GUI Summer 2020

Assignment – Interface1 (I1)

Due on June 9

Submission Instructions:

1. Please submit your work directly in TRACS (using the TRACS editor) or as a text/MS-word/PDF attachment by the due date/time.
2. Please write your name in the assignment header and as a part of the file name of the attachment.
3. It must be your own work – a penalty of at least one grade in your final grade and a report to the Dean of Students will result from sharing work or using other people work.
4. Please submit your assignment by the deadline – late submission will not be accepted and will result in a grade of 0.
5. A grade of 1 denotes an issue with your assignments, which you have to resolve with the instructor.
6. Please do not submit your assignment via email. If you have a justified documented reason for being late then please submit the assignment to your TRACS drop-box and notify me by email.
7. Please use only zip for compression.
8. Please submit only the source code of your program in C (C++) + OpenGL / QT library functions. In addition, please submit an image (in JPEG) that shows the scene obtained through running your program. If I have doubts or concerns about your program, I may request that you submit the entire files required to produce a working program under Linux so that I can test your submission.
9. The code should include remarks that explain any non-trivial part of the code. For example, if you use a spin-box then you do not have to explain what it is. On the other hand, if you connect a spin-box to an LCD, you should have comments to explain what is being done.
10. At this point, I do not care too much about usability and detailed design. The goal is just to exercise using these widgets.
11. Instructions on how to produce an image of your scene are given below.

Assignment Instructions:

The goal of assignments I1 is to construct a part of the user interface to a "software radio." The interface enables control of parameters such as modulation, station, and volume, as well as bass settings, treble settings, stereo, etc.

This is a set of guidelines it does not mean that we will implement all of the above.

For assignment I1, I would like to ask you to implement the controls for modulation (AM, FM, etc), station, and volume, as well as the bass and treble settings. Although this is not a good UI practice, for learning purpose, each of these control and knobs should be a different widget from the QT widget libraries, e.g., dials, spin-boxes, sliders etc. Each input widget should be connected to an output widget (e.g., an LCD) that shows the status of the controlled setup. All the widgets should be in one window that enables "exit" functionality using a widget rather than Ctrl C.

To get ideas for your interface you can look at several players such as iTunes, Window player, Real etc. You can also look at http://bestplay-internet-radio-tuner.findmysoft.com/screenshot/. This should just give you impression; we are not going to implement the same but there is overlap.

GUI Summer 2020

Assignment – Interface1 (I1)

Due on June 9

If the widget is “input only”

- Select Radio functionalities
- For each functionality assign a unique widget (your choice) for enabling the functionality
  - e.g., a dial for station volume
- Find the documentation for a dial widget class
  - Methods,
  - Signals
  - Slots
- Write the code for this widget
- If the widget is “input only”
  - Add a second widget that shows the user selection (e.g., LCD – “output only”)
  - Connect the input widget to the output widget (at least one connection in the assignment)
- Fix the range
- Decide what layout to use (combination of H, V maybe grid).
- You must have an exit push button
- Open a window
- Place your widget in the window.

GUI Summer 2020
Assignment – Interface1 (I1)
Due on June 9

Producing an Image of your scene

The following are instructions for "screen dump". That is, how to capture the screen / current window into a file under Windows-XP and under Linux.

Print screen under Windows-XP>

(From: https://www.howtogeek.com/226280/how-to-take-screenshots-in-windows-10/)

To print the contents of the Screen, you must save an image of your screenshot to the clipboard and then paste it into a document where it can be printed.

To do this, follow this procedure:

1. Maximize the window you'd like to capture.
2. To copy/capture the current window, hold down **ALT + Print Scrn** at the same time. To capture the entire screen, just press **Print Scrn**.
3. Open a new document in MS Word, MS Photo Editor, or Adobe PhotoShop.
4. **Paste** the screen shot by holding down **CTRL** and **V** at the same time.
5. Print the document when you're finished pasting screen shots.

Print screen under Linux

(From: http://www.sb.fsu.edu/~xray/Manuals/ScreenCapture.html)

The same location includes more and allegedly better utilities for screen capture)

Using Linux's native utility 'xwd/xwud'

Most Linux and UNIX operating systems have native utilities called xwd (x-window dump) and xwud (x-window un-dump) that allow the user to capture either the whole screen or a specific window. To capture specific window, issue the following command,

'xwd >myimage.xwd'.

Then click with the mouse inside the window of choice (if the whole screen needs to be captured, simply click anywhere in the screen). The utility will write out an image in the special '.xwd' format. The created image can be viewed either using the related utility 'xwud' or using any number of image processing software like, display, gimp, or xv. To view using 'xwud' issue the following command,

'xwud -in myimage.xwd'

The image can be converted to .jpg or .png using the convert utility.

'convert myimage.xwd myimage.jpg'