

Smart Music Study Companion

Presentasi Akhir ET4107

Thirza Nabila Syafriady (18117011)
M. Naufal Kurniawan (18117012)



Thirza Nabila Syafriady (18117011)



M. Naufal Kurniawan (18117012)

Anggota

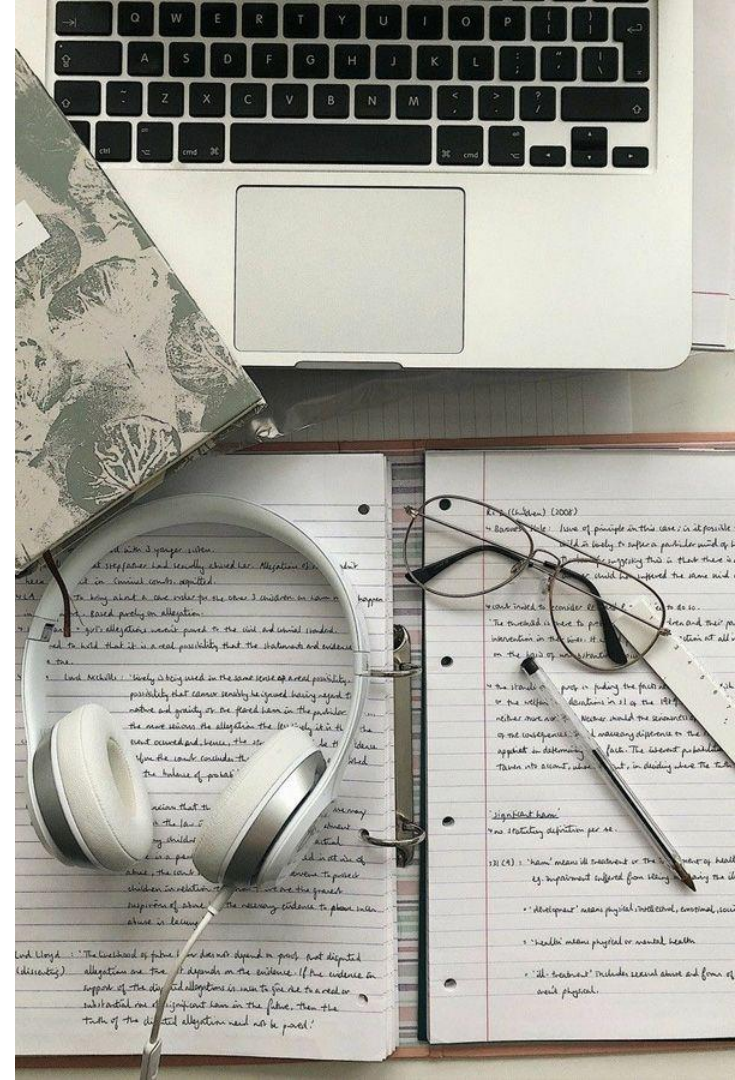
Outline

- Latar Belakang
- Fitur dan Deskripsi Alat
- Hasil Produk
- Kesimpulan dan Saran

Latar Belakang

Latar Belakang

Mendengarkan musik dapat membawa suasana nyaman dan meningkatkan konsentrasi dalam belajar dan mengerjakan tugas. Untuk memfasilitasi suasana belajar dengan musik yang lebih mudah dan efektif, *Smart Music Study Companion* hadir untuk memberikan solusi tersebut. Pengguna hanya perlu duduk dan memberikan tanda kepada alat untuk memulai musik ketika akan mulai belajar, dan musik akan otomatis mati ketika pengguna meninggalkan tempat.



