오픈소스sw 팀플 과제 보고서

**import** os

**import** sys

**import** queue

**import** platform

**from** PyQt5 **import** QtWidgets, QtGui, QtCore

**import** vlc

**class** MiniPlayer(QtWidgets**.**QMainWindow):

"""Stripped-down PyQt5-based media player class to sync with "master" video.

"""

**def** \_\_init\_\_(self, master**=None**):

QtWidgets**.**QMainWindow**.**\_\_init\_\_(self, master)

# Create a basic vlc instance

self**.**instance **=** vlc**.**Instance()

self**.**media **=** **None**

# Create an empty vlc media player

self**.**mediaplayer **=** self**.**instance**.**media\_player\_new()

self**.**init\_ui()

self**.**open\_file()

self**.**timer **=** QtCore**.**QTimer(self)

self**.**timer**.**setInterval(10)

self**.**timer**.**timeout**.**connect(self**.**play\_mode)

//timeout.connext함수가 QTimer가 돌아가면서 setinterval함수의 간격마다 마감되었는지 검사하다가 마감이 되었을 때 작동되는 함수이므로 무한재생 기능을 play\_mode함수로 정의하여 타이머가 다 돌아갔을 때 play\_mode 함수를 호출하도록 넣었다

self**.**timer**.**start()

**def** init\_ui(self):

"""Set up the user interface

"""

self**.**widget **=** QtWidgets**.**QWidget(self)

self**.**setWindowFlags(QtCore**.**Qt**.**WindowStaysOnTopHint) ##상단 레이어

//상단 레이어 고정을 위해 윈도우 창에 상단 플래그 상수를 넣었다.

self**.**setCentralWidget(self**.**widget)

# In this widget, the video will be drawn

**if** platform**.**system() **==** "Darwin": # for MacOS

self**.**videoframe **=** QtWidgets**.**QMacCocoaViewContainer(0)

**else**:

self**.**videoframe **=** QtWidgets**.**QFrame()

self**.**palette **=** self**.**videoframe**.**palette()

self**.**palette**.**setColor(QtGui**.**QPalette**.**Window, QtGui**.**QColor(0, 0, 0))

self**.**videoframe**.**setPalette(self**.**palette)

self**.**videoframe**.**setAutoFillBackground(**True**)

self**.**vboxlayout **=** QtWidgets**.**QVBoxLayout()

self**.**vboxlayout**.**addWidget(self**.**videoframe)

self**.**widget**.**setLayout(self**.**vboxlayout)

**def** open\_file(self):

"""Open a media file in a MediaPlayer

"""

filename **=** "한양대학교 ERICA 홍보영상.f248.webm”

self.media = self.instance.media\_new(filename)

self**.**mediaplayer**.**set\_media(self**.**media)

//filename 변수에 영상제목으로 설정하여 상대경로를 통해 영상을 찾도록 하였다.

# Parse the metadata of the file

self**.**media**.**parse()

# Set the title of the track as the window title

self**.**setWindowTitle("{}"**.**format(self**.**media**.**get\_meta(0)))

# The media player has to be 'connected' to the QFrame (otherwise the

# video would be displayed in it's own window). This is platform

# specific, so we must give the ID of the QFrame (or similar object) to

# vlc. Different platforms have different functions for this

**if** platform**.**system() **==** "Linux": # for Linux using the X Server

self**.**mediaplayer**.**set\_xwindow(int(self**.**videoframe**.**winId()))

**elif** platform**.**system() **==** "Windows": # for Windows

self**.**mediaplayer**.**set\_hwnd(int(self**.**videoframe**.**winId()))

**elif** platform**.**system() **==** "Darwin": # for MacOS

self**.**mediaplayer**.**set\_nsobject(int(self**.**videoframe**.**winId()))

# Start playing the video as soon as it loads

self**.**mediaplayer**.**play()

**def** play\_mode(self):

**if** self**.**mediaplayer**.**get\_state() **==** vlc**.**State**.**Ended:

self**.**mediaplayer**.**stop()

self**.**mediaplayer**.**set\_position(0)

self**.**mediaplayer**.**play()

//play\_mode함수를 통해 무한재생 기능을 제작했다. Mediaplayer의 상태가 끝났을 경우 미디어 플레이어의 재생을 멈추고 mediaplayer의 재생 위치를 0%만큼 진행된 상태로 되돌리고 다시 재생을 하도록 만들었다.

**def** main():

app **=** QtWidgets**.**QApplication(sys**.**argv)

player **=** MiniPlayer()

player**.**show()

player**.**resize(480, 480)

sys**.**exit(app**.**exec\_())

**if** \_\_name\_\_ **==** "\_\_main\_\_":

main()

수정방안

개인적인 생각으로는 Qtwidget으로 창을 띄울 때 테두리가 생기는게 마음에 들지 않았다. 아마도palete를 어떻게 건들여서 깔끔하게 테두리를 없엤으면 좋겠다고 싶었고 무한재생 기능에 대해 검색을 해보면서 아예 재생모드를 loop으로 설정하면 한줄로 끝나는 함수가 있던데 이번에는 실패해서 넣지 못하였다. 또 재귀로 무한재생을 구현하는 사례도 보았는데 그러면 메모리가 많이 사용되지 않을까 싶어서 프로젝트 목적인 사이니지의 특성상 영상 재생이 많이 반복될 것 같아서 사용하지 않았다.