Gabriel Molina

**Launch Pad for Windows**

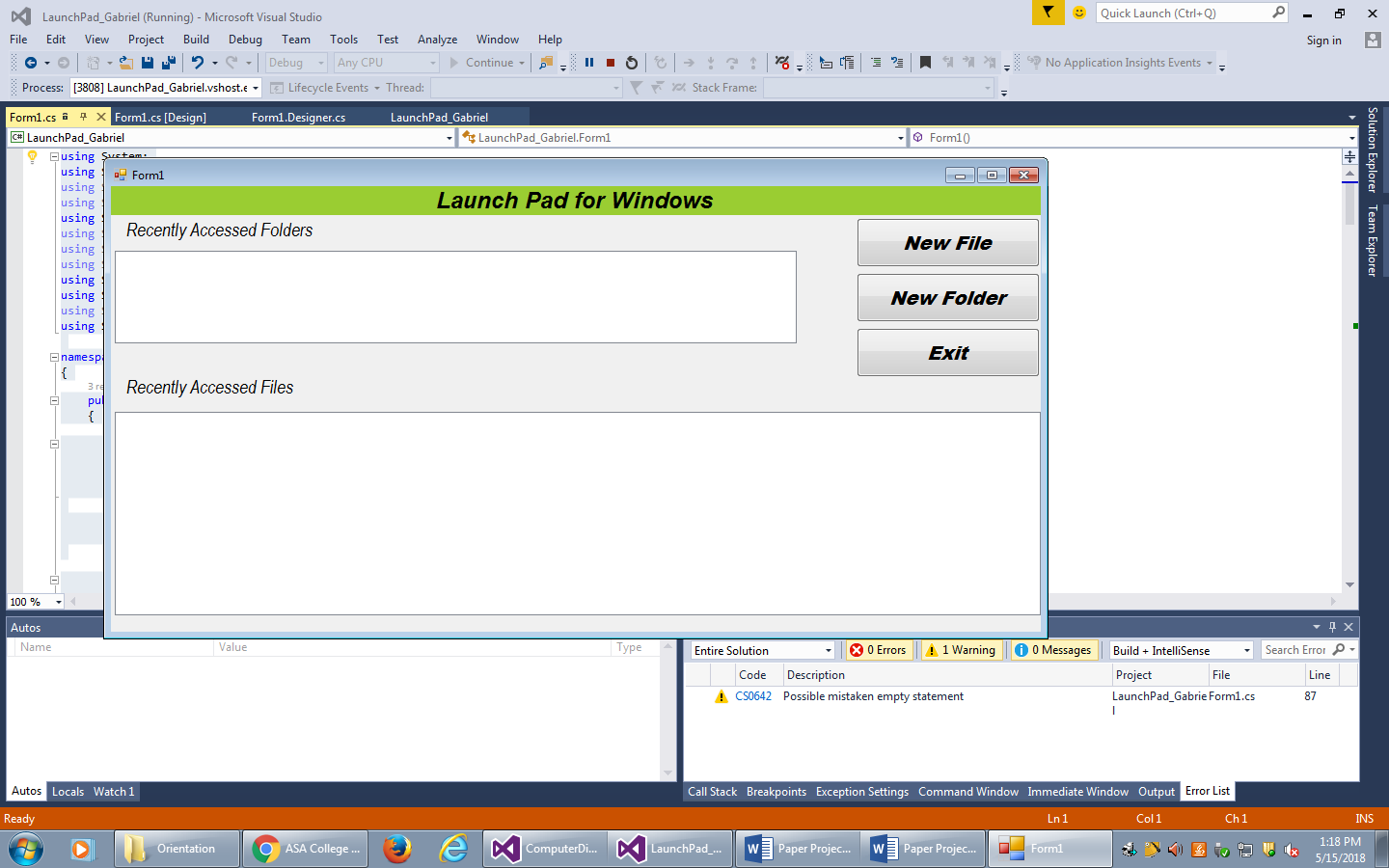
Windows has a lot of application that makes itself functionality and efficient like the other Operating Systems. One of them is Paint, it an application that can you manipulate any image from the computer. Another application is the Media Player, it’s a program that plays media files from the computer video or audio locally. In this project “Launch Pad for Windows” is a good resource because it helps the people to look any file on your computer and leave a history. Basically is a small application that connects to the server to look any file on the computer. The advantage about this application is that shows the person the files and the folder that he or she was before, it’s meant that the person can open and see the files and folder again, and doesn’t back to look again the file or folder that he or she wants to open.