Fitur dan Deskripsi Alat

Fitur



Automasi musik ketika pengguna berada di jarak tertentu dari alat



Playlist yang dapat berasal dari USB



Audio tersambung melalui speaker atau earphone/headphone

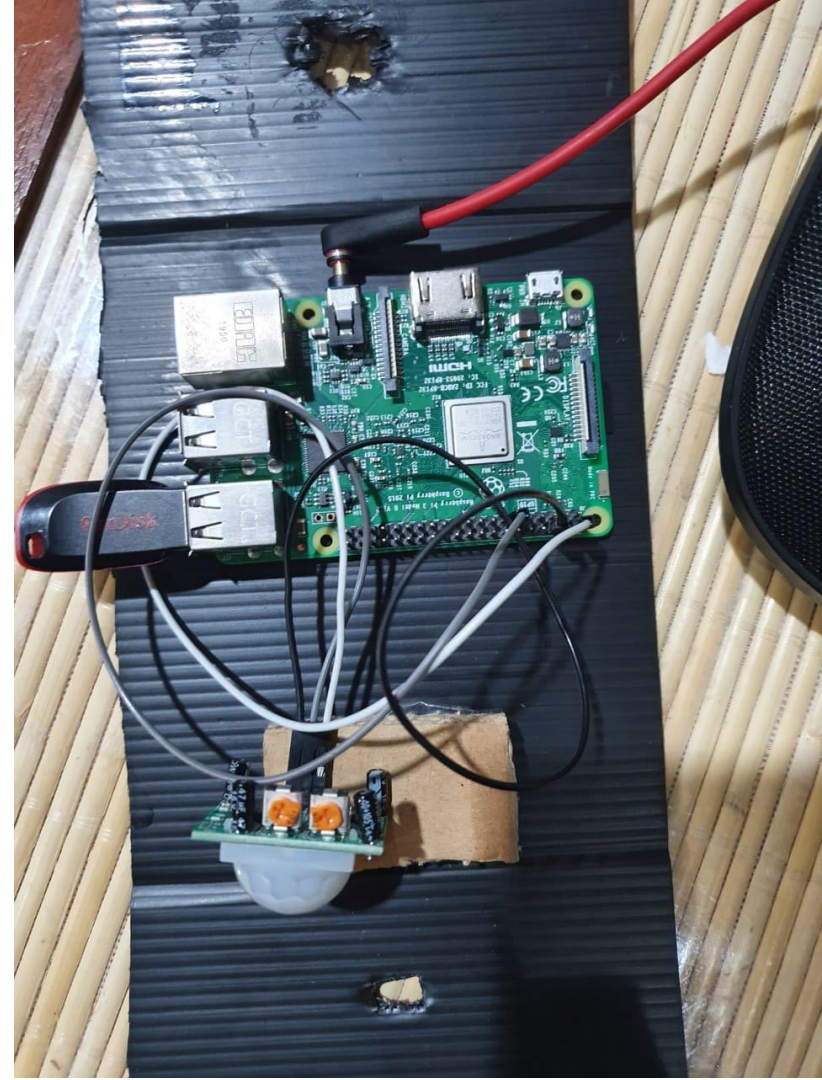


Deskripsi Alat

Terdiri dari:

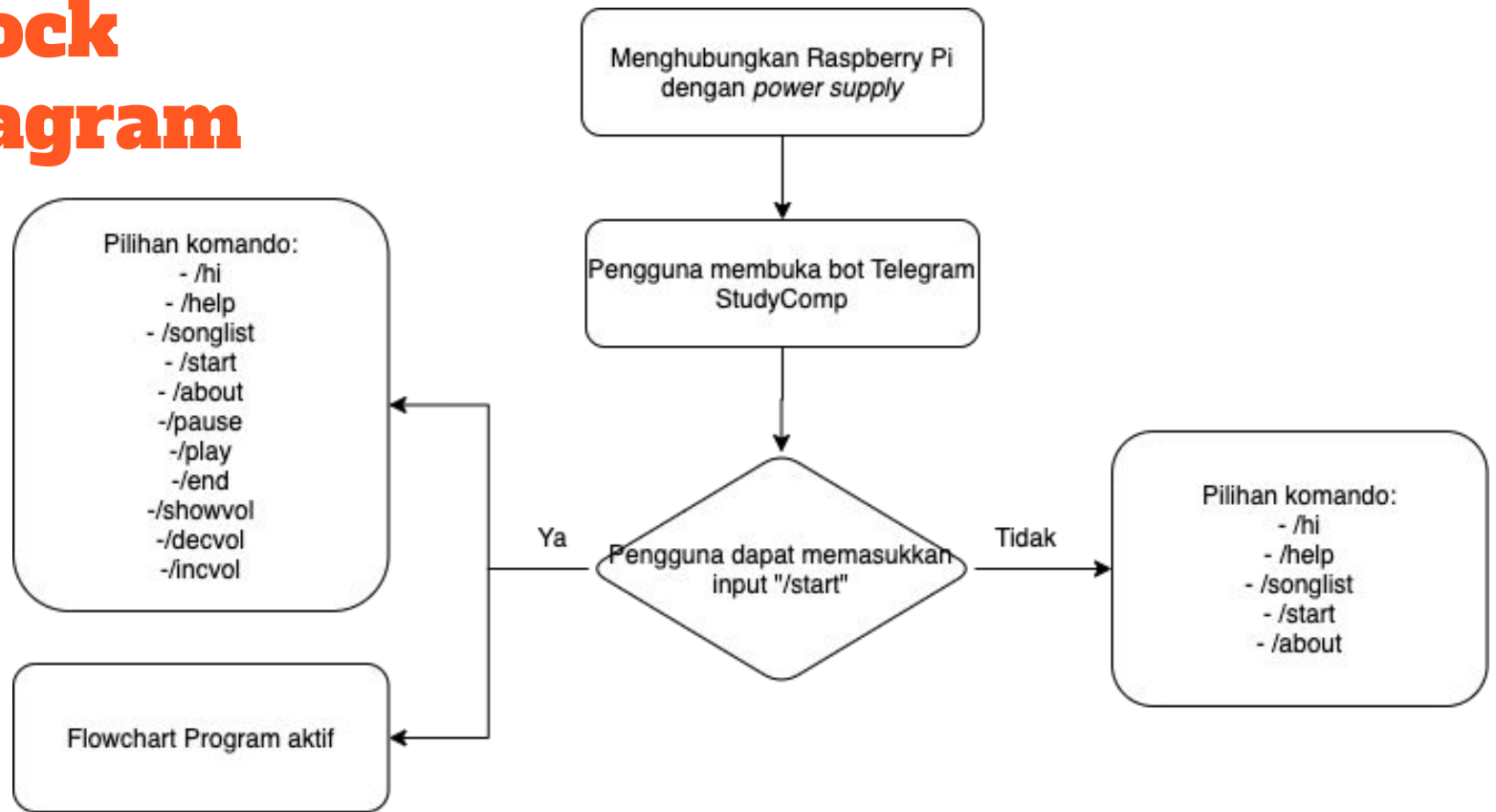
- Raspberry Pi 3 B
- PIR Sensor HC SR501
- Speaker/Earphone

Menggunakan program Python, module oxmplayer dan dapat dikontrol dengan bot telegram.



