CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Implementation

Installation of Linux Ubuntu

1. Choose the desired language on Linux installation.
2. Ubuntu will check the available drive space and the internet connection network.
3. Choose the instalation type, “Something else”, to configure the harddisk capacity which to use on the system.
4. Next, apply the swap partition space and mount point “/” on Linux instalation.

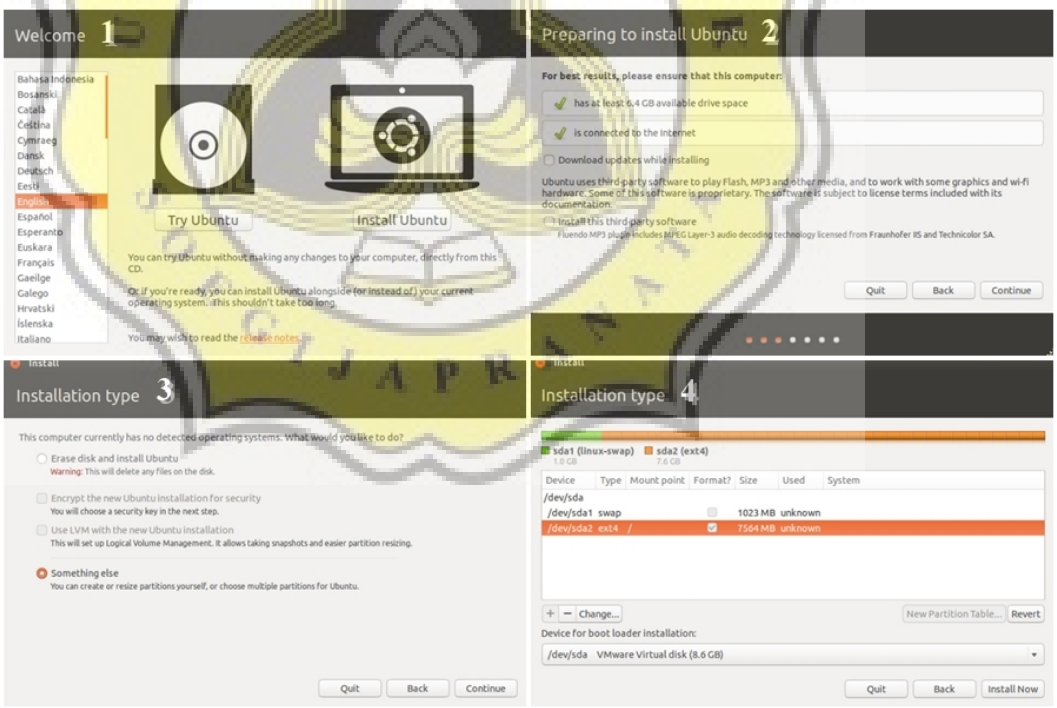

Illustration 5 1: Linux Installation

5. Choose Indonesia as the state options.

6. Choose “United States” for the keyboard layout.

7. Input the user identity and configuring the password to ubuntu user login.

8. System starts the installation process.

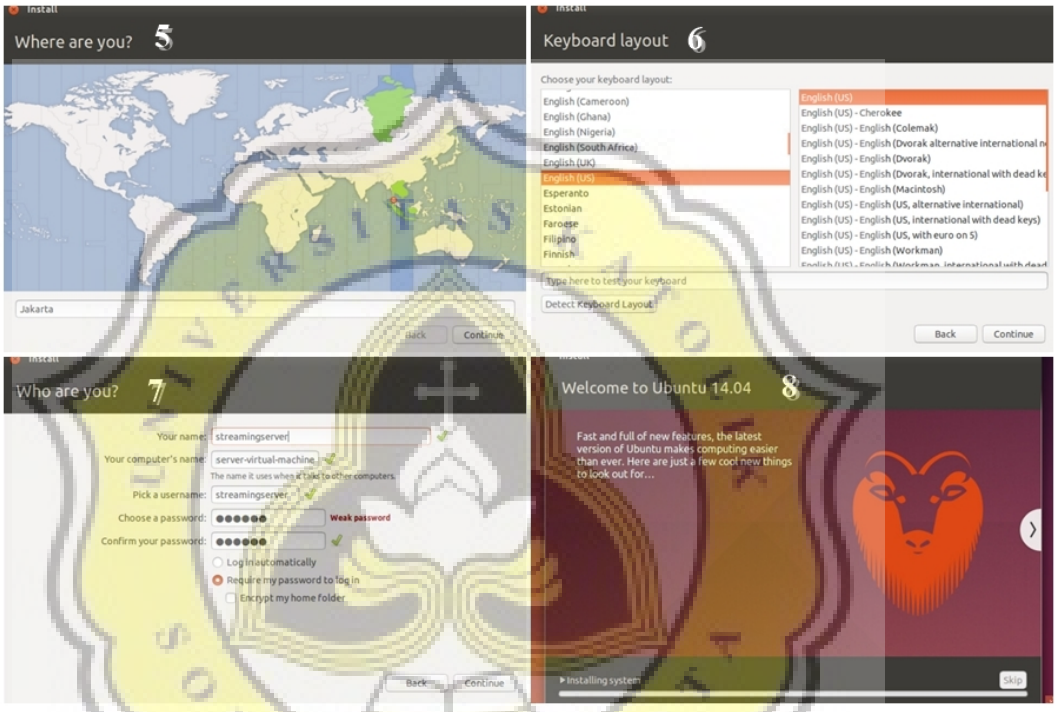

Illustration 5 2: Linux Installation

9. User needs to restart the computer after installation.

10. User inputs the pre-arranged username and password.

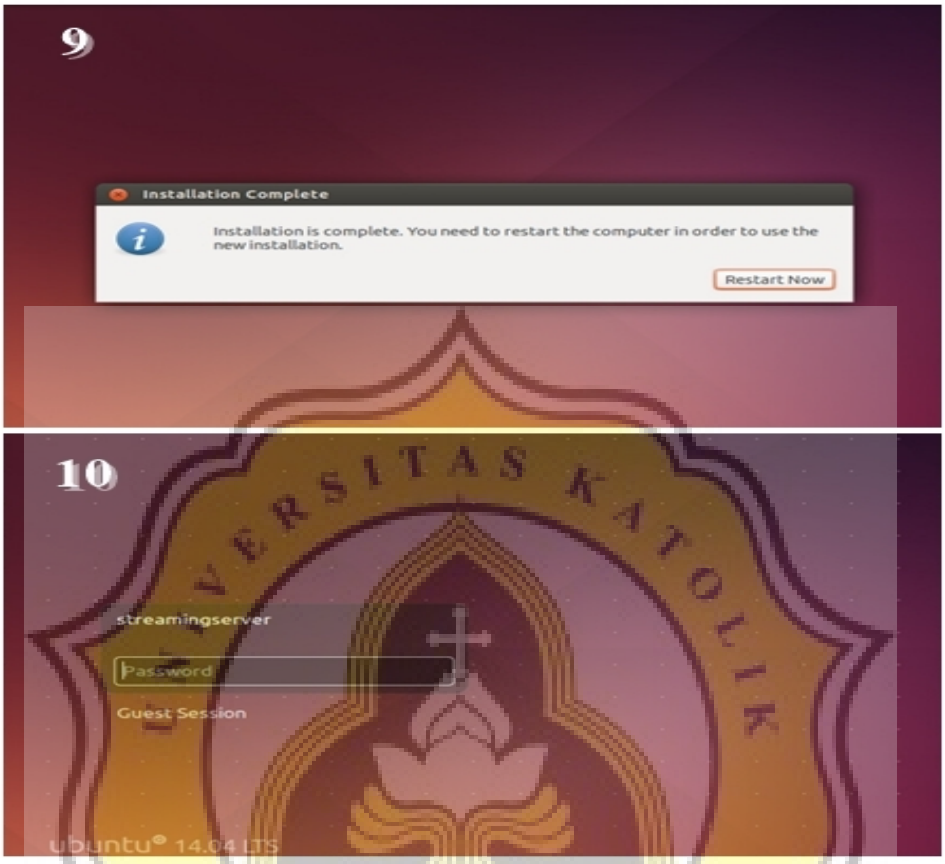

Illustration 5 3: Linux Installation

Installation of Nginx Server[1]

1. Install the packages required to create a Nginx server.

   sudo apt-get install build-essential libpcre3 libpcre3-dev libssl-dev

2. Create a working folder that is used for the preparation between the nginx server and the RTMP module.

   Mkdir ~/working

1 https://www.vultr.com/docs/setup-nginx-rtmp-on-ubuntu-14-04 Diakses pada 6 September 2017, pukul 13.10)

3. Go into the working directory for storing process of the Nginx server files and RTMP Module.

   cd ~/working

4. Download the nginx server version 1.7.5 from the nginx.org website.

   wget http://nginx.org/download/nginx-1.7.5.tar.gz

5. Download the RTMP module package to support the Nginx server.

   wget https://github.com/arut/nginx-rtmp-module/archive/master.zip

6. Download the Unzip program to access the zip files on the Nginx server and the RTMP Module.

   sudo apt-get install unzip

7. Extract the file of nginx.tar.gz in the working folder.

   tar -zxvf nginx-1.7.5

8. Extract the file of master.zip in the working folder.

   unzip master.zip

9. Enter the nginx folder to process the installation.

   cd nginx-1.7.5

10. Check the installation of nginx server by completely combining the RTMP module and the http ssl module.

    ./configure --with-http_ssl_module –add-module=../nginx-rtmp-module-master

11. Process to build a pre-aggregated program.

    make

12. Process to install nginx server and RTMP module into Ubuntu linux system.

    sudo make install

13. Install the Nginx Init system to facilitate the turning on or off the Nginx server.

    sudo wget https://raw.github.com/JasonGiedymin/nginx-init-ubuntu/master/nginx -O /etc/init.d/nginx

14. Change the permissions on the system to facilitate the opening and running the server nginx program.

    sudo chmod +x /etc/init.d/nginx