Imagine the people that work in a computer store, and the person in charge wants to back up to a computer. He or she usually does first is to save the files and the folder, every importation and essential information. For making the life easier, the person can install this application “Launch Pad for Windows” and use it for his or her commodity. People can say how is going to be easier than before. With this application, the person doesn’t look again between the bunch of files and folder that the computer has It. Just him or she has to press click on the meaning folder and it is going to she appeared in the history that the application has it.

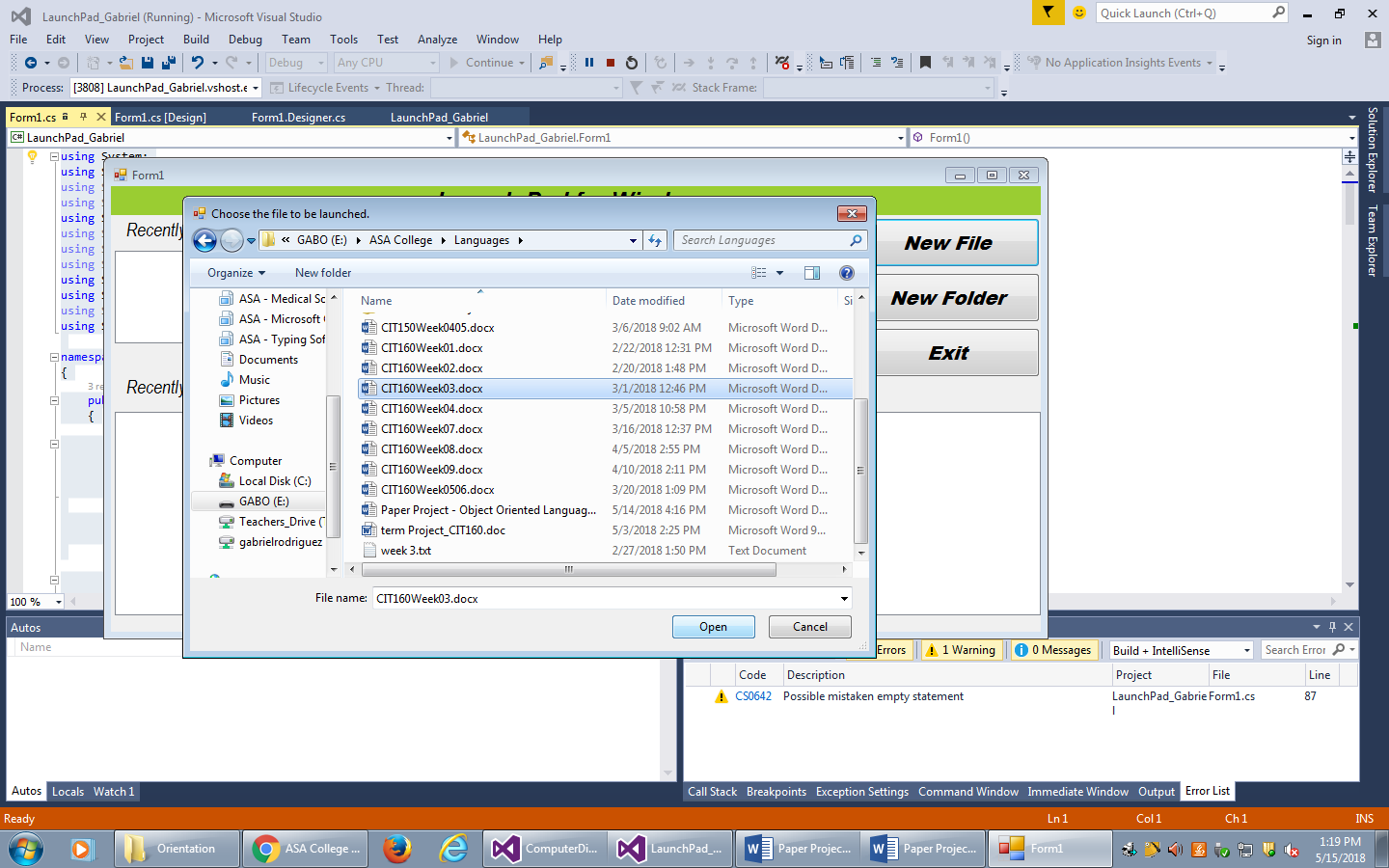
Another example of the advantage of this application is that anybody can use it for his or her own business. This program can be useful for the programmers when they are doing a program and need to check the work, or a regular person forgets where the files or the folder is. Also, a secretary to manage the information faster about the information that she went before and she needs to look it again.

