

# “TASK IT” – PERSONAL TIME MANAGER APPLICATION

Iurie GAITUR, scientific supervisor Mihail KULEV

Technical University of Moldova

**Abstract:** “Task it” application allows user to program the time more efficiently by recording and categorizing the tasks what have to be performed. Software will take care of reminding user how much time should be offerd for doing the task and how much remains from the full time required for finishing it.

**Keywords:** time manager application, time management, management techniques.

## 1. Introduction

The person becomes more productive using improved time management tools and can accomplish more with less effort and time. Time management can help to reduce wasted time and energy, helps become more creative and productive and enable to do the right thing at the right time. This will lead to more balance and fulfillment in person’s life.

## 2. Application description

Application provides user with necessary tools for efficient and adaptive time management.

When application starts, it remind to user the following things: the tasks what due dates are today, the tasks what due dates are tommorow and show the tasks that user didn’t accomplished or forgot to make (fig.1). At the same time, application starts the mini form (fig.2) where are shown the tasks that needs to be performed and the due dates when they expire.

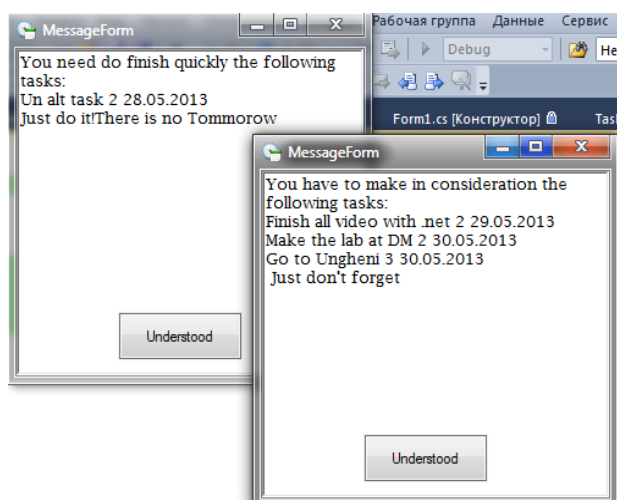


Fig.1. Screenshots of the reminders for user

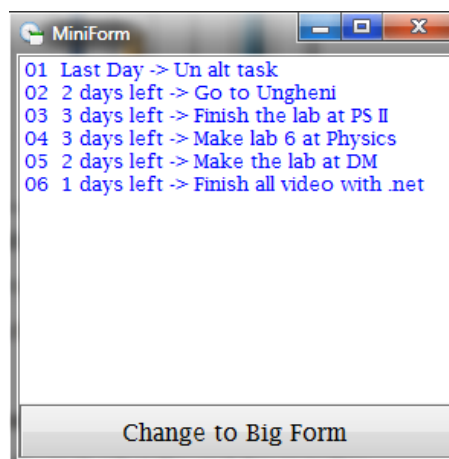


Fig.2. Screenshot of the mini form

By clicking the button „Change to Big Form”, appears the form (fig.3) with following options:

- Add a task (fig.4)
- Edit a task
- Delete a task
- Random task – in the moment when user can’t decide what task to perform, application proposes a random task from existing one
- Done! – is used for completing the task.

By clicking at one of the tasks, the user can see the details of the task.