15. Set the Nginx service to run automatically when turning on the computer.

    sudo update -rc.d nginx defaults

16. Input the command to run the Nginx server.

    service nginx start

17. When nginx is successfully installed, the user can open the browser and type a localhost or ip address on the computer network.

Illustration 5 4: The display of nginx server localhost

Configuration of Nginx & RTMP server[2]

1. The first step is log in as the root user and still be on the server computer, it is intended to be able to change or edit files that have permission.

   sudo su

2. Checking the nginx server installation which is conducted on a folder containing the previous linux installation then adding an RTMP module that contains the http stub status that works for live streaming through a web browser.

   ./configure --with-http_ssl_module --with-http_stub_status_module --add-module=../nginx-rtmp-module-master

3. The next process is arrange the modules which have been designed in advance by typing the command

   make

4. The next process is installing the modules which have been arranged by typing command **make install**.

5. Go into the nginx server configuration folder located in the usr/local/nginx/conf. Directory.

   cd usr/local/nginx/conf

6. The command works for duplicating the nginx.conf file in order to re-functionalized the program if error is occured.

   cp /usr/local/nginx/conf/nginx.conf /usr/local/nginx/conf/nginx.conf.original

2 https://www.vultr.com/docs/setup-nginx-rtmp-on-ubuntu-14-04 accessed on 21 September 2017, pukul 10.35)

7. Inside the usr/local/nginx/conf folder, create a new configuration file with the name nginx.conf so the server can run more stable and smooth.

   gedit nginx.conf

8. Create a new nginx configuration by adding multiple commands that function as a recipient of a media server application. The ip address for the server needs to be set in order to be accessible through the client/ user web browser.

   worker_processes 1;
   error_log logs/error.log debug;
   events {
   worker_connections 1024;
   }

   explanation: **worker_proses 1;** is the adjustment of processor cores when the computer starts up. The explanation of **error_log logs/error.log debug;** is a storage file that keeps error reports in nginx configuration. The command of **worker_connections 1024;** is used to find out the number of clients that can access that come to 1024 clients.

   rtmp {
   server {
   listen 1935;
   allow play all;

   Explanation: **rtmp{** is a command used to store all blocks of rtmp settings. S**erver** command **{** declaration of rtmp block settings. **Listen 1935** command**;** is used to find out which port will be used in streaming server which is 1935 servers. **Allow play all** command**;** is

used for streaming video to be played on all ip address with 1935 ports.

