# Assignment 6: ToDo Reminder

## 1. Objectives

The main objectives are:

- Working more with enum, constructors, properties and List<T>.
- Using some new common and useful controls, e.g. MenuStrip, ToolTip and Timer control, as well as the MessageBox.

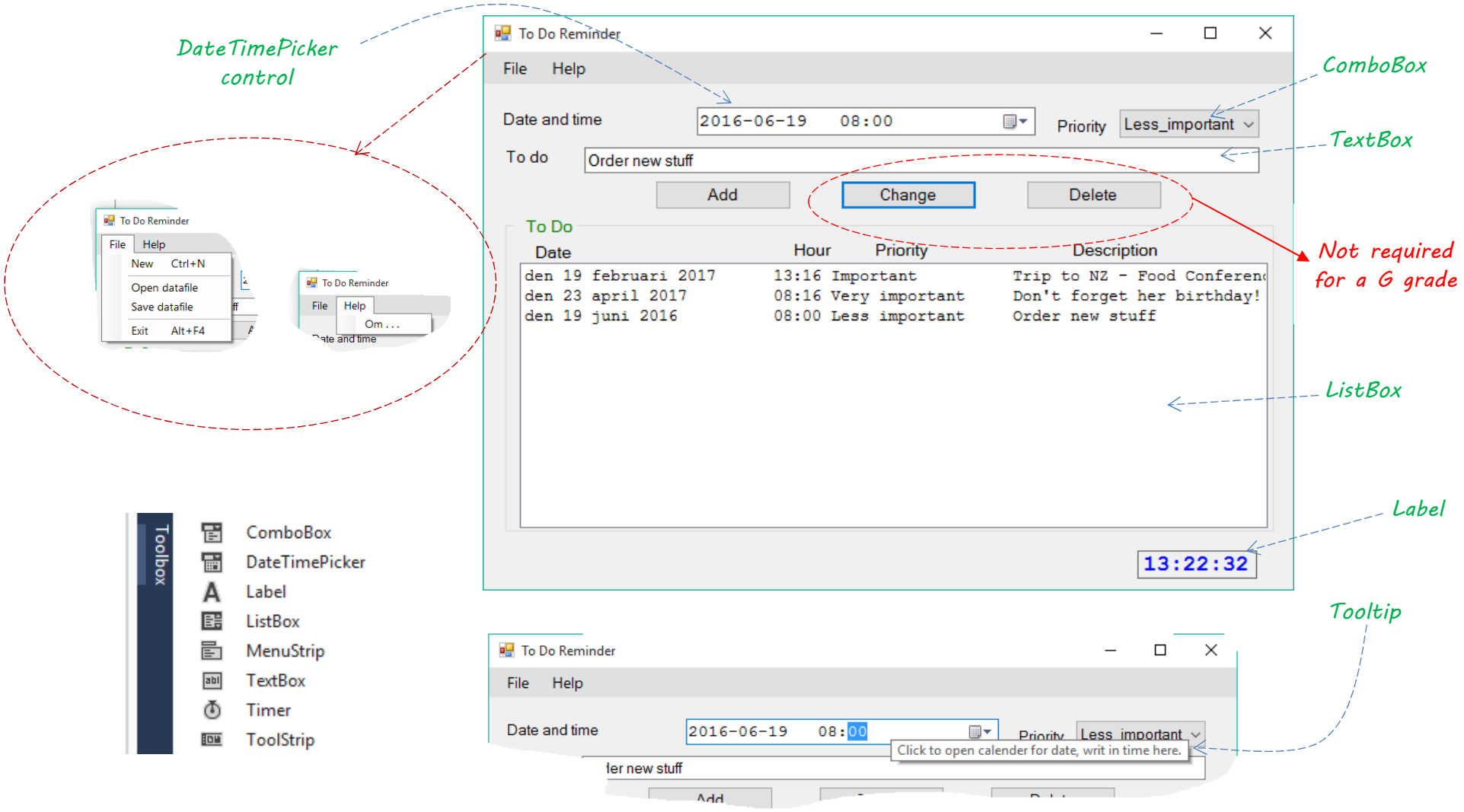
You are expected to solve this assignment using your experiences from the previous assignments. Your code must be well-structured, documented using proper comments and that you use properties, constructors and encapsulation in a correct way as you have learned throughout the course.

## 2. Description

Apu is having a hard time getting organized with all the tasks which he has to do, and remember to do them on time. He should not of course miss birthdays, shopping, getting a haircut and so on. Your assignment is now to write a C# program for him (and all others having the same problem) that keeps track of all to-dos.

The program should allow the user to specify a task, select a task date and time, and the program should display the data in a list (ListBox, or ListView) on the GUI. When working with a ListBox, you should check so an item is selected. This can be done by controlling so the value of the **index = lstTasks.SelectedIndex** is not -1

Create a class **Task** to store data and handle operation for a to-do (a task item) and a container class, **TaskManager**, containing a List<Task> object to manage a list of the tasks inputted by the user.


