

CSED232 Object-Oriented Programming (Spring 2020)

Programming Assignment #5

GUI

Due Date: 2020년 6월 28일 일요일 23:59

담당조교 : 강민석 (masonkang@postech.ac.kr)

1. Objective

본 과제물은 C++와 Qt Framework를 사용해 GUI Application (Digital Watch) 제작을 실습해보는 것을 목적으로 한다.

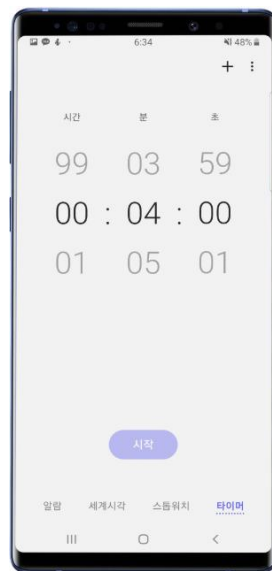


Figure 1. Digital Watch 예시

2. 과제 설명

본 과제의 최종 목표는 현재시간 표시, Timer 와 Stopwatch 기능(총 3가지 mode)을 포함한 Digital Watch를 구현하는 것으로, 과제 시작 전 구현할 프로그램에 대한 이해를 돕기 위해 아이폰/갤럭시 기본 “시계” 앱을 실행해 보는 것을 권장한다.

2.1 Watch

Digital Watch의 첫 번째 mode인 watch는 다음과 같이 구성되어 있다. 현재의 날짜와 요일이 표시되며 추가적으로 12시간 기준에 맞춰 시, 분, 초 AM/PM 이 구분되어 표시된다.

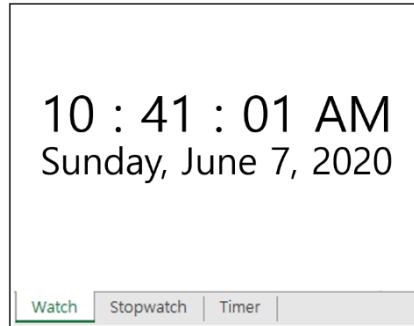


Figure 2. Watch 예시

2.2 Stopwatch

Digital Watch의 두 번째 mode인 Stopwatch는 다음과 같이 구성되어 있다. 기본적으로 측정되는 시간이 표시되는데, 분과 초 단위로만 구성 되어있으며 초 단위는 소수점 2자리까지 표시된다. 그리고 시간 측정을 위한 Start, Stop, Reset 버튼들로 구성되어 표시된다.

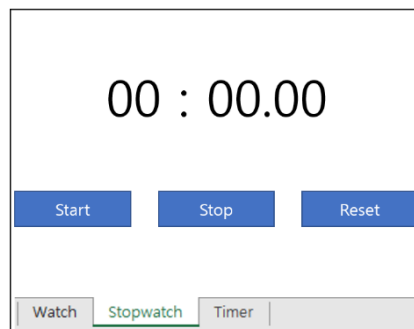


Figure 3. Stopwatch 예시

2.3 Timer

Digital Watch의 마지막 mode인 Timer는 다음과 같이 구성되어 있다. 기본적으로 Timer 시간(시, 분, 초)이 표시된다. 그리고 Timer 시간 설정을 위한 Up/Down , Start, Stop, Reset 버튼들로 구성되어 표시된다. 마지막으로 Timer 설정한 시간이 경과할 경우, 알람소리가 울리게 된다.

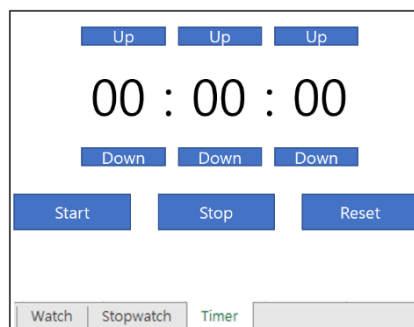


Figure 4. Timer 예시

3. PROBLEM : GUI PROGRAMMING (구현 요구사항) (90%)

Qt Framework과 C++ programming을 활용하여 아래와 같은 GUI를 구현한다. figure 5는 최종 GUI 예상도이며, Watch / Stopwatch / Timer 의 3가지 mode에 따라 각각 figure 6, figure 7, figure 8 와 같이 표시되도록 구현되어야 한다. 3개의 window를 구현하는 것이 아니라, 하나의 window 내에서 tab을 클릭함에 따라 3가지 mode가 표현되도록 구현하여야함을 주의.