#creates our "live" full-resolution HLS videostream from our incoming encoder stream and tells where to put the HLS video manifest and video fragments

```
application live {
allow play all;
live on;
record all;
record_path /video_recordings;
record_unique on;
hls on;
hls_nested on;
hls_path /HLS/live;
hls_fragment 10s;
```

Explanation: **"#creates our "live" full-resolution HLS videostream from our incoming encoder stream and tells where to put the HLS video manifest and video fragments"** is used to support HLS streaming applications with high screen resolution. **Application live** command **{** is used to run the live streaming smoothly. **allow play all;** is aimed to run the video on the web browser. **Live on;** in the nginx configuration is aimed to see the video live on the web browser.

Explanation: **record all;** is used to record video that is played on the applications with sounds in it. **Record_path video_recordings;** is a place to keep and set the video recordings. **Record_unique on;** is used to input the time in the video when the recording begins. **Hls on;** is to turn on access and to display video on the web browser, if the setting is turned off, the video cannot be showed in the web browser. **Hls_nested on;** to create playlist on hls video setting to be played while streaming

runs. **Hls_path /HLS/live;** is to set video playlist on the hls streaming. **Hls_fragment 10s;** is to set the number of encryption fragments on the hls created at the beginning of the stream to the end.

#creates the downsampled or "trans-rated" mobile video stream as a 400kbps, 480x360 sized video

Explanation: **"#creates the downsampled or "trans-rated" mobile video stream as a 400kbps, 480x360 sized video"** is a command to make the video streaming accessible through phone with a speed of 400kps and size of 480x360 according to the size of the web browser screen on the phone.

exec ffmpeg -i rtmp://192.168.41.66:1935/$app/$name -acodec copy -c:v libx264 -preset veryfast -profile:v baseline -vsync cfr -s 480x360 -b:v 400k maxrate 400k -bufsize 400k -threads 0 -r 30 -f flv rtmp://192.168.41.66:1935/mobile/$;

}

#creates our "mobile" lower-resolution HLS videostream from the ffmpeg-created stream and tells where to put the HLS video manifest and video fragments

```
        application mobile {
        allow play all;
        live on;
        hls on;
        hls_nested on;
        hls_path /HLS/mobile;
        hls_fragment 10s;
}
```

Explanation: **“exec ffmpeg -i rtmp://192.168.41.66:1935/$app/ $name -acodec copy -c:v libx264 -preset veryfast -profile:v baseline -vsync cfr -s 480x360 -b:v 400k maxrate 400k -bufsize 400k -threads 0 -r 30 f flv-rtmp://192.168.41.66:1935/mobile/$;”** is to run video with ffmpeg codec, libx264 and can set ip-address on the server by adding the existing port on the web server nginx, in addition to set the video buffer size when connected online or off line. This setting can also manage access to streaming video via mobile by setting up the ip-address and the port used by nginx server.

```
#allows you to play your recordings of your live streams using a URL like "rtmp://my-ip:1935/vod/filename.flv"
        application vod {
                play /video_recordings;
        }
}
}
```

Explanation: **“#allows you to play your recordings of your live streams using a URL like "rtmp://my-ip:1935/vod/filename.flv"** is a command to play the record on the ip-address that has been specified with the stored file. Explanation of **“ application vod { “** is used to recognize the type of application used in recording the streamed video. Explanation of **“ play /video_recordings; “** is to play the video located in the determined file.

```
http {
include       mime.types;
default_type  application/octet-stream;
```

server {

listen 80;

server_name 192.168.41.66;

Explanation: ip-address is used and it has been set up with 192.168.41.66 server with listen 80 which means the http port and the existing http port are suitable.

#creates the http-location for our full-resolution (desktop) HLS stream - "http://192.168.41.66/live/true/index.m3u8"

```
        location /live {
        types {
                application/vnd.apple.mpegurl m3u8;
        }
        alias /HLS/live;
        add_header Cache-Control no-cache;
}
```

Explanation: **“ #creates the http-location for our full-resolution (desktop) HLS stream-"http://192.168.41.66/live/true/index.m3u8" ”** is the server/ live ip-address setting by adding the existing strean-key in the obs application setting, and index.m3u8 as a streaming file type. Explanation of **“ location /live { “** to set up the storage location of the streaming file. Explanation of: “application/vnd.apple.mpegurl m3u8;” is for supporting m3u8 streaming file. Explanation of **“alias /HLS/live; “**for a live streaming http-based storage location. Explanation of **“add_header Cache-Control no-cache;”** is to set the video cache that will be played.

#creates the http-location for our mobile-device HLS stream - "http://my-ip/mobile/my-stream-key/index.m3u8"

```
        location /mobile {
        types {
        application/vnd.apple.mpegurl m3u8;
}
        alias /HLS/mobile;
        add_header Cache-Control no-cache;
}
```

Explanation: **"#creates the http-location for our mobile-device HLS stream - "http://my-ip/mobile/my-stream-key/index.m3u8"** is used to manage the live streaming via android and ios by adjusting the ip-address according to server, and adding stream-key that has been set up according to the obs application.

#allows us to see how stats on viewers on our Nginx site using a URL like: "http://my-ip/stats"

```
        location /stats {
        stub_status;
        }
```

Explanation: **"#allows us to see how stats on viewers on our Nginx site using a URL like: "http://my-ip/stats"** is to see who have been signed in and viewed the streaming videos by accessing ip-address on the server.

#allows us to host some webpages which can show our videos: "http://192.168.41.66/my-page.html"

```
        location / {
        root   html;
        index  index.html index.htm;
    }
}
}
```

Explanation: **"#allows us to host some webpages which can show our videos: "http://192.168.41.66/my-page.html"** is used as a command to display a streaming page by entering the ip-address on the server and the name of the html page.