**Screenshots**

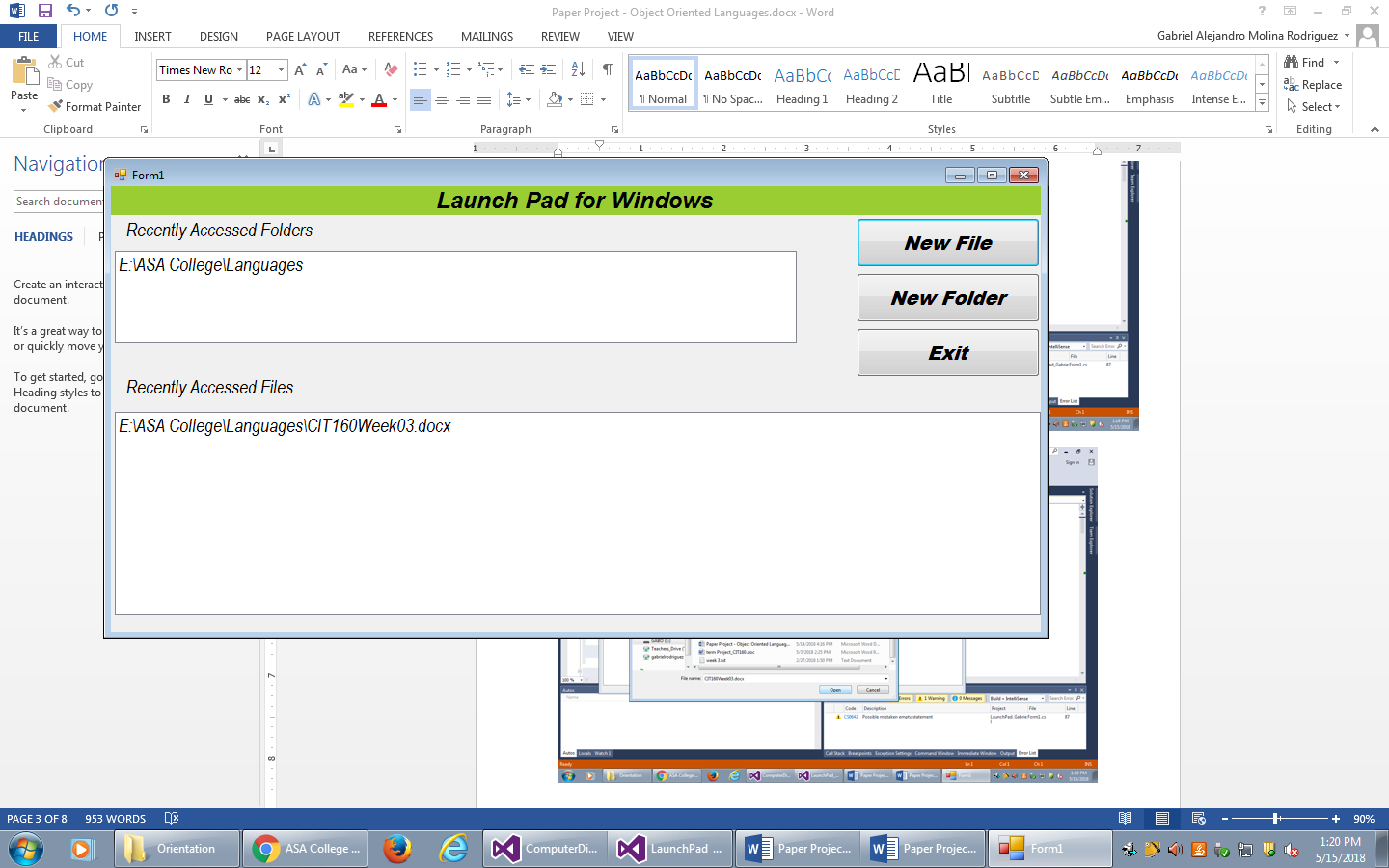
1. Opening the program



