SURIGAO DEL NORTE STATE UNIVERSITY

S.Y. 2023-2024

GUI DOCUMENTATION PROJECT IN ITE 121 - INTERMEDIATE PROGRAMMING

Submitted by: Alfred Cedric D. Libay, BSCS 1-A

Submitted to: Dr. Unife O. Cagas

Description of the Project

The Multifaceted Music Player or simply MMP is a provisional digital interface that allows you to play music files. Designed primarily for playing audio files randomly according to your replay. It typically includes several components to provide a comprehensive and interactive user experience, basic controls such as play, pause, and stop buttons, a volume adjuster, and a progress bar to track the song’s duration. Some players may also display the album art and provide a quick function to easily find your favorite tracks. MMP aims to combine functionality, aesthetics, and ease of use to provide an immersive and enjoyable music listening experience for users.

What is the Project all about?

The Multifaceted Music Player is all about providing a comprehensive and customizable music listening experience that goes beyond basic playback, catering to the diverse needs and preferences of music lovers.

MMP is a versatile tool that provides users with a convenient way to listen to their favorite music. It provides users with a simple and intuitive interface to browse, select, and play music tracks randomly based accordingly to their replay. Rather than just playing audio files, it aims to enhance the overall music listening experience by incorporating various tools, options, and integrations. Whether you're a casual listener or a die-hard music enthusiast, MMP is an essential tool for enjoying music on your terms.

Benefit

MMP offers a myriad of benefits. But the primary benefit of this Project is it allows users to listen to their favorite music whenever they want. Whether you're at home, on the go, or at work, MMP gives you the ability to enjoy your preferred songs, and playlists randomly with ease. It provides entertainment, relaxation, motivation, and mood enhancement, catering to your personal preferences and musical tastes. Ultimately, MMP brings joy and comfort through the power of music, enriching your daily life and creating memorable experiences.

Importance

The importance of this Project lies in its ability to facilitate access to and enjoyment of music, which serves various essential purposes in individuals' lives. This also serves as a conduit between individuals and the vast world of music. In essence, MMP plays a crucial role in enriching individuals' lives by providing access to music, entertainment, personal expression, productivity, and fostering emotional well-being. This serves as a relaxation tool, enhancing user experiences and fostering a deeper connection with the music we love.

Purpose

The purpose of this Project is to provide individuals with a platform to access, select, and enjoy their favorite music. MMP extends beyond basic music playback to provide users with a comprehensive and enriching music listening experience. In essence, the purpose of MMP is to serve as a versatile tool that enables individuals to access,

personalize, and enjoy their favorite music in a way that enhances their overall listening experience.

Features

Basic Playback Controls:

- Play, pause, stop, next track, previous track.
- Scrubbing or seeking through a track.
- Repeat and shuffle modes.

Visual Enhancements:

- Album artwork display.
- Dynamic visualizations synced with the music.

Customization Options:

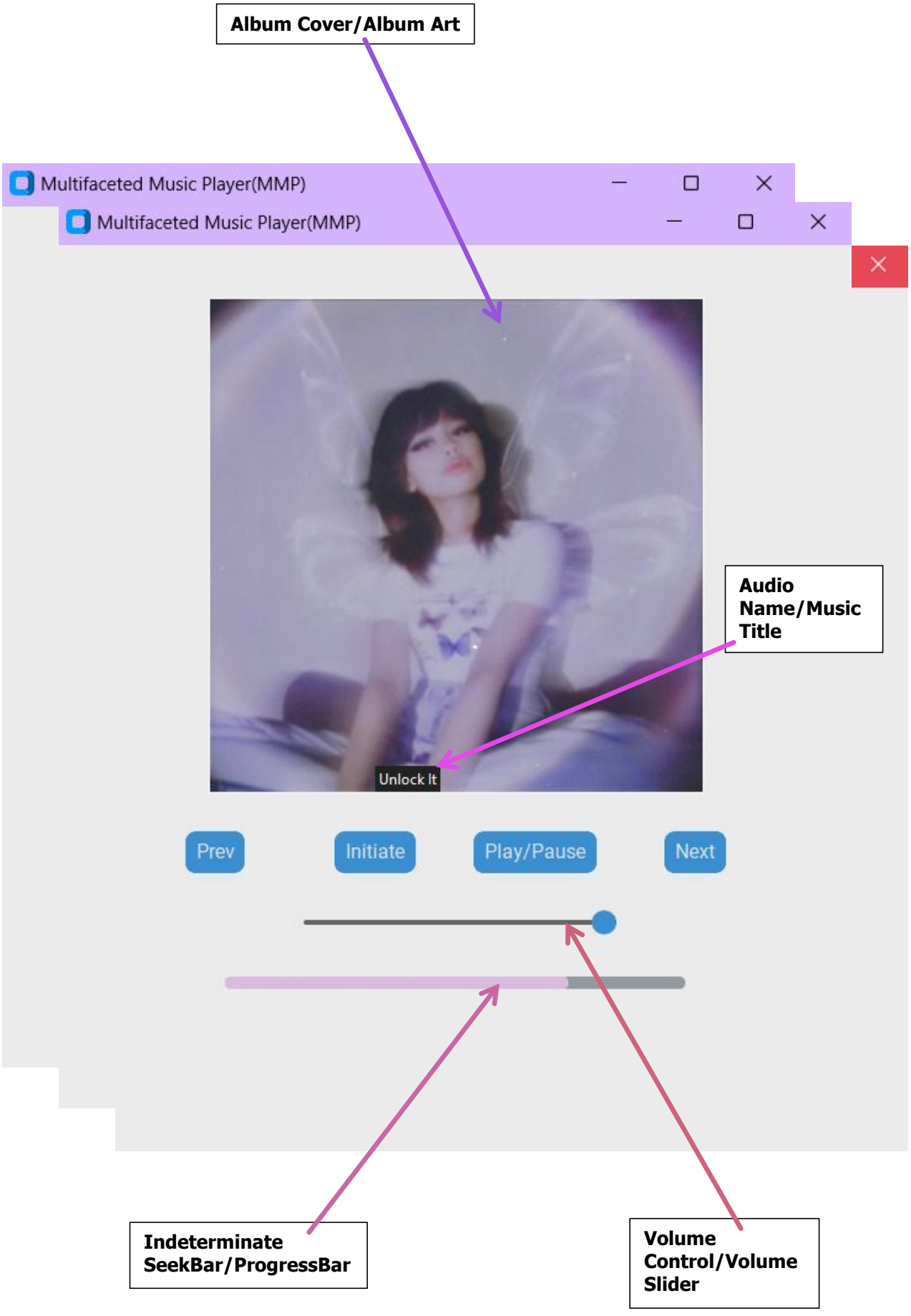
- Themes, skins, and color schemes.
- Customizable playback controls and gestures.
- Ability to change fonts, layouts, icon sizes and album grid dimensions.