9. After the nginx confiiguration is compiled, the user needs to restart the device to maximize the configuration.

   service nginx restart

Install the Open Broadcast Studio (OBS)

1. First add the ffmpeg repository package, because the official ubuntu repository does not provide one.

   sudo add-apt-repository ppa:kirillshkrogalev/ffmpeg-next

2. "apt-get update" is aimed to update the existing applications in ubuntu and fix the problematic apps. While "apt-get install" is aimed to install ffmpeg into ubuntu system.

   sudo apt-get update && sudo apt-get install ffmpeg

3. Add obs studio repository into the system since the official ubuntu repository does not provide the applications.

   sudo add-apt-repository ppa:obsproject/obs-studio

4. Write an update command to update the applications and to fix the problematic apps, and write the install to add obs program into the ubuntu system.

   sudo apt-get update && sudo apt-get install obs-studio

Configuration of Open Broadcast Studio(OBS)

1. Open the obs studio application in ubuntu.

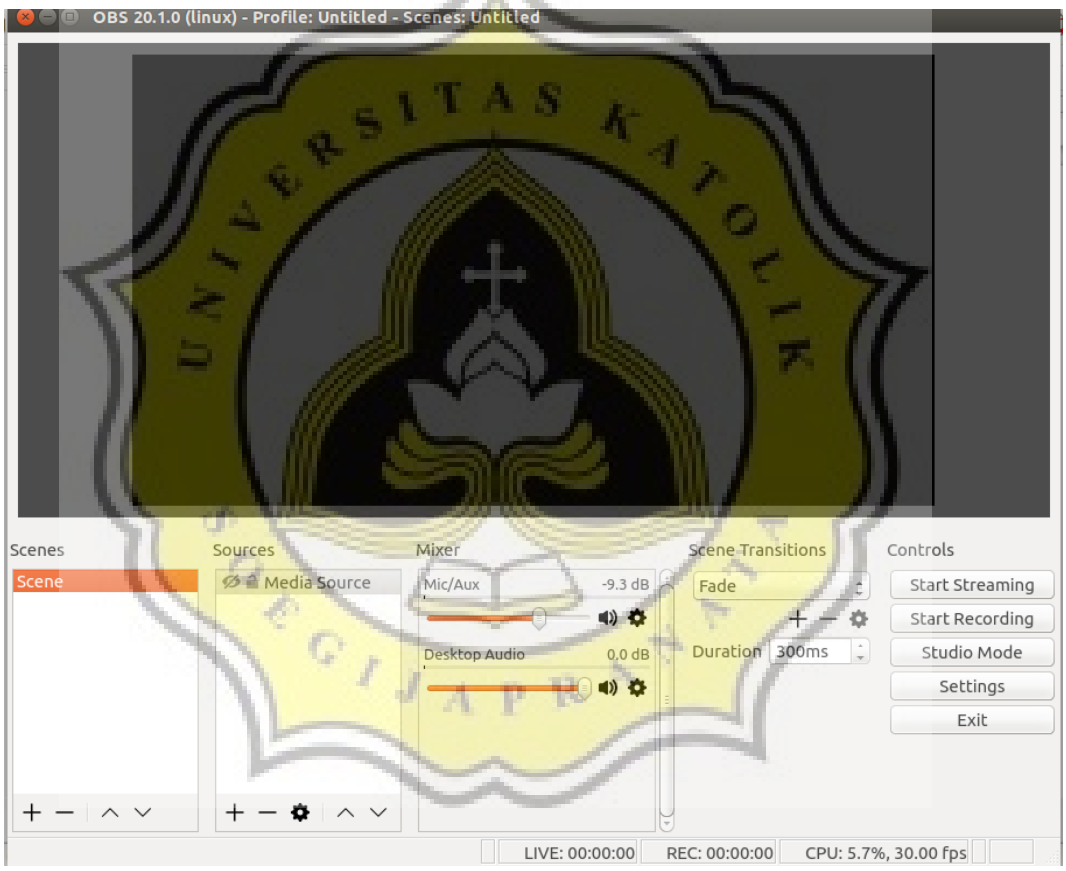

Illustration 5 5: The display of OBS-studio

2. Click “setting” to go into the server settings that will be used in the streaming.

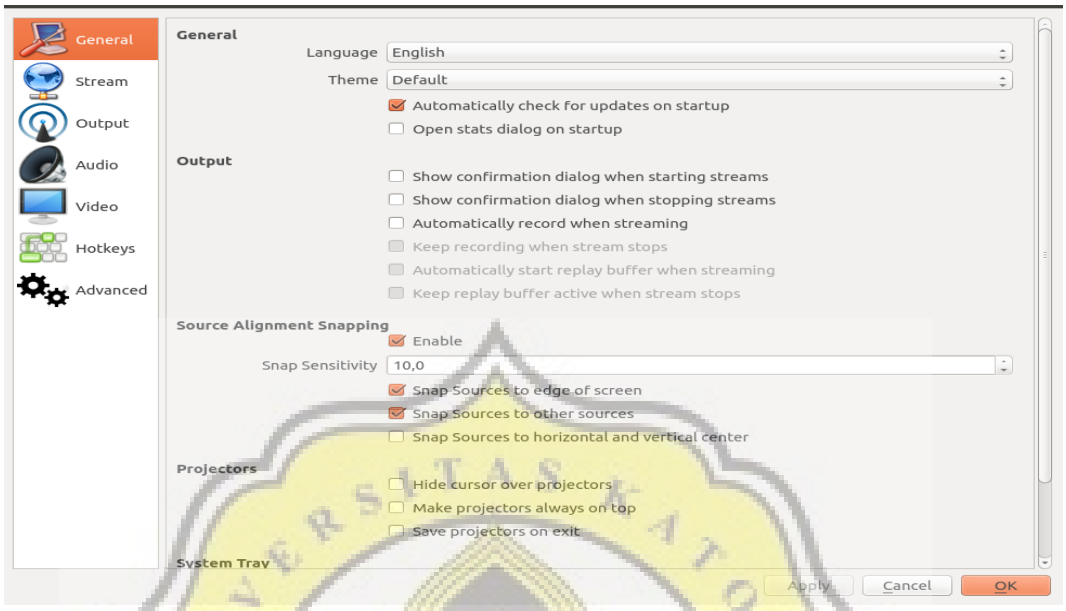

Illustration 5 6: The setting of obs-studio

3. Select the stream panel, click the stream type and select Custom Streaming Server. Set the URL that connect to rtmp server by typing the ip address of nginx server plus “/live” as a sign in entering the html file. Then set the stream key by typing any desired word, for example “true”. If the stream key is not visible, click show then okay to return to the home page.