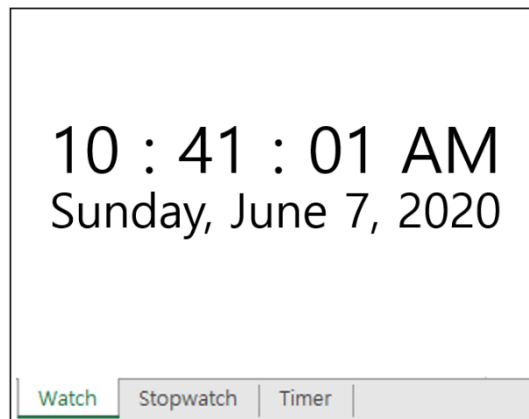


Figure 5. GUI 최종 예상도(기본 mode : Watch mode)

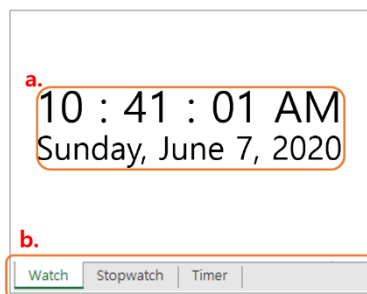


Figure 6. Watch



Figure 7. Stopwatch

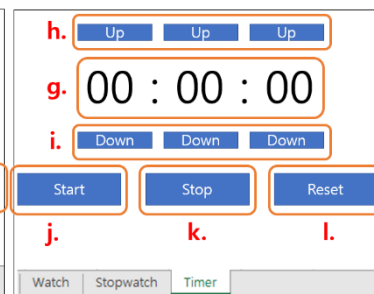


Figure 8. Timer

3.1 UI 요구사항(1) : Watch mode

- a. 현재 시간 표기 - 현재 날짜, 요일, 시간(12시간단위), AM/PM을 표시한다. 프로그램 실행당시의 시간을 한번 띄우고 숫자가 고정되는 것이 아닌, 1초마다 변하는 시간의 변화가 표현되어야 한다.
- b. mode 전환 탭 - 현재 활성화된 mode에서 다른 mode로 전환한다. Watch / Stopwatch / Timer 버튼을 클릭하면 각 mode로 전환되어 활성화되어야한다. 프로그램 실행시작 시

에는 Watch mode가 기본 mode로 활성화된다.

3.2 UI 요구사항(2) : Stopwatch mode

c. Stopwatch 시간 표기 - Stopwatch에서 시간의 경과를 분과 초 단위로 표시한다. 초 단위는 소수점 아래 두 번째 자리까지 표시되어야한다. 00 : 00.00를 기본 상태로 시작한다. 최댓값은 59 : 59.99이다.

d. Start 버튼 - stopwatch 시작 버튼으로, 누름과 동시에 c에 표시된 시간이 경과하는 시간만큼 상승하도록 한다. c에 표시되는 시간이 59 : 59.99를 초과하게 되면 그 즉시 Reset 버튼을 누른 효과와 동일한 효과를 내도록 한다.

e. Stop 버튼 - stopwatch 멈춤 버튼으로, 누름과 동시에 c에 표시되는 시간의 변화를 멈춘다. 0초에서는 Stop 버튼을 누르게 되더라도 아무 변화가 일어나지 않는다.

f. Reset 버튼 - 누름과 동시에 c에 표시된 시간을 00 : 00.00 으로 되돌린 후 멈춘다.

3.3 UI 요구사항(3) : Timer mode

g. Timer 시간 표기 - Timer에서 설정한 시간을 표시한다. 그리고 Start 버튼을 눌러 Timer가 실행되는 경우, 설정한 시간에서 Start를 누른 뒤 경과한 시간만큼 감소된 시간을 표시한다. 시/분/초 단위로 표시되며 00 : 00 : 00를 기본 상태로 시작한다. 최솟값은 00 : 00 : 00이며 최댓값은 23 : 59 : 59이다.

h. Up 버튼 - 각각의 up 버튼을 누를 때마다 g에 표시되는 시/분/초 단위가 따로 1만큼 상승하게 되어, 원하는 Timer 시간을 설정할 수 있게끔한다. 시/분/초 단위가 각각 23/59/59 일 때 각 단위에 해당하는 Up 버튼을 누르면 각각 0이 되도록 한다.

i. Down 버튼 - 각각의 down 버튼을 누를 때 마다 g에 표시되는 시/분/초 단위가 따로 1만큼 감소하게 되어, 원하는 Timer 시간을 설정할 수 있게끔한다. 시/분/초 단위가 각각 0/0/0 일 때 각 단위에 해당하는 Down 버튼을 누르면 각각 23/59/59가 되도록 한다.