DateTimePicker control
To Do Reminder
File
Help
New Ctrl+N
Open datafile
Save datafile
Exit Alt+F4
Om . . .
Date and time
To Do Reminder
File Help
Om . . .
Date and time
To Do Reminder
File Help
Date and time
2016-06-19 08:00
Priority
Less_important
To do
Order new stuff
Add
Change
Delete
To Do
Date Hour Priority Description
den 19 februari 2017 13:16 Important Trip to NZ - Food Conferenc
den 23 april 2017 08:16 Very important Don't forget her birthday!
den 19 juni 2016 08:00 Less important Order new stuff
13:22:32
ComboBox
TextBox
Not required for a G grade
ListBox
Label
Tooltip
Toolbox
ComboBox
DateTimePicker
Label
ListBox
MenuStrip
TextBox
Timer
ToolStrip
To Do Reminder
File
Help
Date and time
2016-06-19 08:00
Priority
Less_important
Click to open calender for date, writ in time here.
ler new stuff
Add

## 3. Features and requirements for a Pass (G) grade

The application should be functioning well and it should control the user input so it does not crash or give unexpected output for invalid input.

3.1 The user should be able to select a date, time and priority and add a description for the task.

3.2 At program start, the Form should be clear of all design-time texts (Form1 as the title of the form is not accepted).

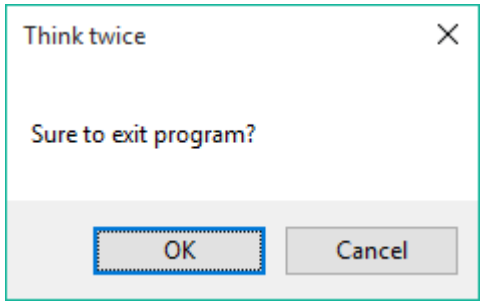
3.3 **Add a new Task:** The user clicks the Add-button after inputting a date a time, and giving a description (or name) of the task, to save the data into the ListBox (see the previous image).

3.4 The GUI should have a main menu bar consisting of the menus File and Help and the submenus as demonstrated in the previous figure. You may obviously have more menu items and sub-menu items.

3.5 The **File** menu should have the following sub-menus:

3.5.1.1 **New** with **ctrl-N** as a short-cut key: to reset the program exactly as at start-up. (The program should not terminate. All you have to do is to call the **InitilizeGUI** method).

3.5.1.2 **Exit** with **Alt-F4** as a short-cut key: to close the program after a confirmation message given to the user and when the user clicks the OK button in the message box. Do not exit the application if the user presses the Cancel button; nothing should change in this case.




3.5.2 **MessageBox**: If the user selects the Cancel button, the form should not exit and the application should continue as before.

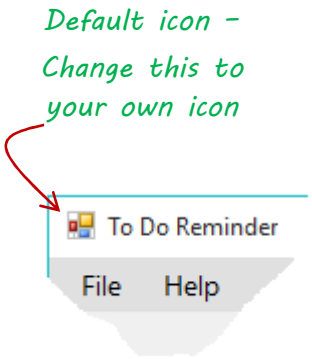
3.6 **Tooltip**: When the mouse is over the **DateTimePicker** control, a help message is to be displayed as the run example given earlier.

3.7 **Timer**: use an object of the Timer control to show the time on a Label (as displayed in the run example earlier).

3.8 **About Box**: When the user selects the sub-menu **Help** - **About**…, an About Box is to be displayed. Open the Properties >> **AssemblyInfo.cs** file and change the Description text and other relevant data. Change also the image to one of your own one, your photo, an icon or an image from the Internet. **Remember**: use a **PictureBox** control as a placeholder for the image.

## 4. Requirements and feature for a Pass with Distinction (VG) Grade

Skip this section if you are not going for a VG grade.



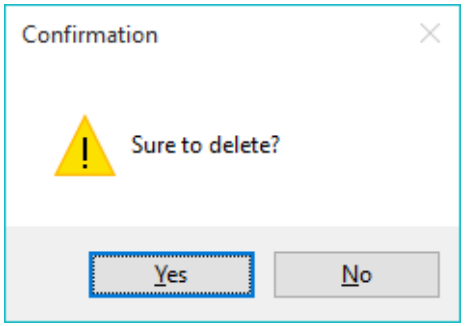

In addition to the requirements listed above for a G-grade, the following items are also to be done to qualify for a VG grade.

4.1 Create an icon for your application and replace the VS's default icon. You will need to draw both a 16 *16 and a 32 * 32 Pixels icon. Read the document "Working with Resources in VS" that is available in the module and get help from MSDN or other resources on the Internet.

4.2 The change and delete buttons must be disabled when the ListBox is empty, or when no item in the ListBox is selected (highlighted) by the user.

4.3 **Change a Task:** The user gives input using the input controls (just as adding a new task), and then selects an item in the ListBox. While the item in the ListBox is highlighted, the user clicks the Change button. The program then fetches the input and replaces the selected item with new task data..

4.4 **Delete a Task:** The user highlights an item in the ListBox and then presses the Delete button. Your program shows a message box letting the user have a chance to confirm or cancel deletion.




**Optional**: If you wish to do more, you can try to save the contents of the ListBox as strings into a file. You can read back the lines from the same file and display them in the ListBox through the menu File - Open. In the help document accompanying this assignment on Its L, you will find some code that you can use directly (using the copy-paste method).

If you would like to let the user select a file on a device, try the **SaveFileDialog** and **OpenFileDialog** controls.

**Remember**: File handling will be discussed thoroughly in the next course**, Programming in C# II**.

**Optional**: You can also test print-controls **PrintDialog**, **PrintDocument**, etc. in the Toolbox in VS to print out values to the printers connected to the computer.

## 5. Help och Guidance

It is expected that you solve this application by using your experiences from the previous assignments. However, a help document will be available on Its L.

## 6. Submission

Submit your assignment in the same way as the previous one.

Good Luck.

*Farid Naisan*
Course Coordinator and teacher