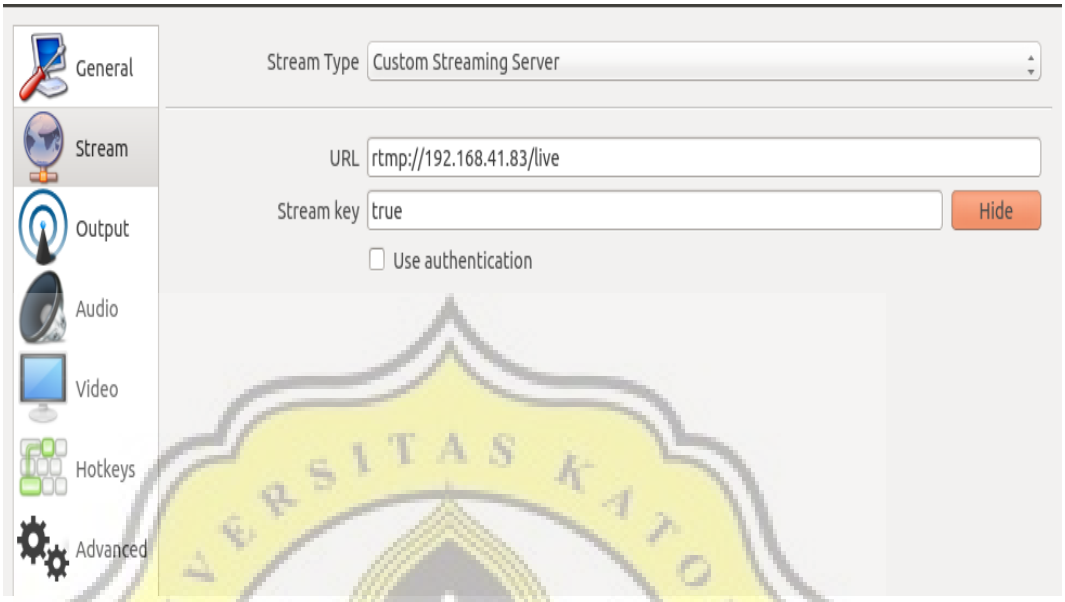

Illustration 5 7: The stream of Obs-studio

4. Add video media into the source panel by selecting the media source on the "+" button, select "create new" and create your own name, then click ok. After that, the media source page will pop up, then select "browse" to find the video that will be streamed, click ok.