j. Start 버튼 - Timer 시작 버튼으로, 누름과 동시에 g에 표시된 시간이 경과하는 시간만큼 감소하도록 한다. 0초에서는 Start 버튼을 누르게 되더라도 아무 변화가 일어나지 않는다.

k. Stop 버튼 - Timer 멈춤 버튼으로, 누름과 동시에 g에 표시되는 시간의 변화를 멈춘다. 0초에서는 Stop 버튼을 누르게 되더라도 아무 변화가 일어나지 않는다.

l. Reset 버튼 - 누름과 동시에 g에 표시된 시간을 00 : 00 : 00 으로 되돌린 후 멈춘다.

m. Alarm - Timer 시간 설정 후에 Start 버튼을 눌러 Timer 시간이 감소하게 되어 정상적으로 00 : 00 : 00 에 다다를 경우(Reset 버튼을 누른 경우 제외), 과제공지에 첨부한 alarm.wav 파일이 재생되어 알람이 울리도록 구현한다.

4. 참조사항

4.1 MinGW 위에서 Qt IDE 환경 (QtCreator) 설정하기

MSYS2 MINGW64 터미널에서, 먼저 모든 설치된 패키지를 최신 버전으로 업데이트한다.

- pacman -Syu

MSYS2 MINGW64 터미널에서 다음 패키지를 설치한다.

- pacman -S mingw-w64-x86_64-toolchain

- C++ 컴파일러와 디버깅 툴

- pacman -S mingw-w64-x86_64-qt5

- Qt5 프레임워크

- pacman -S mingw-w64-x86_64-qt-creator

- ◆ QtCreator IDE

설치 완료 후 터미널에서 "qtcreator &" 명령어를 실행하여 IDE 를 실행한다

4.2 구현 참조사항

오디오 파일과 같은 리소스들을 Qt 프레임워크에서 관리하는 방식에 대해서 다음의 공식 문서를 참조하면 배울 수 있다. (<https://doc.qt.io/qt-5/resources.html>)

오디오 파일을 실제로 재생하는 것은 Qt 프레임워크의 Audio Overview 문서를 참조하면 배울 수 있다. (<https://doc.qt.io/qt-5/audiooverview.html>). QSoundEffect 클래스는 .wav 파일을 재생하는 직관적인 인터페이스를 제공한다. (QSoundEffect::play(), QSoundEffect::stop() 슬롯 참조.)

Audio 등 Qt 추가 기능을 사용할 때, "QT += multimedia" 처럼 QT 변수에 추가 기능의 모듈 이름을 추가해줘야 함에 유의한다. Qt 레퍼런스 문서에 명시되어있으니 참조 바람. (예시:

<https://doc.qt.io/qt-5/qsoundeffect.html>)

위 사항들 뿐만 아니라, 구현에 필요한 내용들은 아래 documentation들을 참조하기 바람.

<https://doc.qt.io/qt-5/qtwidgets-module.html>

<https://doc.qt.io/qt-5/qtcore-module.html>

4.3 코드 작성 관련 주의사항

- 새로운 .h / .cpp 파일을 만들면 반드시 해당하는 프로젝트의 .pro 파일에 소스 코드와 헤더를 명시해줘야 한다. 이 과정을 지나칠 시 해당 소스 코드가 빌드 되지 않는다.
- 만약 소스 코드를 제대로 작성한 것 같은데도 컴파일/런타임 오류가 발생하면 프로젝트를 Clean 후 Rebuild All 하는 것이 도움이 될 수 있다.

5. 채점기준

- a - (6%)
- b, c, d, e, f, g, h, i, j, k, l, m - (7%)
- 위의 UI 요구사항에 명시되지 않은 기능은 구현하지 않아도 된다.
- UI 의 스타일은 점수에 반영하지 않는다.
 - 예) 버튼 색깔, 디자인, 글꼴, 글자 크기 등
- 보고서 - 프로그램 구현 완료 후 각 기능 별로 본인의 알고리즘과 UI들을 간략히 설명 (10%)
- 채점 환경은 MinGW 상 QtCreator 사용할 것임.
- 제출기한을 넘길 시 조교에게 직접 메일(masonkang@postech.ac.kr)로 제출. 제출지연에 따른 패널티 있음.

5. 과제 제출

컴파일이 되는 것을 확인한 프로젝트 폴더를 Clean 후 makefile과 함께 그대로 압축해서 보고서와 함께 제출할 것.

학번_project.zip, 학번_report.pdf 형태로 제출할 것.