Hasil Produk

Block Diagram



Komando Bot Telegram

`/hi` : Pesan selamat datang

`/start` : Memulai program

`/about` : Menunjukkan informasi mengenai bot dan program

`/songlist` : Menunjukkan lagu-lagu yang akan diputar

`/help` : Menunjukkan daftar komando pada Bot Telegram

`/play` : Memainkan lagu ketika di-*pause* *

`/pause` : Men-*pause* lagu ketika lagu dimainkan *

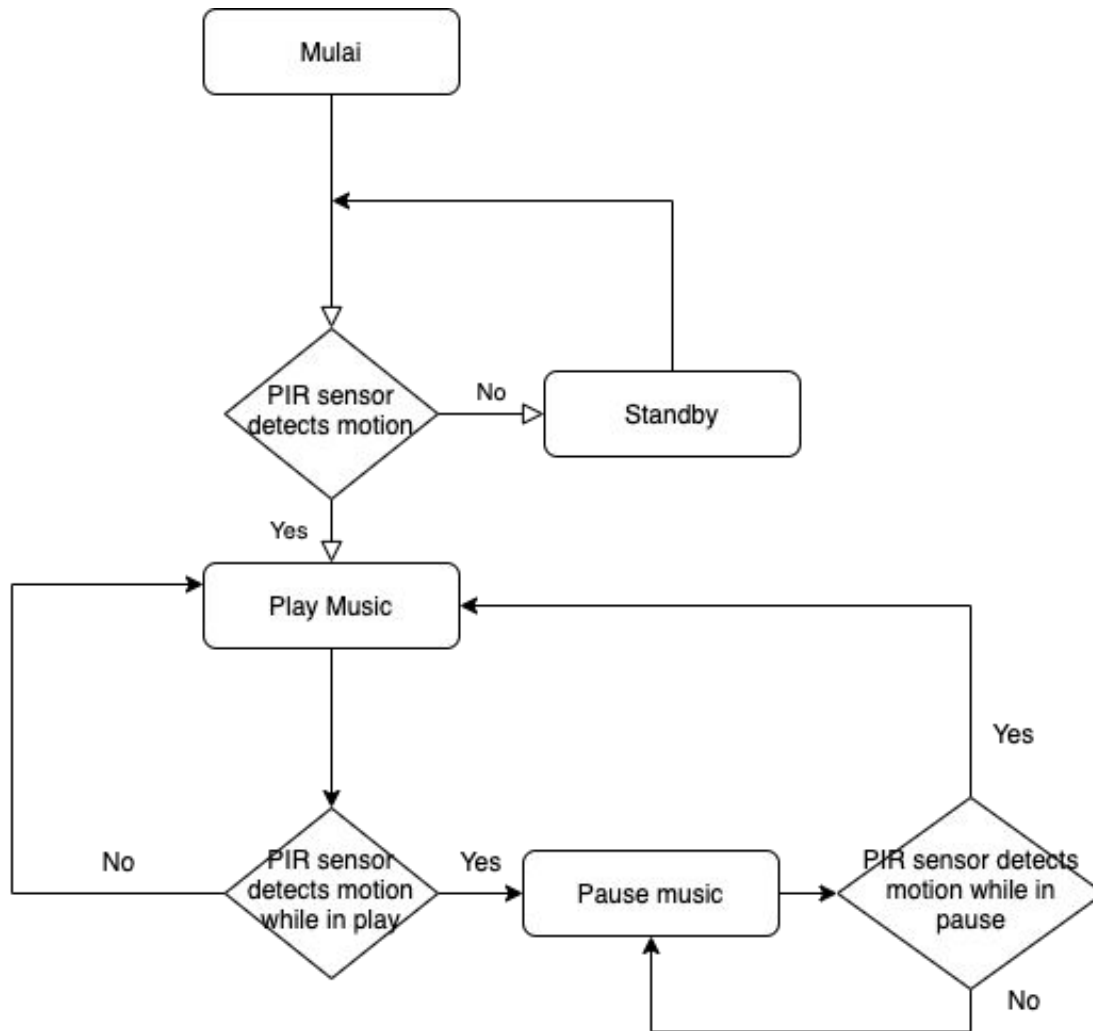
`/showvol` : Menunjukkan volume lagu sekarang (volume dalam jangkauan 0-10) *

`/decvol` : Mengurangi volume sebesar 1 *

`/incvol` : Meningkatkan volume sebesar 1 *

`/end` : Memberhentikan program

Flowchart Program



Program - Libraries

```
#Main library : time, GPIO, Thread, telepot, os, OMXPlayer, argparse
import time
from time import sleep
from threading import Thread
import RPi.GPIO as GPIO
import telepot
from telepot.loop import MessageLoop
import os
from os import listdir
from os.path import isfile, join
from omxplayer.player import OMXPlayer
from pathlib import Path
import argparse
import logging
logging.basicConfig(level=logging.INFO)
```

