|  |
| --- |
| COLEGIUL NAȚIONAL TUDOR VLADIMIRESCU, TÂRGU JIU |
| **PROIECT DE ATESTARE PROFESIONALĂ LA INFORMATICĂ** |
| ManageRStudio Aplicație utilitară C# Windows Forms |
|  |
| ELEV: STOCHIȚOIU RADU-DUMITRU |

**PROFESORI COORDONATORI:**

**NODEA GABRIELA**

**NODEA EUGEN**

|  |
| --- |
| 2013 - 2014 |

**CUPRINS**

|  |  |  |
| --- | --- | --- |
| I. | Argument | 3 |
| II. | Prezentarea proiectului | 4 |
| III. | Listing | 22 |
| IV. | Concluzii | 36 |
| V. | Bibliografie | 37 |
| VI. | Anexă CD/ DVD | 38 |

**I. ARGUMENT**

Având în vedere multitudinea de aplicații distincte ce pot rula pe un computer, este de înțeles dorința oamenilor de a extinde constant gama de funcții ce pot fi îndeplinite într-un sistem de operare.

Tehnologia se extinde în fiecare clipă și fiecare persoană dorește eficiență și rapiditate în activitatea sa.

Motivul pentru care am ales crearea unei aplicații utilitare este în principal unul practic: grupând într-o casetă estetică Windows funcțiile importante și frecvent utilizate, le putem accesa rapid și în timp util astfel încat să răspundă necesităților zilnice.

Am creat o alarmă modernă cu posibilitatea alegerii melodiei pentru declanșare, un browser web, un player mp3, unul video, un editor de poze și un player slideshow, aplicații pe care le consider fudnamentale în ziua de astăzi.

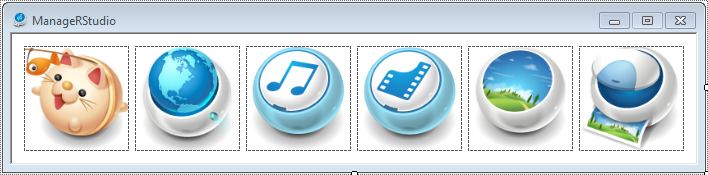
Sunt de părere că o astfel de aplicație utilitară, cu posibilitatea de a fi minimalizată în bara de iconițe ascunse, pentru o comportare silențioasă, reprezinta o aplicație practică și eficientă pe care, personal, o consider utilă oricărui utilizator al unui calculator.

**II. PREZENTAREA PROIECTULUI**

Am creat acest proiect de atestare profesională în limbajul de programare **C# / Windows Forms ( Visual Studio 2012 )** deoarece acesta dispune de o multitudine de unelte care pot fi de ajutor atunci când lucrezi cu fișiere executabile.

Interfața inițială este alcătuită din:

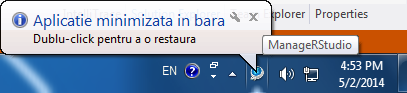
* O formă Windows de dimensiuni fixe, blocată 3D (MainR.cs)
* 6 controale de tip PictureBox
* O iconiță de notificare pentru minimizarea aplicației



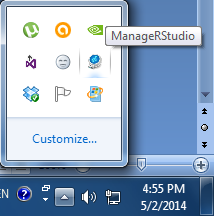
Dacă în orice moment după începerea rulării aplicației se apasă click pe butonul de minimizare,



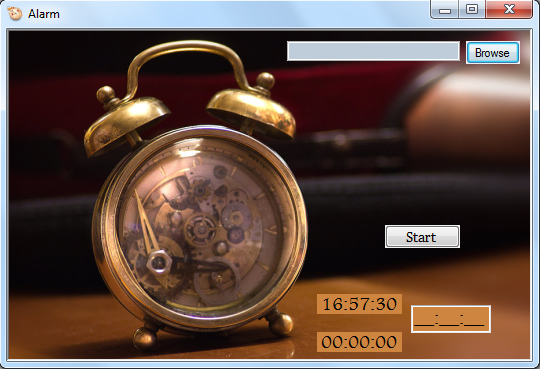
aplicația va dispărea de pe ecran și din bara de start, iar iconița de notificare se va poziționa afișând un mesaj corespunzător în meniul de “Iconițe ascunse” în bara Start.



Aplicația se poate restaura dacă utilizatorul efectuează dublu-click pe iconiță în interiorul meniului.



Apăsându-se click pe primul control de tip PictureBox se va afișa o nouă fereastră care simulează o alarmă:

.

Într-un control de tip Label se afișează ora exactă.