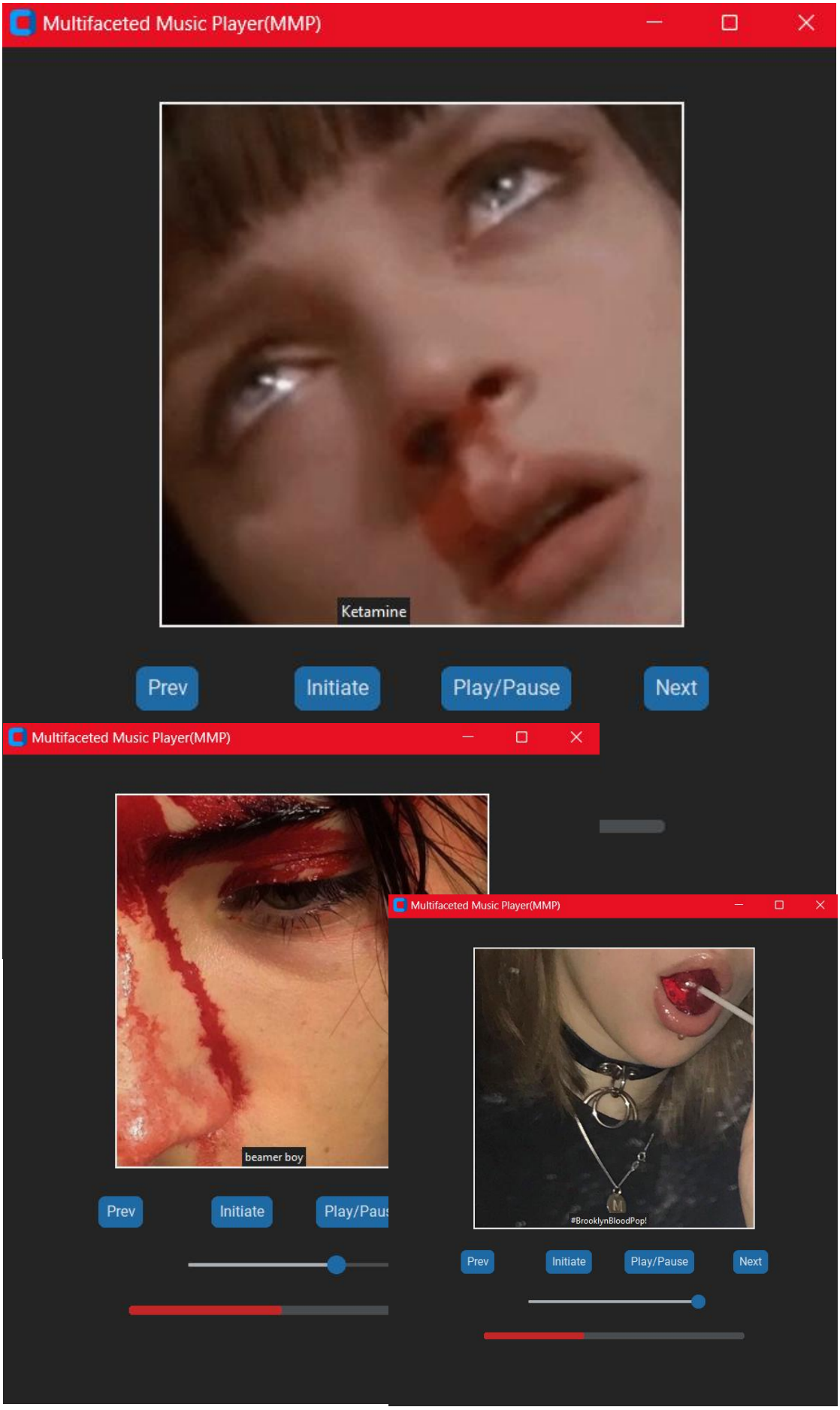
Remote Control and Casting:

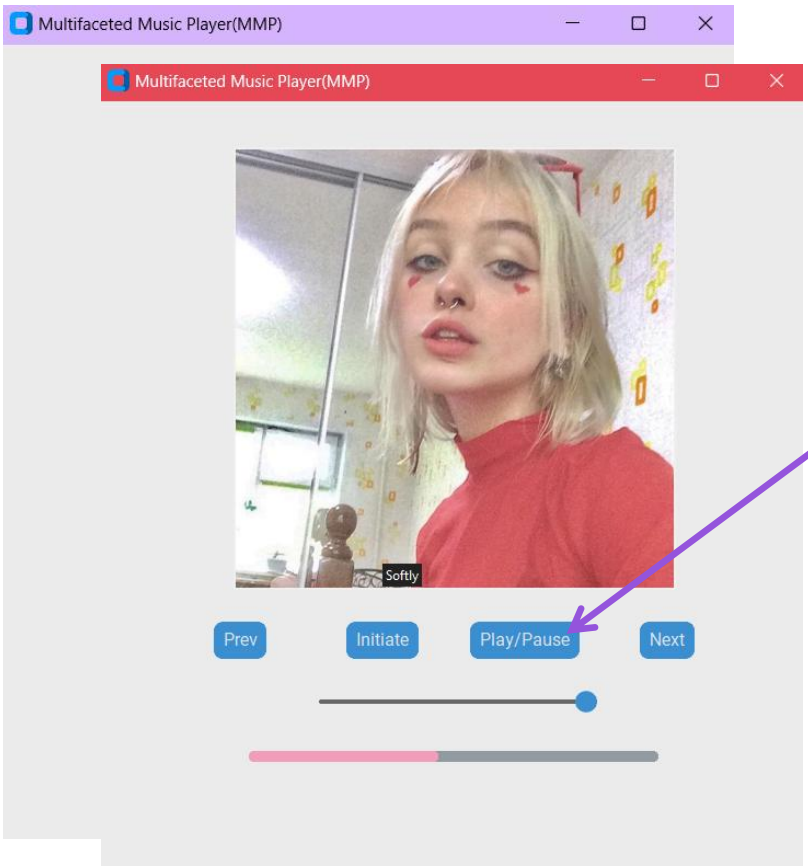
- Remote control functionality using mobile apps or web interfaces.
- Casting or streaming music to other devices such as smart speakers or TVs.

Accessibility Features:

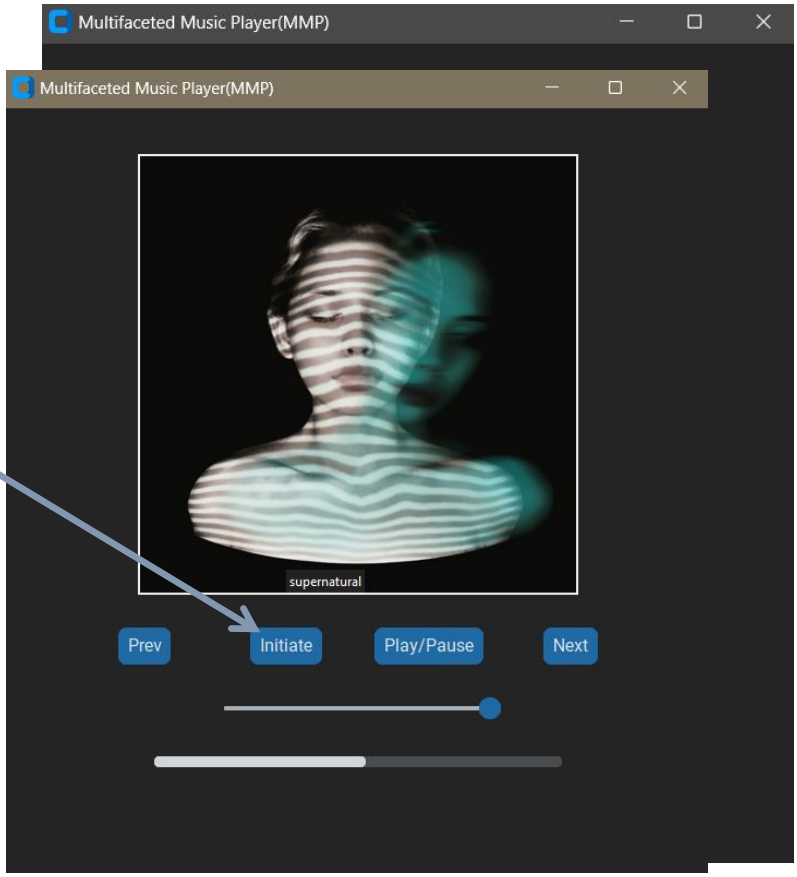
- Support for screen readers and accessibility settings.
- High contrast themes and other accessibility options.