Illustration 5 8:
Sources of Obs-studio

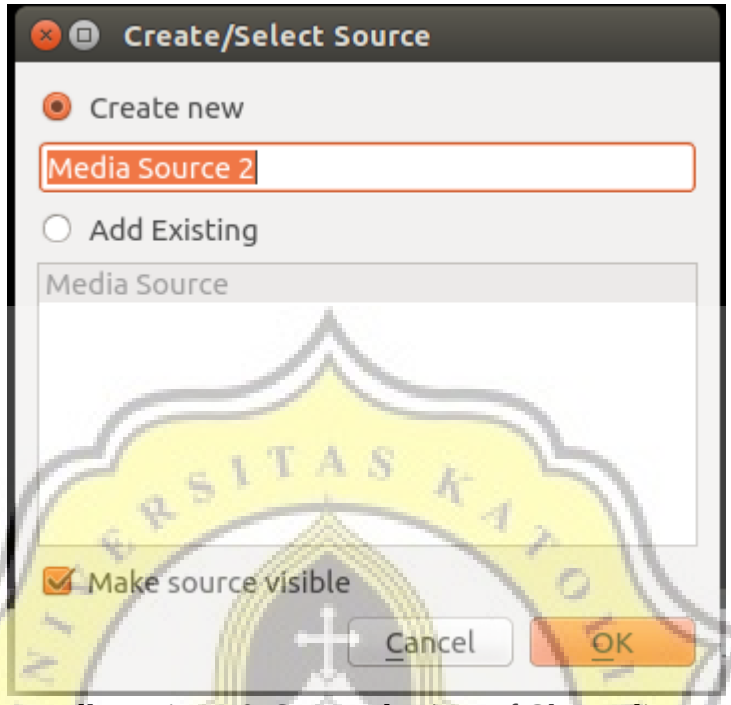

Illustration 5 9: Scene selections of Obs-studio

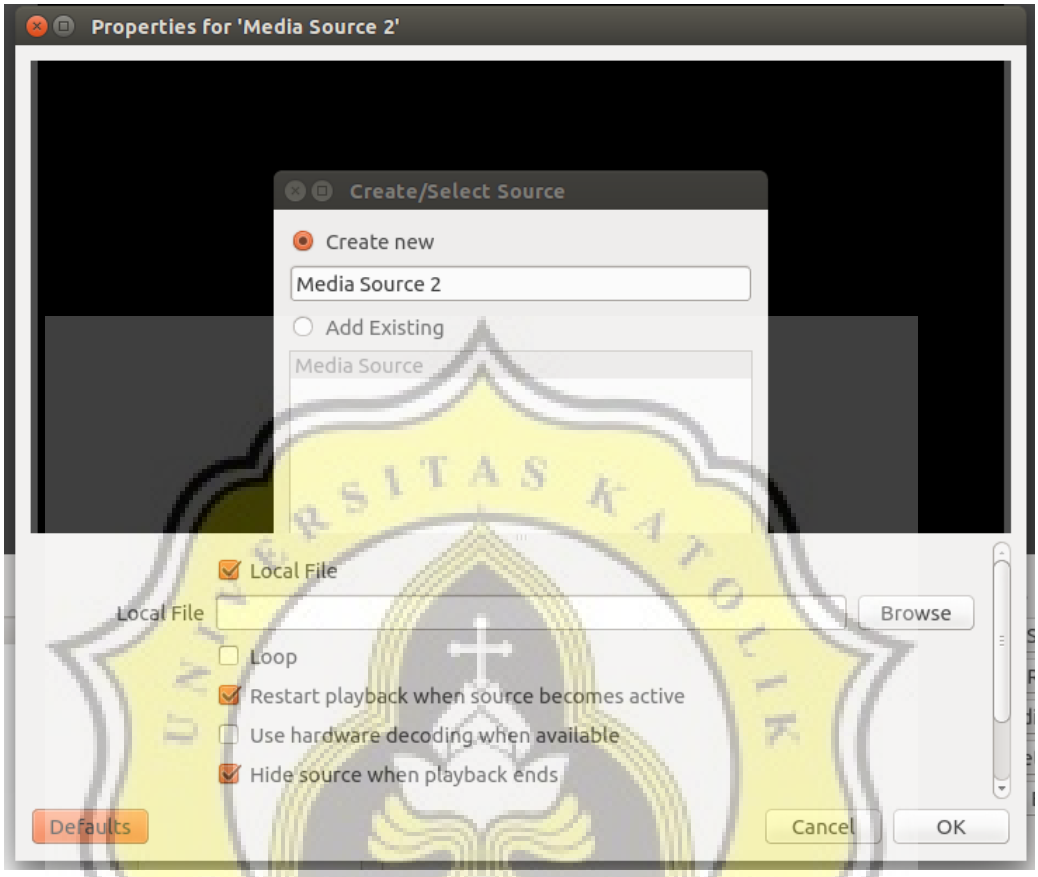

Illustration 5 10: Streaming Video Selection

5. After inserting the video, select start streaming and the video streaming will be running.

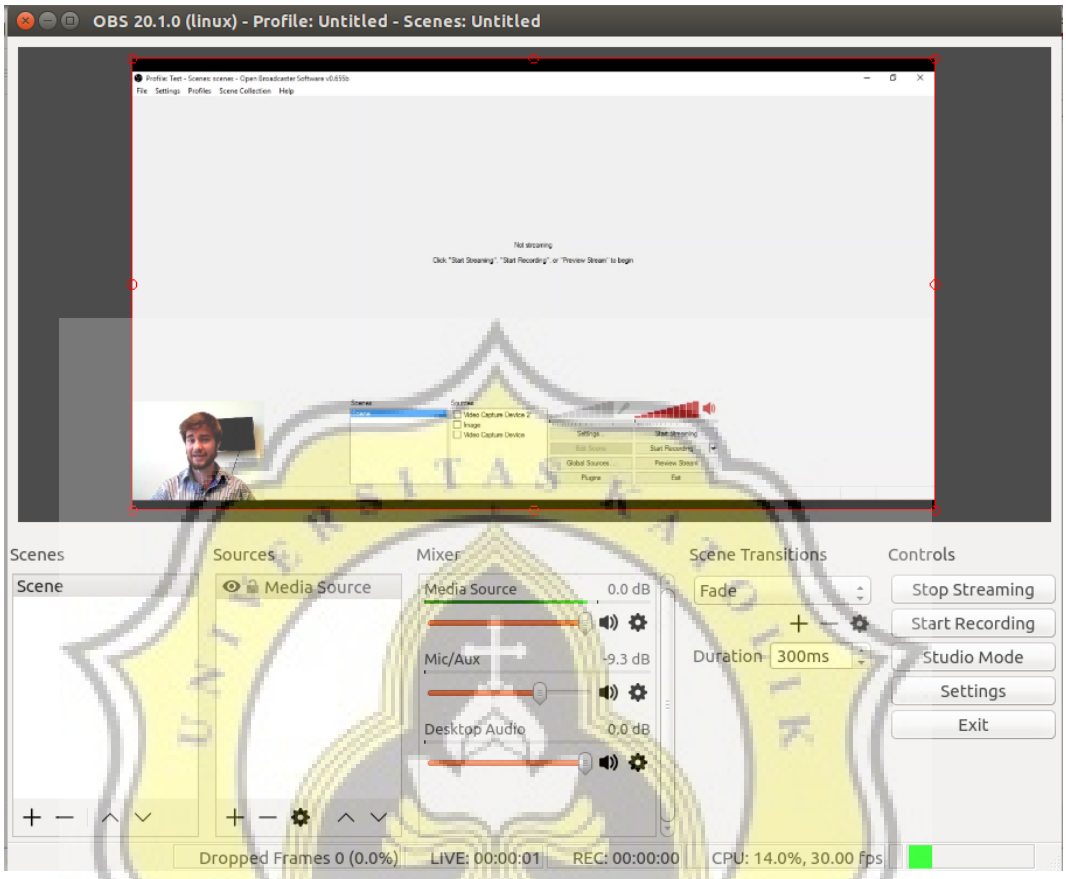

Illustration 5 11: The display of Obs-studio streaming

Display Html Web Browser results

1. By inserting the server's ip address into a web browser on devices such as computers, laptops, and smartphones with android or ios system, users can see the streaming video.