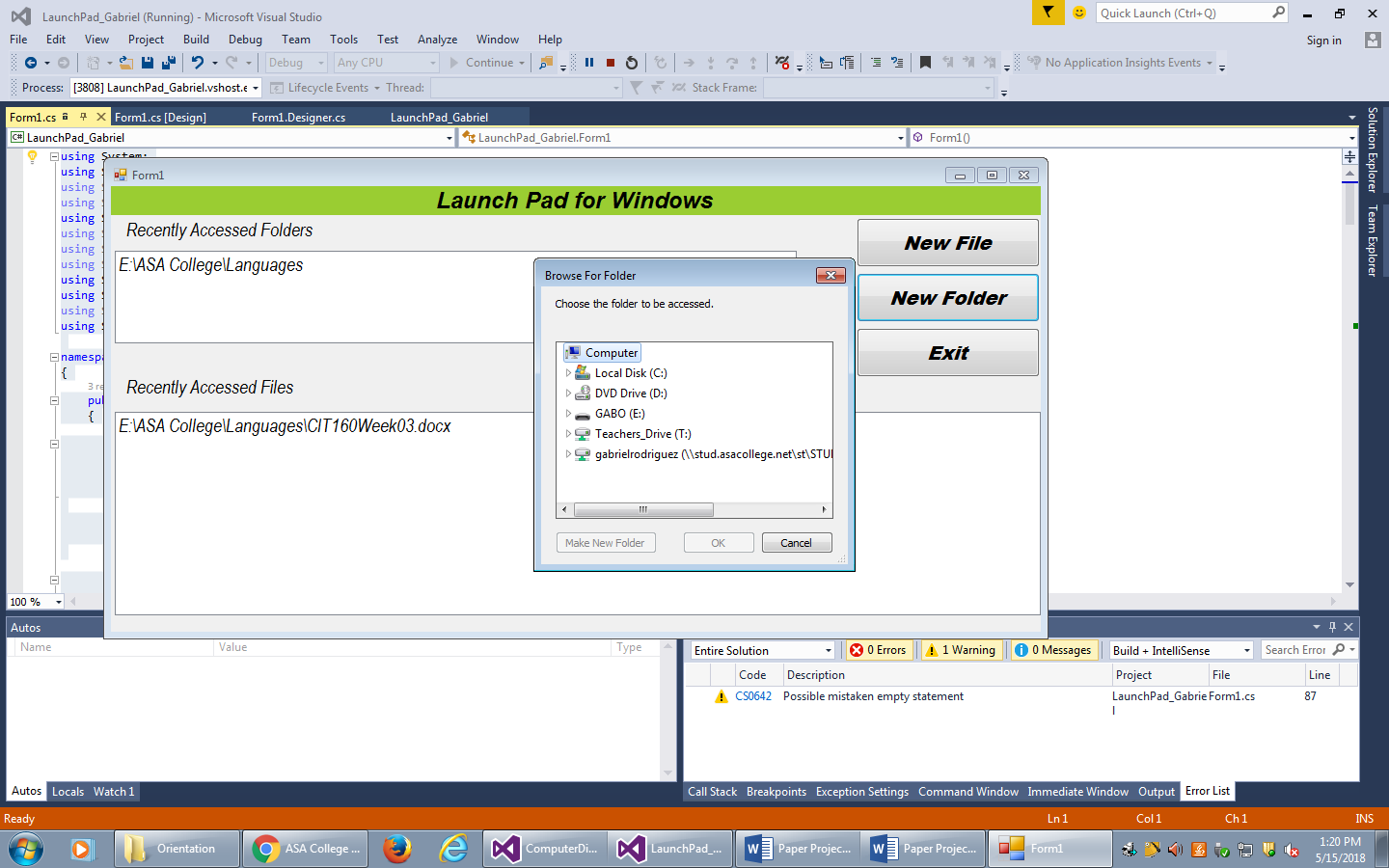
1. Running when pressed the New File button