Album Cover/Album Art
Multifaceted Music Player(MMP)
Multifaceted Music Player(MMP)
Unlock It
Audio Name/Music Title
Prev
Initiate
Play/Pause
Next
Indeterminate SeekBar/ProgressBar
Volume Control/Volume Slider

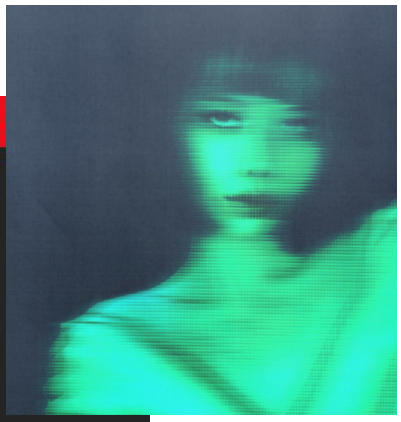

Multifaceted Music Player(MMP)
Ketamine
Prev
Initiate
Play/Pause
Next
Multifaceted Music Player(MMP)
beamer boy
Prev
Initiate
Play/Paus
Multifaceted Music Player(MMP)
#BrooklynBloodPop!
Prev
Initiate
Play/Pause
Next

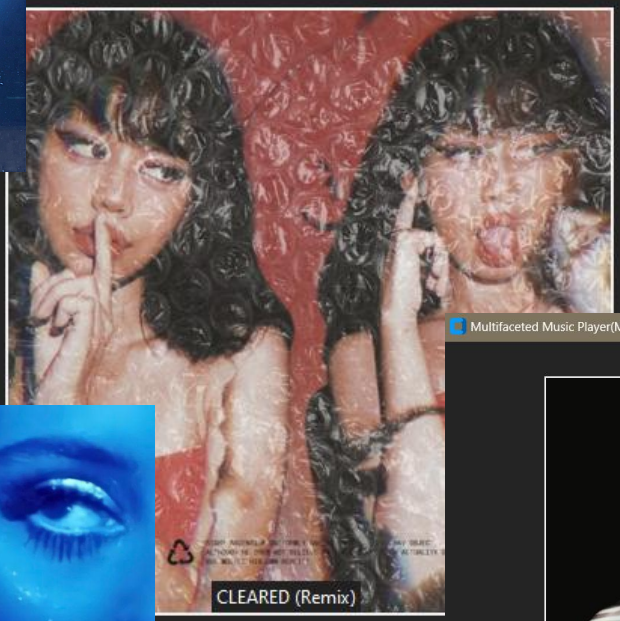


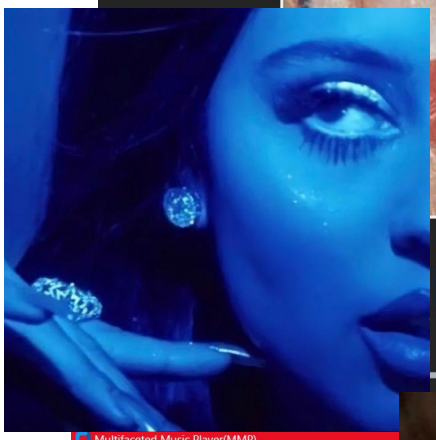
Play/Pause Button - Allows the user to start or stop audio playback



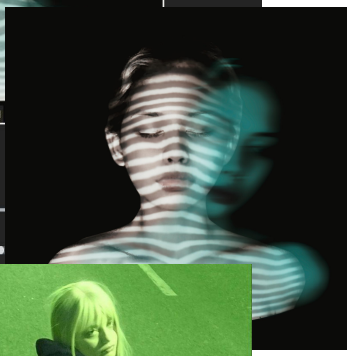
Initiator - Prompts the user to initiate audio playback directly



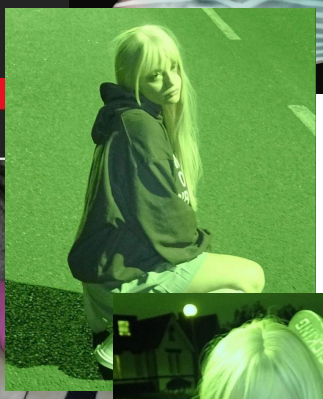

CLEARED (Remix)

