

Technical University of Moldova
CIM Faculty

REPORT

Windows Programming
Laboratory Work #1

Done by:
Roman ROIBU

Verified by:
Alexandr TRUHIN

Chisinau 2013

Windows Programming Laboratory Work #1

Title

Window. Window handling. Basic window's form elements.

Introduction

Main purposes of this lab work: * Understand Event-Driven Programming. * Study Win32 API.

Secondary purposes of this lab work: * Use a Version Control Systems (GIT). * Write code according to a Programming Style Guide.

Mandatory Objectives

- Create a Windows application
- Choose Programming Style Guidelines that you'll follow
- Add 2 buttons to window: one with default styles, one with custom styles (size, background, text color, font family, font size)
- Add 2 text inputs to window: one with default styles, one with custom styles (size, background, text color, font family, font size)
- Add 2 text elements to window: one with default styles, one with custom styles (size, background, text color, font family, font size)

Objectives With Points

- Make elements to fit window on resize. **(1 pt)**
- Make elements to interact or change other elements. **(2 pt)**
- Change behavior of different window actions (at least 3). **(1 pt)**
- Write [your own PSG](#). **(1 pt)**

Programming Style Guide

For this laboratory work (and for future personal projects), I elaborated a [C Programming Style Guide](#), which was inspired by [Rhombus C style guide](#). I also added to this PSG a section specific for Windows Programming with Win32 API. I intend to continue using, and contributing to this PSG, because it insures code consistency, and writing much cleaner applications.

Application Creation Steps

In the development process, I consulted the recommended book ([“Programming Windows, 5th Edition”](#), by [Charles Petzold](#)). Also, I found myself consulting the documentation on the [MSDN](#) page.

I developed my program on Mac OS X using MinGW32 and Wine, as suggested by Mike Iachimovski in his [blog post](#). Also, I have Windows 8 installed on Parallels virtual machine, which I used to run the application for the final User Acceptance Test.

First of all, I made a wireframe model of the application general layout and functionality, using the [MockFlow](#) tool. Here is the mockup I came up with:

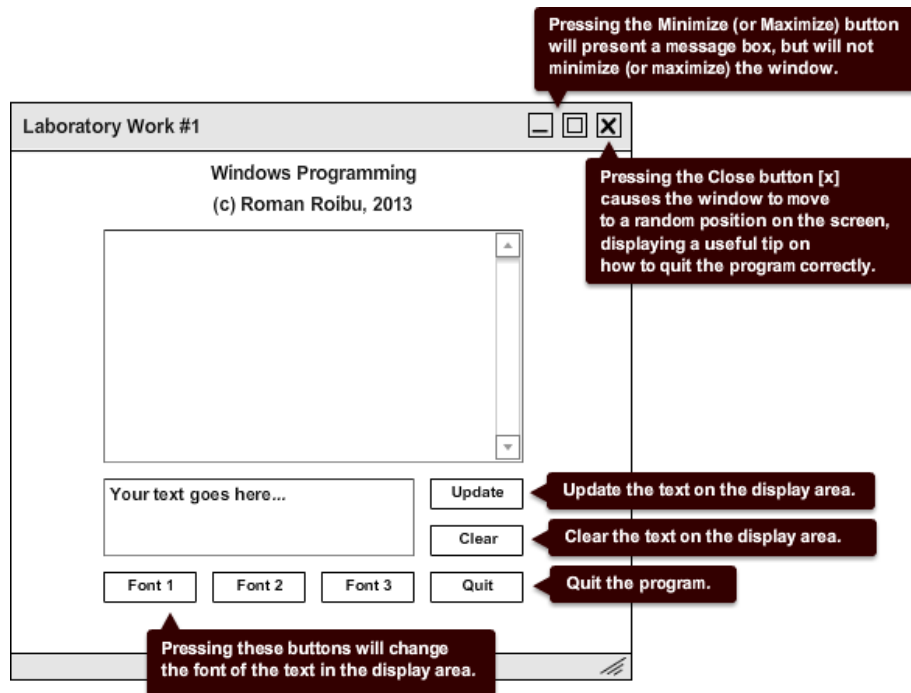


Figure 1: Mockup

The dialog bubbles describe each element and it's functionality.