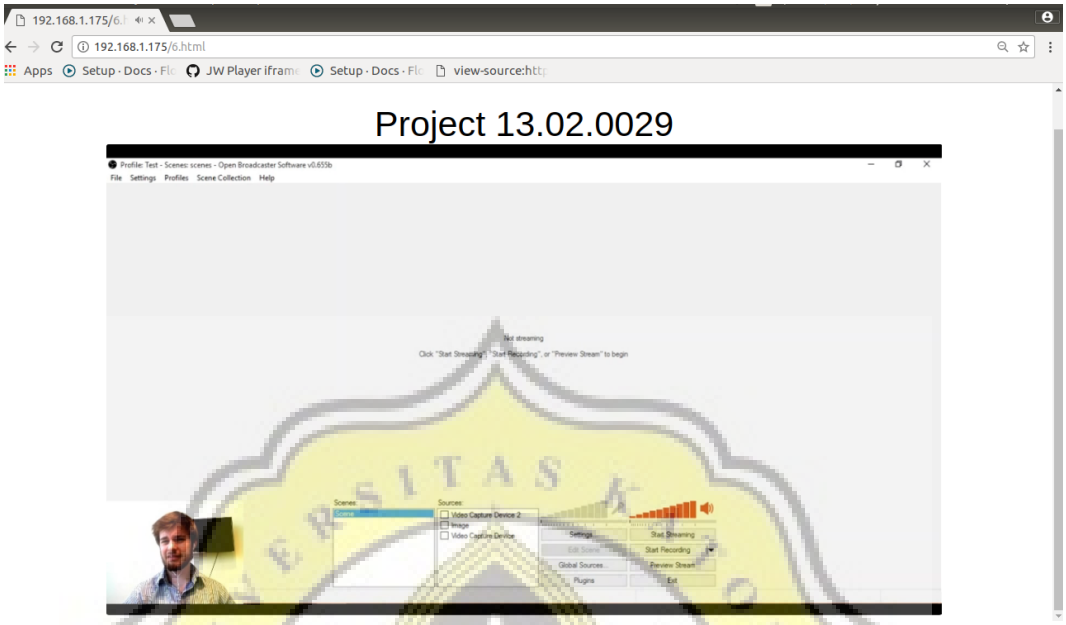

Illustration 5 12: Video streaming on PC

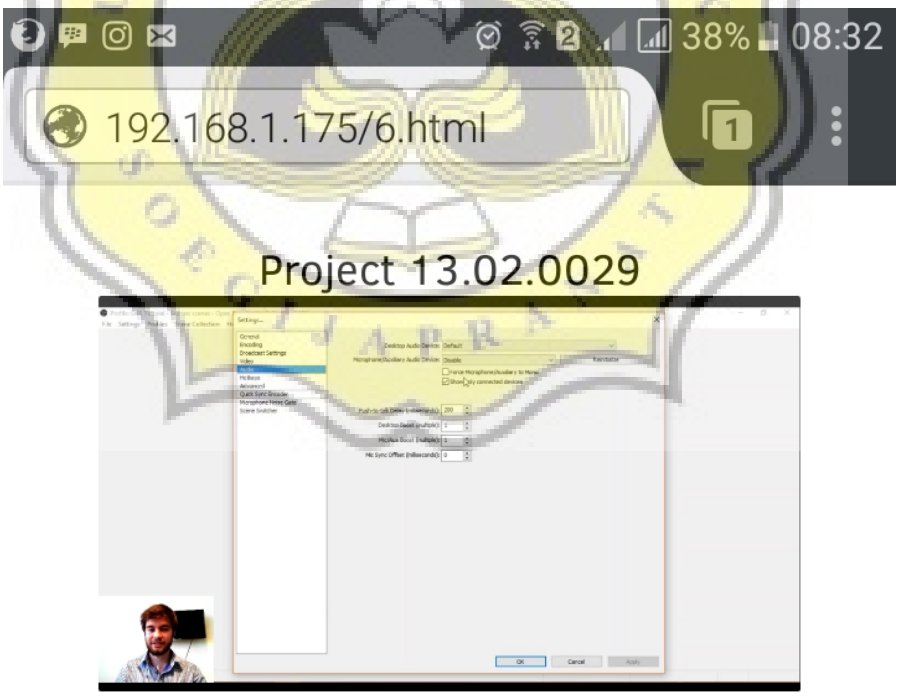

Illustration 5 13: Video streaming on android or ios system.

5.2 Testing

Testing a streaming video project is done by inputing the server ip address into html files, obs programs, and web browser on the devices.

```
<!DOCTYPE html>
    <head>
        <!-- player skin -->
      <link rel="stylesheet" href="flowplayer/skin/skin.css">
```

The script above is used to to set the theme that will be used in the flowplayer.

```
    <!-- site specific styling -->
      <style>
      body { font: 12px "Myriad Pro", "Lucida Grande", sans-serif; text-align: center; padding-top: 5%; }
      .flowplayer { width: 80%; }
      </style>
```

The script above is used to set the display in the flowplayer and the position of the video to be played.

```
    <!-- Flowplayer depends on jquery 1.7.2+ for video tag based installations -->
      <script src="//code.jquery.com/jquery-1.12.4.min.js"></script>
```

The command above is used to enter a jquery file as a media player in the video.

```
    <!-- Flowplayer library -->
      <script src="//releases.flowplayer.org/7.2.1/flowplayer.min.js"></script>
```

The command above is used to add additional packages owned by the flowplayer.

```
<!-- The hlsjs plugin (light) for playback of HLS without Flash in modern browsers -->
  <script src="//releases.flowplayer.org/hlsjs/flowplayer.hlsjs.light.min.js"></script>
</head>
```

The script command above is used as a replacement plugin for flash video.

```
<body>
<font size="10">Project 13.02.0029</font><br>
        <div data-live="true" data-ratio="0.5625" data-share="false" class="flowplayer">
        <video controls autoplay>
            <source type="application/x-mpegurl" src="http://192.168.1.175/live/true/index.m3u8">
        </video>
</div>
</body>
```

The command above is used to set the ip-address server and to set the stream-key on OBS.