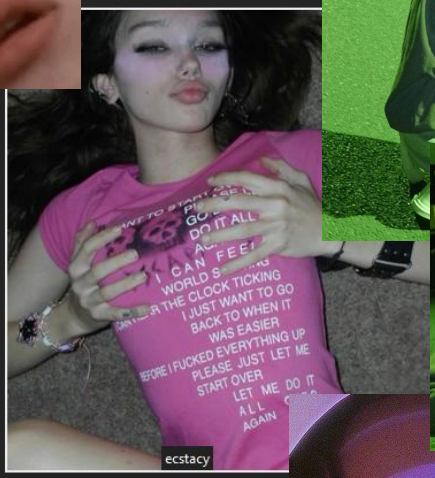


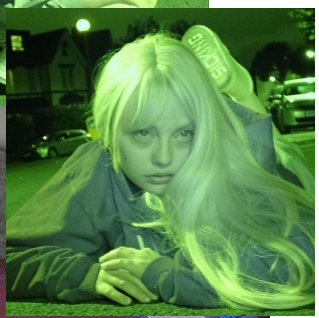

supernatural

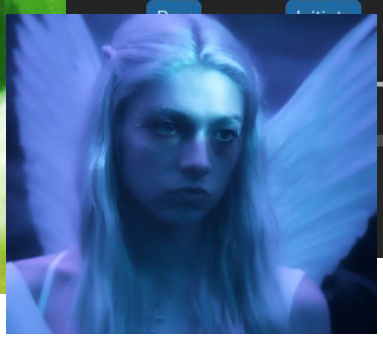


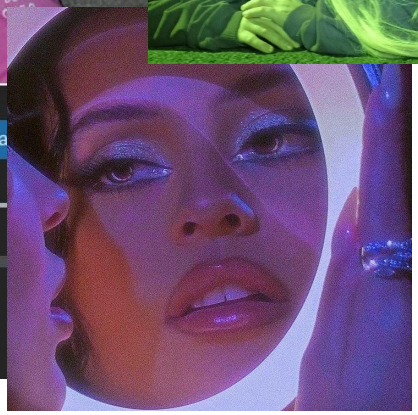

your lips
my lips
apocalypse
i like the way you kiss me

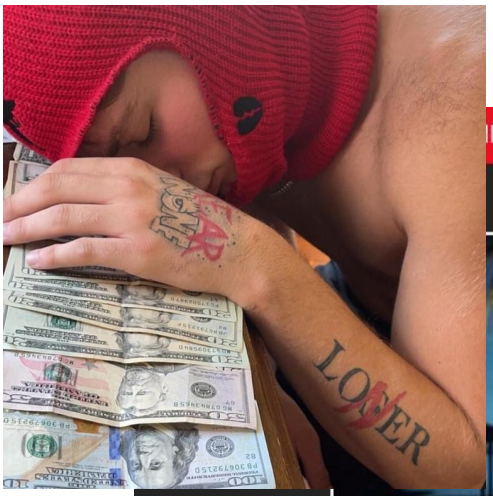


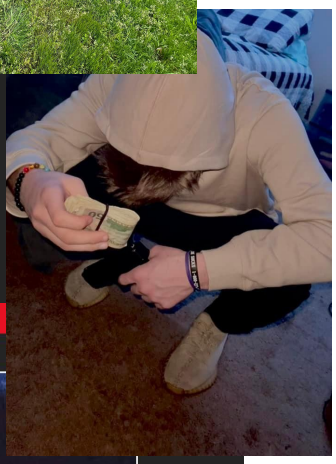

ecstacy

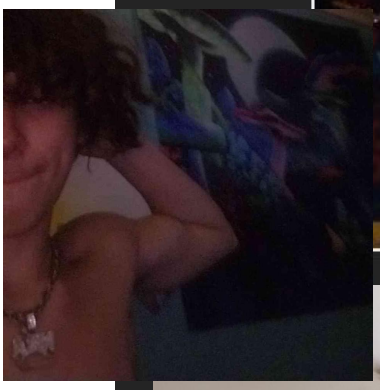






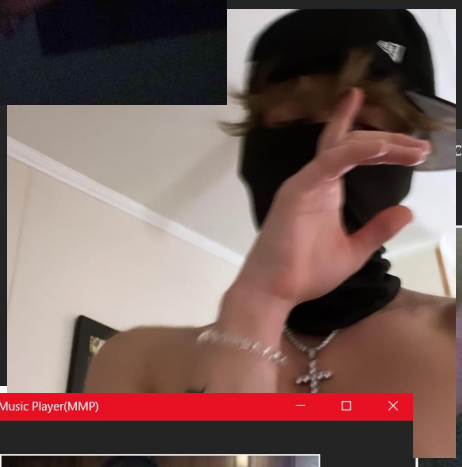

LOSER



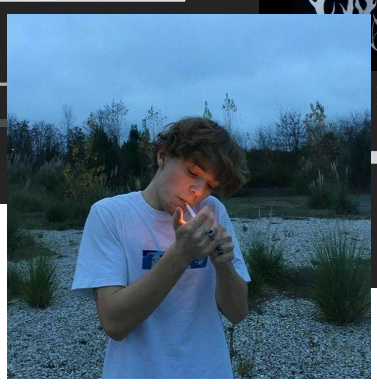


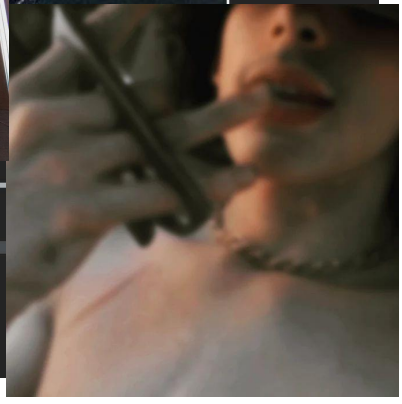

Freaks

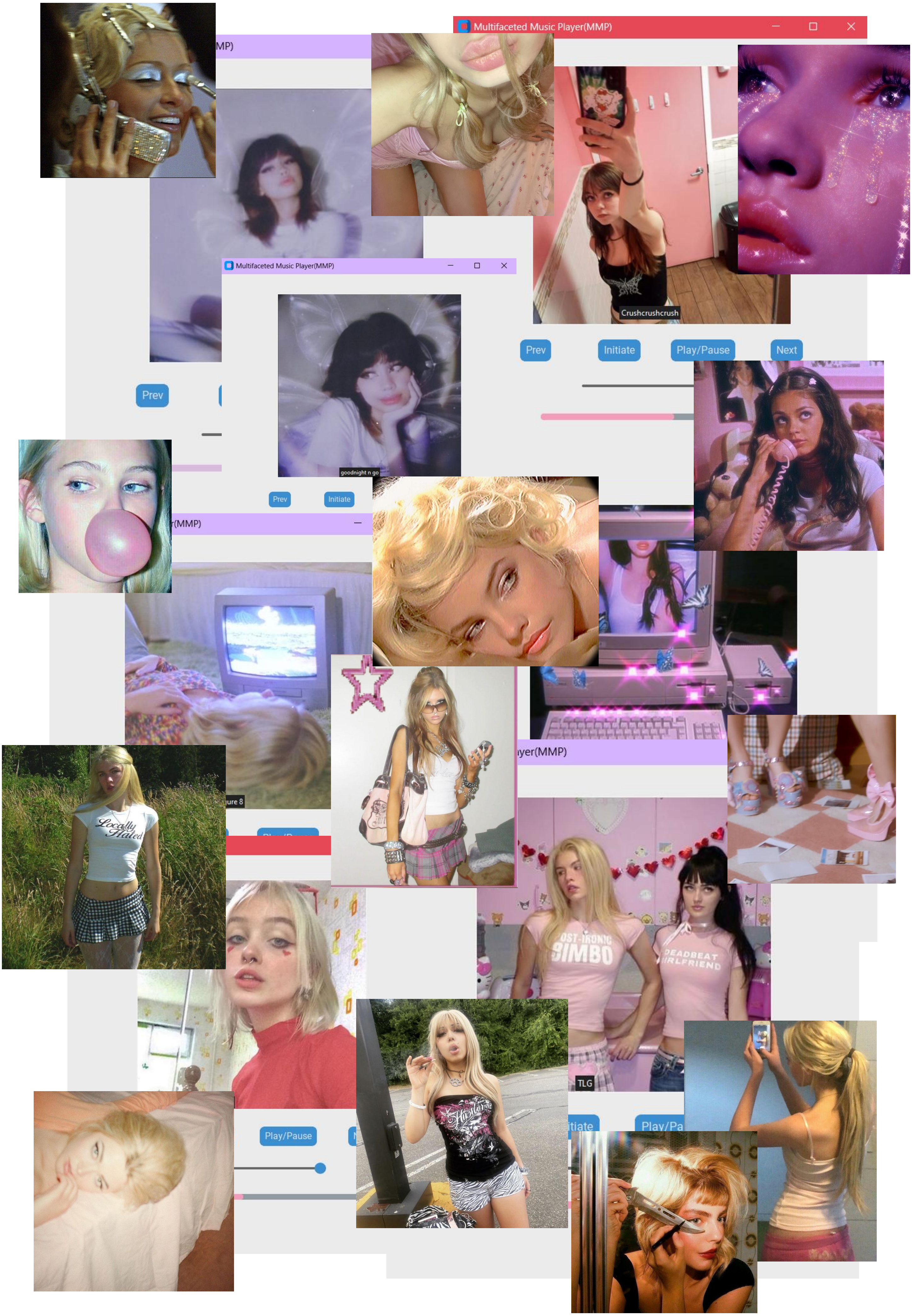

COKEWHORE

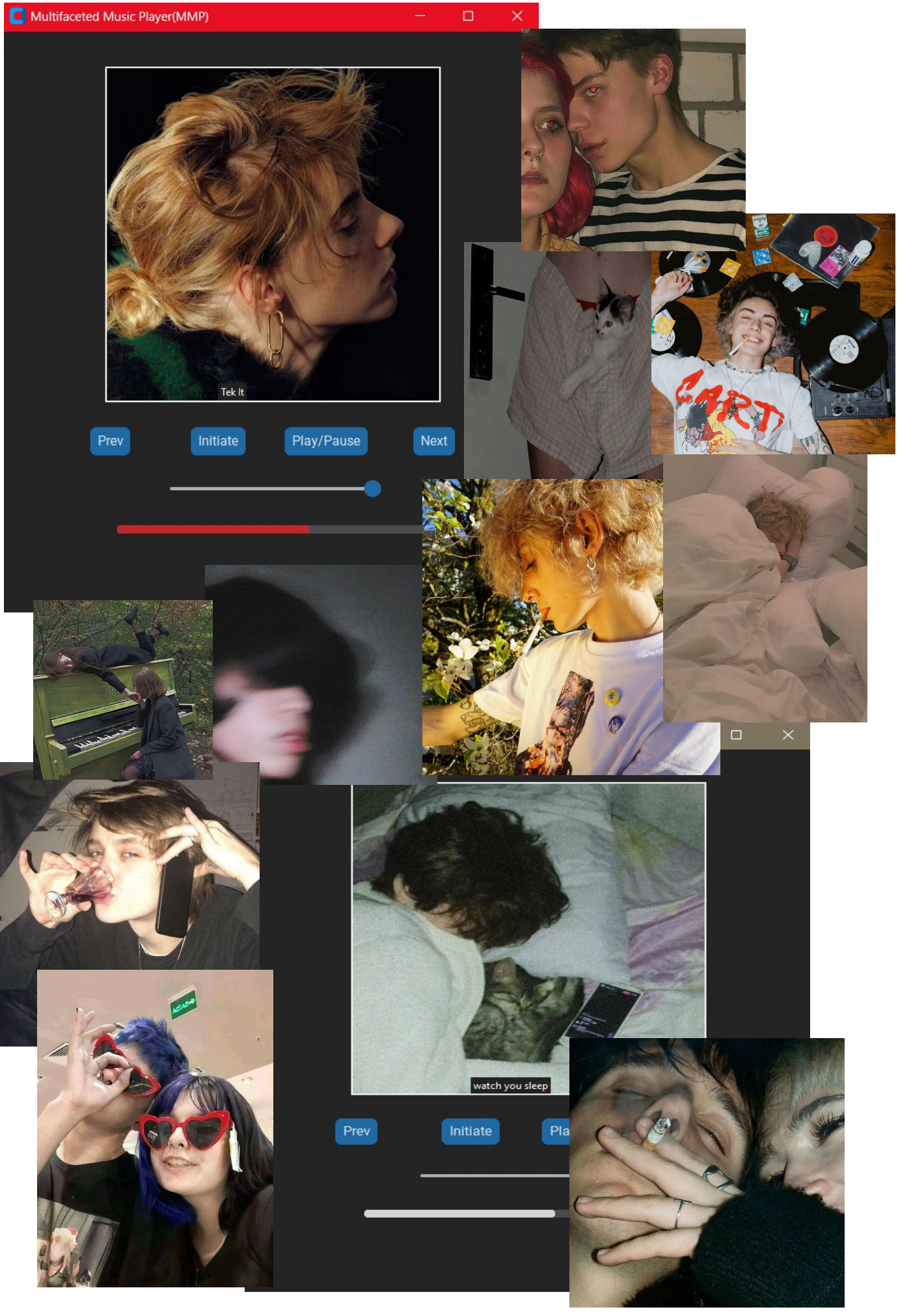



So Bitter






Multifaceted Music Player(MMP)
Crushcrushcrush
Prev
Initiate
Play/Pause
Next
Multifaceted Music Player(MMP)
goodnight n go
Prev
Initiate
Locally Hated
TLG
Play/Pause


Multifaceted Music Player(MMP)
Tek It
Prev
Initiate
Play/Pause
Next
watch you sleep
Prev
Initiate

Code

```
import tkinter
from tkinter.ttk import Progressbar
import customtkinter
import pygame
from PIL import Image, ImageTk
from threading import *
import time
import math

customtkinter.set_appearance_mode("Syste
m")  # Modes: system (default), light, dark
customtkinter.set_default_color_theme("blue
")  # Themes: blue (default), dark-blue,