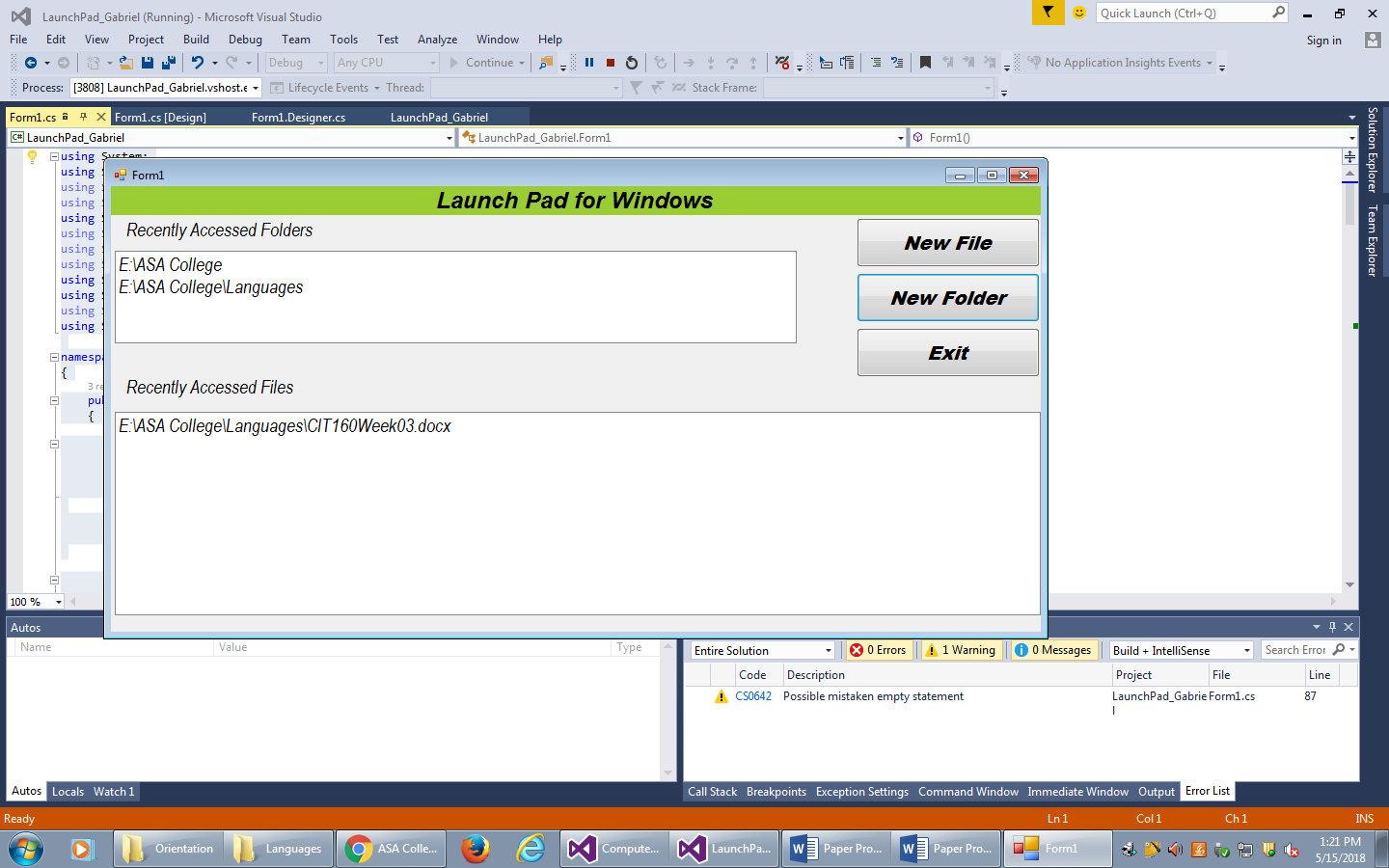
1. After finishing the action



1. Running when pressed the New Folder button



1. After finishing the action



**Code**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO;

using System.Reflection;

using System.Diagnostics;

namespace LaunchPad\_Gabriel

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

// Creating a list which is an object which holds variables in a specific order.

// The type of variable that the list can store is defined using the generic syntax.

// Here is an example of defining a list called string which holds integers.

List<string> folderList = new List<string>();

List<string> fileList = new List<string>();

// Creating a method that takes no parameters and returns no value is declared as

// follows public void, this method allow to get access to the Files

public void PathFileToListandListBox(string pathFile, ListBox lb, List<string> ls)

{

StreamReader sr = new StreamReader(pathFile);

lb.Items.Clear();

ls.Clear();

while (sr.Peek() > -1)

{

ls.Add(sr.ReadLine());

}

lb.Items.AddRange(ls.ToArray());

}

// Writing this code helps you when you open the program, it clears the files or the lines on the ComboBox

public void RefreshListBox(ListBox lb, List<string> list)

public void RefreshListBox(ListBox lb, List<string> list)

{

lb.Items.Clear();

lb.Items.AddRange(list.ToArray());

}

public void UpdatePathFile(List<string> list, string pathFile)

{

StreamWriter sw = new StreamWriter(pathFile, false);

foreach (string line in list)

{

sw.WriteLine(line);

}

sw.Close();

}

public void AddPathToList(List<string> lst, string item)

{

if (lst.Contains(item))

{

lst.Remove(item);

}

lst.Insert(0, item);

}