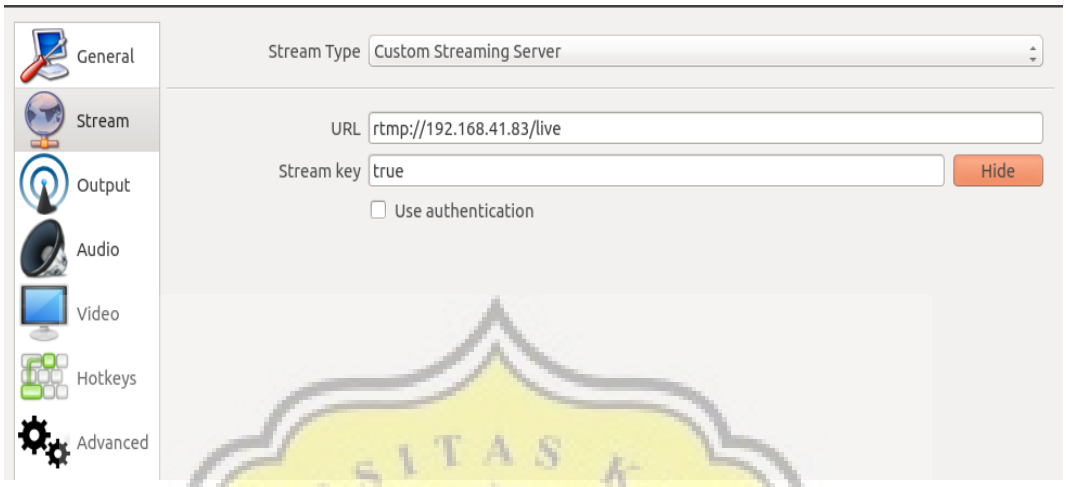

Illustration 5 14: The ip server address and stream key on Obs

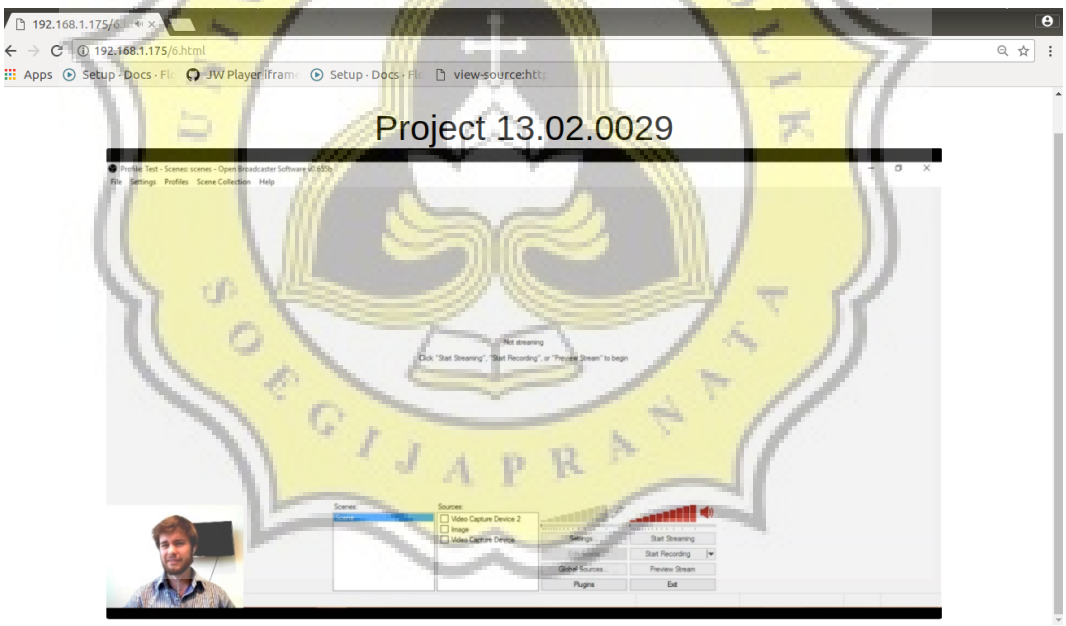

Illustration 5 15: The ip server address on web browser.

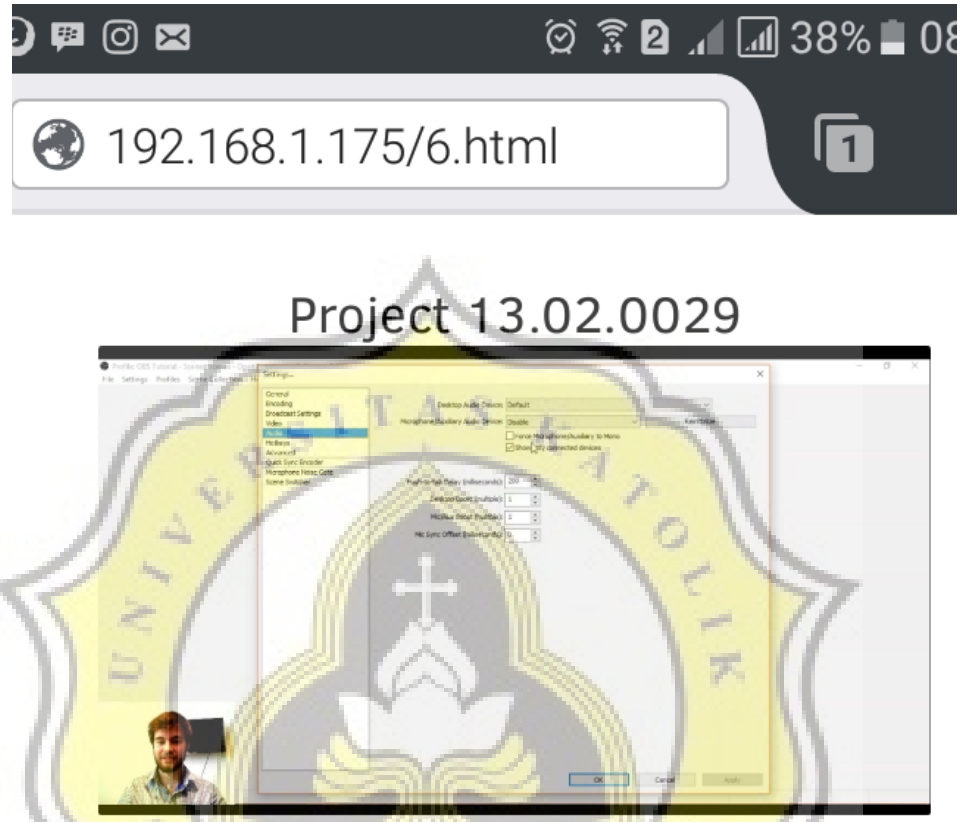

Illustration 5 16: The ip server address on smartphones