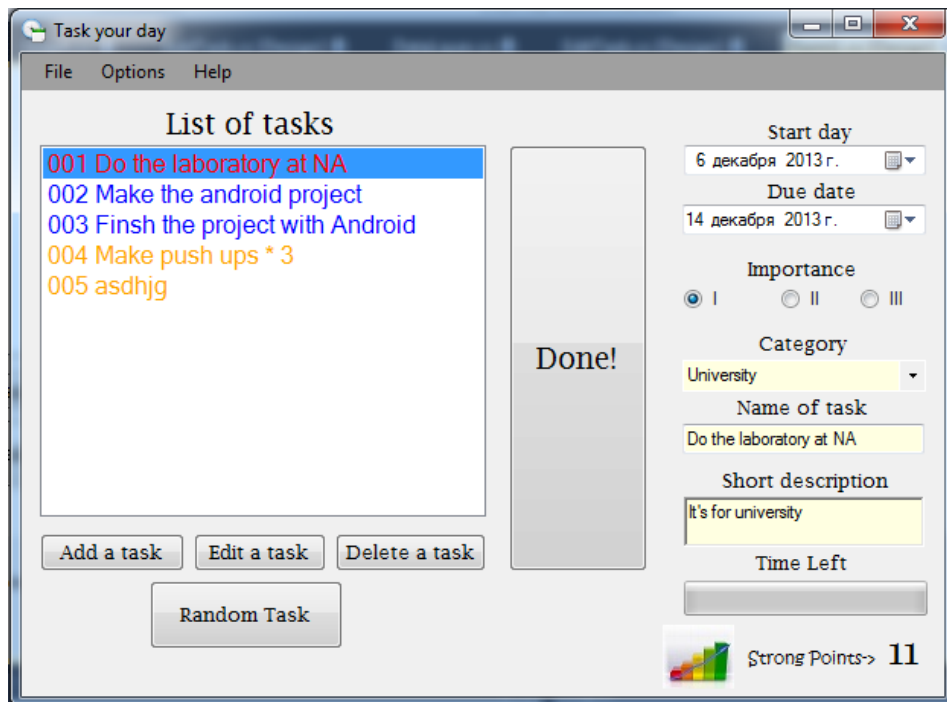


Fig.3.Screenshot of the application main form

By viewing the plot of the last passed tasks (fig.5), user can understand what tasks have been done and what haven't. In this way person can reconsider the actions and thinks over the idea how to distribute time better. Also according to the strong points that were given from the beginning, user can see how important were those tasks that had been done and how much time they took, comparing to others and by this to reconsider the distribution of the free time.

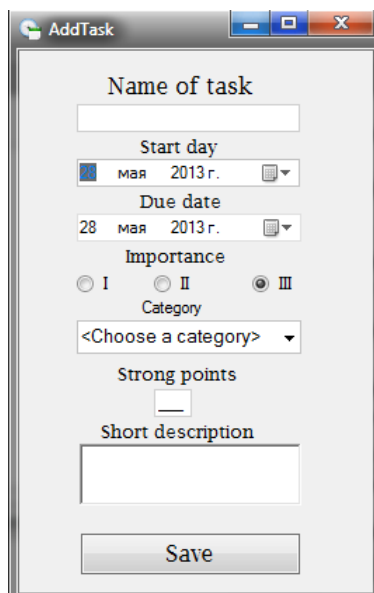


Fig.4.Form for adding a task

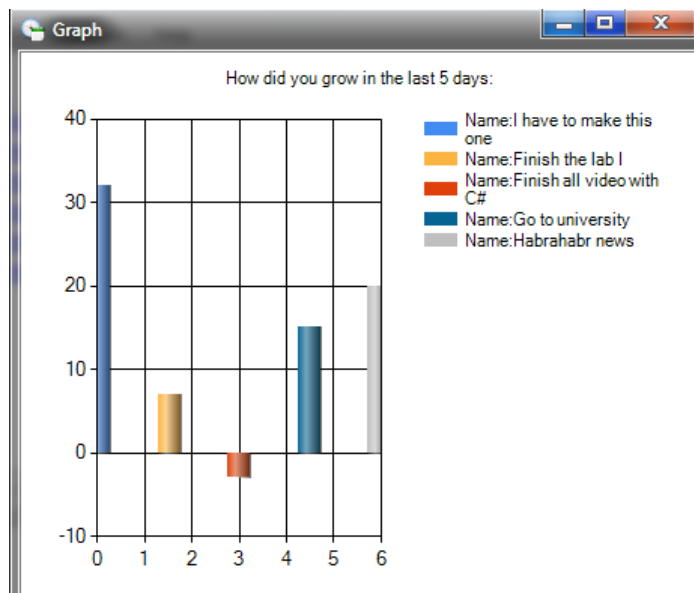


Fig.5.Graph of the strong points with last passed tasks

### Tools and code

The most important function that makes application more user friendly is the functionality that allows to start this program with Windows. It uses the register of the PC and adds a link to the application. In this way program begin with starting the Windows [4].

