is used to make all activities performed encrypted.

6. Avpf = yes, for outgoing calls or media using avpf profiles, and will reject those not using avpf.

7. Force_avp = yes,so that call incoming calls use avp, including DTLS_SRTP.

8. Directmedia = yes, in order to update the media.

9. Disallow = all, does not allow any access except those to be set.

10. Allow = ulaw, only allow ulaw to acces.

11. Dtlscapable = to enable dtls encryption and used in the activity.

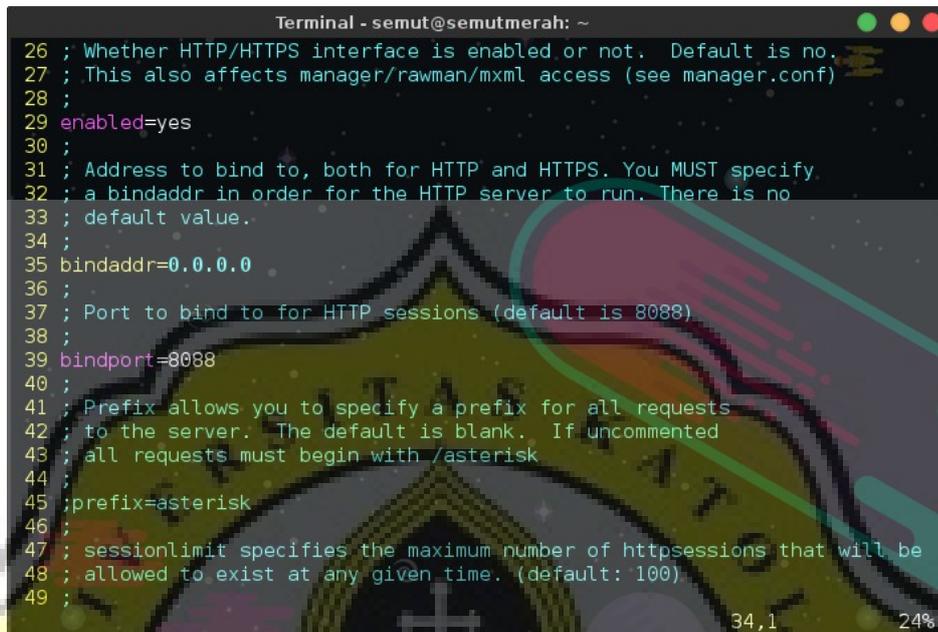
12. dtls_verify = fingerprint, used to make the echo set to dtls.

13. Dtlscertfile = /path, used to access the dtls certificate in the system.

14. Dtlscapable = /path, used to acces the ca certificate in the system.

15. dtls_setup= actpass, allow connection from peer.

16. rtp_mux = yes, used in order for Asterisk to negotiate to use rtp_mux in every media event.

A terminal window titled "Terminal - semut@semutmerah: ~" displaying the configuration for the http.conf file. The text is as follows:

```
26 ; Whether HTTP/HTTPS interface is enabled or not. Default is no.
27 ; This also affects manager/rawman/mxml access (see manager.conf)
28 ;
29 enabled=yes
30 ;
31 ; Address to bind to, both for HTTP and HTTPS. You MUST specify
32 ; a bindaddr in order for the HTTP server to run. There is no
33 ; default value.
34 ;
35 bindaddr=0.0.0.0
36 ;
37 ; Port to bind to for HTTP sessions (default is 8088)
38 ;
39 bindport=8088
40 ;
41 ; Prefix allows you to specify a prefix for all requests
42 ; to the server. The default is blank. If uncommented
43 ; all requests must begin with /asterisk
44 ;
45 ;prefix=asterisk
46 ;
47 ; sessionlimit specifies the maximum number of httpsessions that will be
48 ; allowed to exist at any given time. (default: 100)
49 ;
```

The terminal also shows a cursor at the end of line 49 and a status bar at the bottom right with "34,1" and "24%".

Illustration 4.3.8: http.conf part 1

1. Line 29 used to activated pjsip on Asterisk.
2. Line 35 used to set bindaddr to 0.0.0.0
3. Line 39 used for standard ports used 8088.

```
Terminal - semut@semutmerah: ~
71 ; default page.
72 ; Syntax: redirect=<from here> <to there>
73 ; For example, if you are using the Asterisk-gui,
74 ; it is convenient to enable the following redirect:
75 ;
76 ; redirect = / /static/config/index.html
77 ;
78 ; HTTPS support. In addition to enabled=yes, you need to
79 ; explicitly enable tls, define the port to use,
80 ; and have a certificate somewhere.
81 tlseable=yes ; enable tls - default no.
82 tlsbindaddr=0.0.0.0:8089 ; address and port to bind to - default is bind
  daddr and port 8089.
83 ;
84 tlscertfile=/etc/asterisk/keys/asterisk.pem ; path to the certificate fil
  e (*.pem) only.
85 tlsprivatekey=/etc/asterisk/keys/wss.pem ; path to private key file (*.
  pem) only.
86 ; If no path is given for tlscertfile or tlsprivatekey, default is to look
  in current
87 ; directory. If no tlsprivatekey is given, default is to search tlscertfil
  e for private key.
88 ;
@@@ 79,26 64%
```

Illustration 4.3.9: http.conf part2

4. Line 81 used to enable tls.
5. Line 82 used for the address used to be flexible and use port 8089.
6. Line 84 and 85 used to access the certificate files used.

```

Terminal - semut@semutmerah: ~
1073
1074 [transport-wss]
1075 type=transport
1076 protocol=wss
1077 bind=0.0.0.0
1078
1079 [111]
1080 type=aor
1081 max_contacts=1
1082 remove_exiting=yes
1083
1084 [111]
1085 type=auth
1086 auth_type=userpass
1087 username=111
1088 password=111
1089
1090 [111]
1091 type=endpoint
1092 aors=111
1093 auth=111
1094 use_avpf=yes
1095 media_encrypt=dtls
1096 dtls_ca_file=/etc/asterisk/keys/ca.crt
1097 dtls_cert_file=/etc/asterisk/keys/asterisk.pem
1098 dtls_verify=fingerprint
1099 dtls_setup=actpass
1100 ice_support=yes
1101 media_use_received_transport=yes
1102 rctp_mux=yes
1103 context=ivr
1104 disallow=all
1105 allow=opus
1106 allow=ulaw
1106,10 Bot

```

Illustration 4.3.10: pjsip.conf

1. Line 1074-1077 is used to initialize the use of transport-wss on an Asterisk connection. With type = transport, protocol = wss, and also bind = 0.0.0.0
2. Line 1079-1082 used for set type = aor, max_contacts = 1, and also remove_exiting = yes.
3. Line 1084-1088 used for set username and password use on account 111.
4. Line 1090-1106 used to set the endpoint and then the same set that exists in http.conf then add the allow opus codec at the end of the line.

Finally, download also sipml5 which will be used as a demo via web browser either from pc browser in the following url

<https://github.com/DoubangoTelecom/sipml5/archive/master.zip>. Then extract and move to directory '/usr/share/asterisk/agi-bin/'.