green

##### Tkinter stuff ######
root = customtkinter.CTk()
root.title('Multifaceted Music Player(MMP)')
root.geometry('500x580')
pygame.mixer.init()
############################

list_of_songs = ['music/#Insert music
here.mp3'] # Add more MP3S into the music
directory.
list_of_covers = ['img/#Insert img here.jpeg']
# Add more JPEGS into the img directory.
n = 0

def get_album_cover(song_name, n):
```

```
    image1 = Image.open(list_of_covers[n])
    image2=image1.resize((400, 400))
    load = ImageTk.PhotoImage(image2)

    label1 = tkinter.Label(root, image=load)
    label1.image = load
    label1.place(relx=.19, rely=.06)

    stripped_string = song_name[6:-4] #This
is to exlude the other characters
                                        # 6       :
-4
                                # Example: 'music/
| TLG | .mp3'
                                # This works
because the music will always be between
those 2 values

    song_name_label = tkinter.Label(text =
stripped_string, bg='#222222', fg='white')
    song_name_label.place(relx=.4, rely=.6)

def progress():
    a =
pygame.mixer.Sound(f'{list_of_songs[n]}')
    song_len = a.get_length() * 3
    for i in range(0, math.ceil(song_len)):
        time.sleep(.4)

progressbar.set(pygame.mixer.music.get_po
s() / 299999)
```

```
def threading():
    t1 = Thread(target=progress)
    t1.start()

def play_music():
    threading()
    global n
    current_song = n
    if n > 2:
        n = 0
    song_name = list_of_songs[n]
    pygame.mixer.music.load(song_name)
    pygame.mixer.music.play(loops=0)
    pygame.mixer.music.set_volume(.5)
    get_album_cover(song_name, n)

    n += 1

    global paused
    paused = False

def pause_music():
    global paused
    if paused:
        pygame.mixer.music.unpause()
        paused = False
    else:
        pygame.mixer.music.pause()
        paused = True
```

```
def skip_forward():
    global n
    n == 2
    play_music()

def skip_back():
    global n
    n -= 2
    play_music()

def volume(value):
    #print(value) #if u care 2 see the volume
value in the terminal, un-comment tis btch!
    pygame.mixer.music.set_volume(value)

# All Buttons
play_button =
customtkinter.CTkButton(master=root,
text='Initiate', command=play_music,
width=0)
play_button.place(relx=0.4, rely=0.7,
anchor=tkinter.CENTER)

pause_button =
customtkinter.CTkButton(master=root,
text='Play/Pause', command=pause_music,
width=0)
pause_button.place(relx=0.6, rely=0.7,
anchor=tkinter.CENTER)
```