The code for this functionality is presented below:

```
private void addToAutorunToolStripMenuItem_Click(object sender, EventArgs e)
```

```

    {
        DialogResult dr =
            MessageBox.Show("Do you really want to add this app to
autorun?", "Autorun", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        if (dr == DialogResult.Yes)
        {
            Microsoft.Win32.RegistryKey regkey =
Microsoft.Win32.Registry.LocalMachine.OpenSubKey(@"SOFTWARE\Microsoft\Windows\C
urrentVersion\Run", true);
            regkey.SetValue("Task manager", Application.ExecutablePath);
            //Application.StartupPath
            if (MessageBox.Show("Do you want to restart you pc in order
to put the changes in function?",
                "Warning!", MessageBoxButtons.YesNo,
                MessageBoxIcon.Exclamation) == DialogResult.Yes)
            {
                RestartOs();
            }
        }
        else return;
    }
}

```

One of the most interesting parts of the application is how the program stores the information. It uses xml file for storing, that's why there is used a parser that parses the data – writes in and reads from file [1]. The code for such a parsing is listed below [2]:

```

public static bool AddNewTask(Task newTask, ref string message)
{
    if (CheckForSuchTask(newTask))
    {
        message = "You already put such a task in reminder!";
        return false;
    }
    FileStream fs;
    XmlDocument xml;
    try
    {
        using (fs = new FileStream(pathXml, FileMode.Open,
FileAccess.Read, FileShare.ReadWrite))
        {
            xml = new XmlDocument();
            xml.Load(fs);
            fs.Close();
        }
    }
    catch
    {
        message = "Cannot create a new task because the database is
crushed";
        return false;
    }
    try
    {
        XmlElement newItem;
        XmlElement newOuter = xml.CreateElement("Task");

```

```

foreach (KeyValuePair<string, string> kvp in
newTask.TaskInfo)
{
    newItem = xml.CreateElement(kvp.Key);
    newItem.InnerText = kvp.Value;
    newOuter.AppendChild(newItem);
}
xml.DocumentElement.InsertAfter(newOuter,
xml.DocumentElement.LastChild);

FileStream writer = new FileStream(pathXml,
FileMode.Truncate, FileAccess.Write, FileShare.ReadWrite);
xml.Save(writer);
writer.Dispose();
}
catch
{
    message = "An error hapened while creating a new task";
    return false;
}
message = "Task inserted";
return true;
}

```

### Conclusion

This application is effective in managing the time by making plans and adding tasks to accomplish in the list of tasks. By using it continuously user will observe what powerful tool is this program, because it offers the chance to see how person has been evolving over time by looking at the graphs what provides the data about how many strong point user have and the strong points during the last five tasks and either they were accomplished or not.

Application “Task it” can be upgraded by:

- Improving the user interface of the application mini form
- Changing the interface of the program to a better one in order to make application more attractive for users
- Developing the mobile version.

### Bibliography

1. *Parsing xml in c-a quick working sample/* <http://www.doublecloud.org/2013/08/parsing-xml-in-c-a-quick-working-sample/> (accessed 15.12.2013)
2. *How to parse XML with XMLReader /* [http://msdn.microsoft.com/ru-ru/library/cc189056\(v=vs.95\).aspx](http://msdn.microsoft.com/ru-ru/library/cc189056(v=vs.95).aspx) (accessed 15.12.2013)
3. *Autorun the Application using C#/* <http://stackoverflow.com/questions/7608225/autorun-the-application-using-c-sharp> (accessed 15.12.2013)