private void flowLayoutPanel1\_Paint(object sender, PaintEventArgs e)

{

}

// This code on the Form is showing where to get the files and

// folder which means you are using AppDomain to direct your project

private void Form1\_Load(object sender, EventArgs e)

{

string specialFolder = AppDomain.CurrentDomain.BaseDirectory + @"LaunchPadFiles";

string MRUfilePath = specialFolder + @"\MRUfiles.txt";

string MRUfoldersPath = specialFolder + @"\MRUfolder.txt";

if (!Directory.Exists(specialFolder))

{

Directory.CreateDirectory(specialFolder);

}

if (!File.Exists(MRUfilePath))

{

File.CreateText(MRUfilePath);

}

if (!File.Exists(MRUfoldersPath)) ;

{

File.CreateText(MRUfoldersPath);

}

if (!Size.Equals(Properties.Settings.Default.appSize, new Size(0, 0)))

{

this.Size = Properties.Settings.Default.appSize;

}

if (!Size.Equals(Properties.Settings.Default.appLocation, new Size(0, 0)))

{

this.Location = Properties.Settings.Default.appLocation;

}

PathFileToListandListBox(MRUfilePath, lstFiles, fileList);

PathFileToListandListBox(MRUfoldersPath, lstFolders, folderList);

}

// Writing this code you can save the settings variables FormClosing

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

string specialFolder = AppDomain.CurrentDomain.BaseDirectory + @"LaunchPadFiles";

string MRUfilePath = specialFolder + @"\MRUfiles.txt";

string MRUfoldersPath = specialFolder + @"\MRUfolder.txt";

UpdatePathFile(fileList, MRUfilePath);

UpdatePathFile(folderList, MRUfoldersPath);

Properties.Settings.Default.appSize = this.Size;

Properties.Settings.Default.appLocation = this.Location;

Properties.Settings.Default.Save();

}

// Code for create a New File button, it does to open a new file.