Program - Inisiasi Awal

```
#Early initialization
ap = argparse.ArgumentParser()
ap.add_argument("-d", "--directory", type=str,
                default="/home/pi/Downloads", help="Path folder containing the songs")
args = vars(ap.parse_args())

mypath = args["directory"]
musicfiles = [f for f in listdir(mypath) if (isfile(join(mypath, f)) and f.endswith(".mp3"))]
```

Program - Eksekusi Main Function

```
#Main program initialization (pin location, call AutoTrigger, stops program from KeyboardInterrupt)
def main(bot): ...

if __name__ == '__main__':
    #Connecting to bot Telegram
    bot = telepot.Bot('1407746688:AAG7gxt9cahWBz_fKP0NbsuthWNNB9I-1vw')
    print (bot.getMe()) #Bot Telegram connected

    #executing main function
    main(bot)

    while 1:
        sleep(10)
```


Program - AutoTrigger

```
#Detecting motion to play music while also receiving commands from Telegram Bot
class AutoTrigger():
    #Play the music from musicfiles list
    def call_omxplayer(self): ...

    ...

    If PIR sensor detects motion, there are 3 possibilities:
    - First time: Executing call_omxplayer function
    - If the song is in play : Pausing the song
    - If the song is in pause : Playing the song
    ...

    def play_song(self): ...

    ...

    Executing responses from the Telegram Bot
    Command /start to execute play_song and call_omxplayer as 2nd Thread
    ...

    def handle(self,msg): ...

    #initialize class variables, executing Thread to handle
    def __init__(self,pin, bot): ...
```

Tampilan Bot Telegram



Naufal Kurniawan

/start

9:41:10 PM



StudyComp

Starting the device

9:41:11 PM



Naufal Kurniawan

/pause

9:41:42 PM



StudyComp

Pausing the song

9:41:44 PM

05 – Caelestinum Finale Termini.mp3

9:41:44 PM



Naufal Kurniawan

/play

9:41:48 PM



StudyComp

Playing the song

9:41:49 PM

05 – Caelestinum Finale Termini.mp3

9:41:50 PM



Naufal Kurniawan

/end

9:41:56 PM



StudyComp

Ending the device

9:41:57 PM

Tampilan Bot Telegram



Naufal Kurniawan

/songlist

9:40:48 PM



StudyComp

List of songs to be played:

9:40:49 PM

05 – Caelestinum Finale Termini.mp3

9:40:49 PM

02 – Dawn Winery Theme.mp3

9:40:50 PM



Naufal Kurniawan

/about

9:40:52 PM



StudyComp

This bot is for LTKA Final Assignment.

Created by Thirza Nabila Syafriady and M. Naufal Kurniawan.

This bot will enable users to control Raspberry Pi by Pausing, Playing and Showing the list of songs.

Please do activate the Raspberry Pi before using this Bot.

9:40:53 PM

Kesimpulan

- Alat sudah bekerja sesuai tujuan (terhubung dengan bot telegram, dan mampu memainkan dan menghentikan musik melalui pir sensor)
- Terdapat delay reaksi hubungan PIR sensor dengan program
- Ada *bug* ketika mesin melakukan aksi start-end berkali-kali.
- Terdapat kemungkinan lagu overlap dengan lagu berikutnya karena fungsi sleep telah memasuki sleep kedua

Saran

- Melakukan multi layer sleep atau fungsi yang dapat menambahkan durasi sleep secara asynchronous
- Meringkas program agar beban daya yang dieksekusi kedua Thread tidak besar
- Melakukan *debugging* terhadap persoalan start-end berkali-kali

**Selesai dan
Terima Kasih**