Result Application

This is a screen shot of the application in the virtual Windows 8 environment:

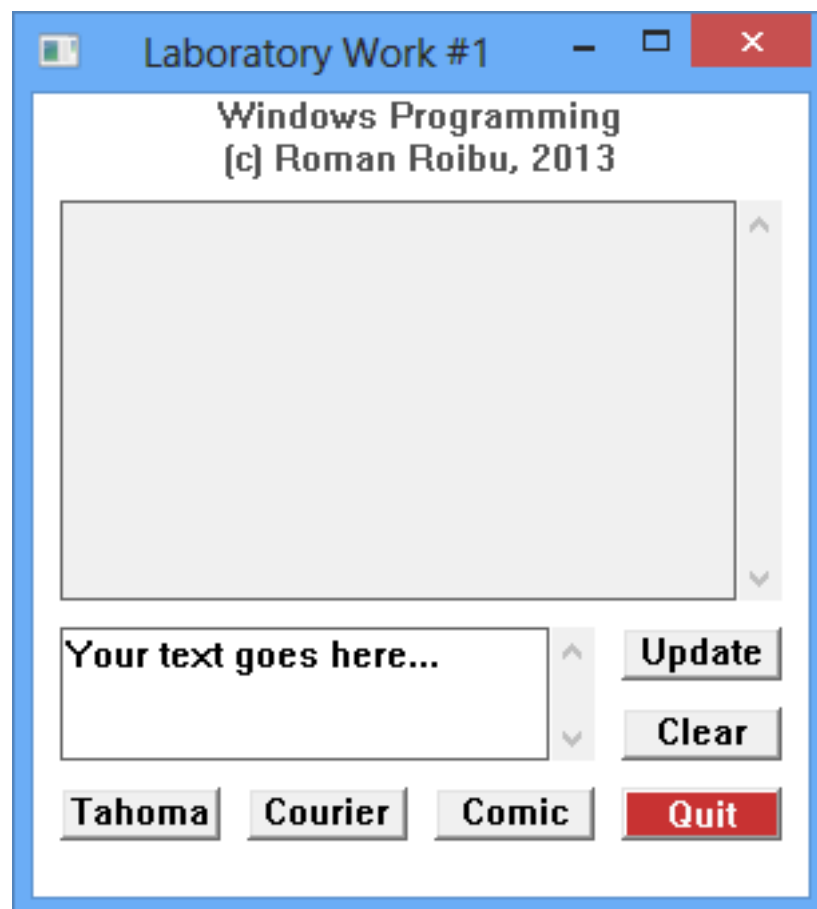


Figure 2: Screenshot

Conclusions

Although I don't intend to become a Windows developer, nor I use a Windows system in my development projects, I found this laboratory work very usefull, because I gained experience in Event-Driven Programming, which aided my understanding of this paradigm. I find the knowledge gained during this laboratory work to be very important, because these concepts can be applied not only in Windows Programming, but in developing any system which uses message-sending and Event-Driven Programming.

The only problem I encountered is that when I run the application with Wine, the redefined behaviour for Minimize and Maximize buttons doesn't work appropriately. However, the exactly same executable compiled with MinGW32 on OS X work perfectly fine on Windows 8 and Windows 7 (I didn't test the program in older versions of Windows), and all the behaviours are as expected. Consulting MSDN, and an extensive search on Google, didn't highlight the problem. My best guess is in the fact how Wine translates/sends the Minimize and Maximize messages, from Windows system command, into an OS X system command.

I think that this laboratory work, even if it didn't cover any advanced topics, was very important, because it laid a good foundation, on which to build upon. I look forward to next laboratory works.