// In this code is used Process component, it is a useful tool for

// starting, stopping, controlling, and monitoring applications

private void btnNewFile\_Click(object sender, EventArgs e)

{

OpenFileDialog oFile = new OpenFileDialog();

oFile.Title = "Choose the file to be launched.";

oFile.InitialDirectory = Environment.SpecialFolder.MyComputer.ToString();

if (oFile.ShowDialog() == DialogResult.OK)

{

string newFile = oFile.FileName;

AddPathToList(fileList, newFile);

RefreshListBox(lstFiles, fileList);

string newFolder = Path.GetDirectoryName(newFile);

AddPathToList(folderList, newFolder);

RefreshListBox(lstFolders, folderList);

Process proc = new Process();

proc.StartInfo.FileName = newFile;

proc.StartInfo.UseShellExecute = true;

proc.Start();

}

}

// Code for create a New File button, it does to open a new folder.

// Using the Process component, you can obtain a list of the processes that are running,

// or you can or start a new process.

private void btnNewFolder\_Click(object sender, EventArgs e)

{

FolderBrowserDialog oFolder = new FolderBrowserDialog();

oFolder.Description = "Choose the folder to be accessed.";

oFolder.RootFolder = Environment.SpecialFolder.MyComputer;

oFolder.ShowNewFolderButton = true;

if (oFolder.ShowDialog() == DialogResult.OK)

{

string newFolder = oFolder.SelectedPath;

AddPathToList(folderList, newFolder);

RefreshListBox(lstFolders, folderList);

Process proc = new Process();

proc.StartInfo.FileName = newFolder;

proc.StartInfo.UseShellExecute = true;

proc.Start();

}

}

// with this code we can open the folders on the list

// that we were before with the method DoubleClick

private void lstFolders\_DoubleClick(object sender, EventArgs e)

{

if (lstFiles.SelectedItem != null)

if (Directory.Exists(lstFolders.SelectedItem.ToString()))

{

Process.Start(@lstFolders.SelectedItem.ToString());

AddPathToList(folderList, lstFolders.SelectedItem.ToString());

RefreshListBox(lstFolders, folderList);

}

else

{

folderList.Remove(lstFolders.SelectedItem.ToString());

RefreshListBox(lstFolders, folderList);

}

}

// with this code we can open the

//files on the list that we were before with the method DoubleClick

private void lstFiles\_DoubleClick(object sender, EventArgs e)

{

if (lstFiles.SelectedItem != null)

if (File.Exists(lstFiles.SelectedItem.ToString()))

{

Process proc = new Process();

proc.StartInfo.FileName = lstFiles.SelectedItem.ToString();

proc.StartInfo.UseShellExecute = true;

proc.Start();

AddPathToList(fileList, lstFiles.SelectedItem.ToString());

RefreshListBox(lstFiles, fileList);

}

else

{

fileList.Remove(lstFiles.SelectedItem.ToString());

RefreshListBox(lstFiles, fileList);

}

}

// Code for the exit button, it stops the program

private void btnExit\_Click(object sender, EventArgs e)

{

this.Close();

}

}

}

**Recommendations**

This project is the focus to have a files reminder, instead, people can improve the program or application adding new things. The first improvement is that the person can put a new button named View, with this button people can see only the view of the file that he or she wants instead to open it and look what it is. Another improvement creates a button that allows the person to create a new file either in Word or Notepad. This button can be called Create a New File, it can be an advantage for the students because they won’t go to look the applications of Word of Notepad they have it on the Launch Pad for Windows. Also, people can put in the interface to show when he or she visited the file or folder. In this case, the people see when the files or folder were visited, it helps the person a little more than unusual because it shows you the specific time that the person was.