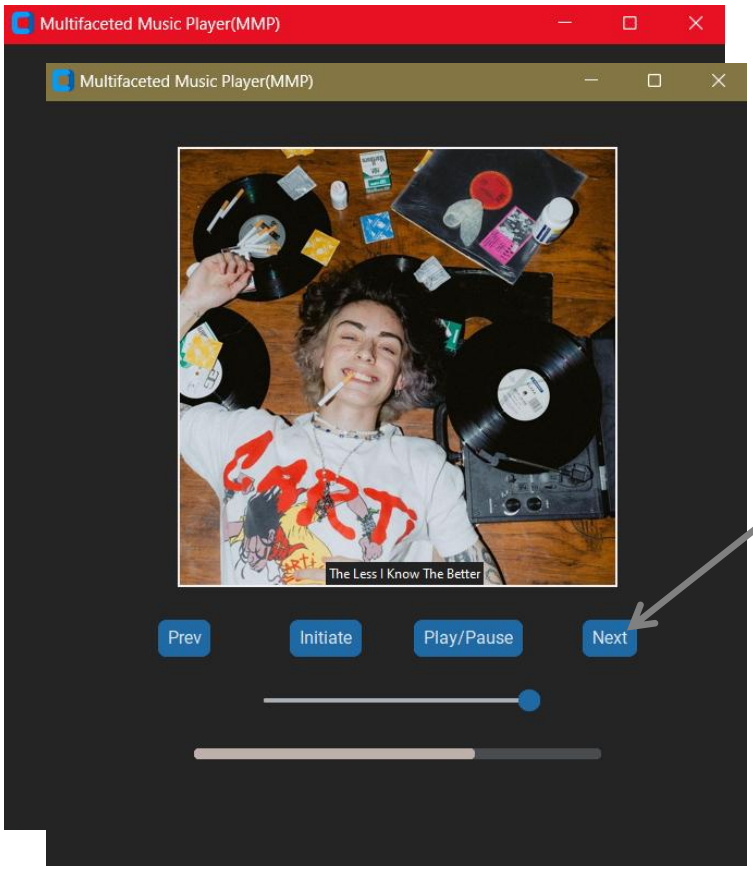
```
skip_f =
customtkinter.CTkButton(master=root,
text='Next', command=skip_forward,
width=0)
skip_f.place(relx=0.8, rely=0.7,
anchor=tkinter.CENTER)

skip_b =
customtkinter.CTkButton(master=root,
text='Prev', command=skip_back, width=0)
skip_b.place(relx=0.2, rely=0.7,
anchor=tkinter.CENTER)

slider =
customtkinter.CTkSlider(master=root,
from_= 0, to=1, command=volume,
width=210)
slider.place(relx=0.5, rely=0.78,
anchor=tkinter.CENTER)

progressbar =
customtkinter.CTkProgressBar(master=root,
progress_color='#F29EBA', width=300)
progressbar.place(relx=.5, rely=.85,
anchor=tkinter.CENTER)

root.mainloop()
```

Note: My code is still experiencing crashes and errors. There are few errors in some part of it, so it's not quite perfect yet. Please do consider thank you.

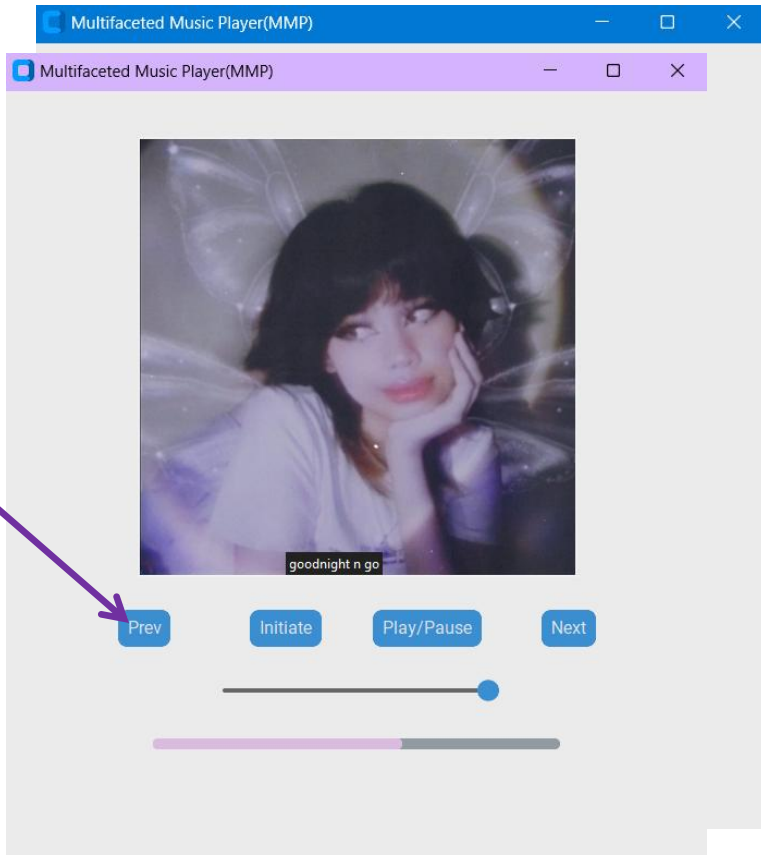
References

PIL:
https://pillow.readthedocs.io/en/stab...
CustomTkinter:
https://github.com/TomSchimansky/Cust...
PyGame (mixer):
https://www.pygame.org/wiki/GettingSt...
Threading:
https://docs.python.org/3/library/thr...
Source Code:
https://github.com/Terranova-Python/musicplayer

Functionality and Display



Skip-Forward Button - Prompts the user to skip the current track to the next



Skip-Backward Button - Prompts the user to skip the current track to the previous