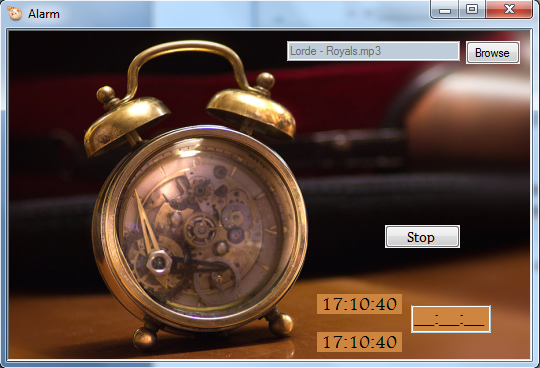
La apăsarea butonului “Browse” se deschide o nouă fereastră care permite selectarea unui fișier de tip mp3. După apăsarea butonului “Open” numele fișierului va apărea în controlul de tip TextBox din stânga butonului.



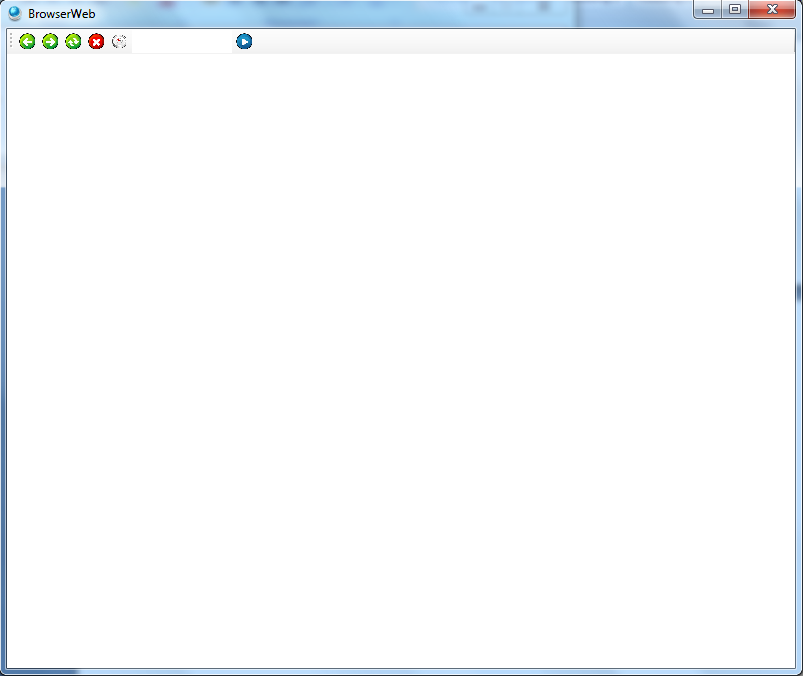
7.png

După introducerea unei ore în controlul de tip mască ce afișează textul „\_\_:\_\_:\_\_” și apăsarea butonului “Start”, controlul TextBox care este inițializat cu textul “00:00:00” va prelua textul controlului mască.

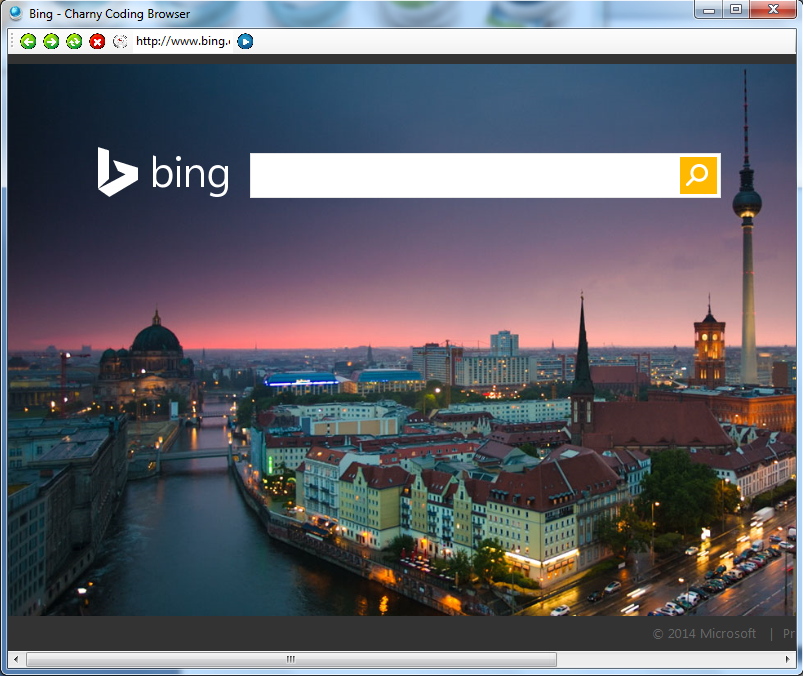
În momentul în care ora va ajunge exact cea dorită, introdusă anterior, va porni redarea fișierului mp3 selectat.



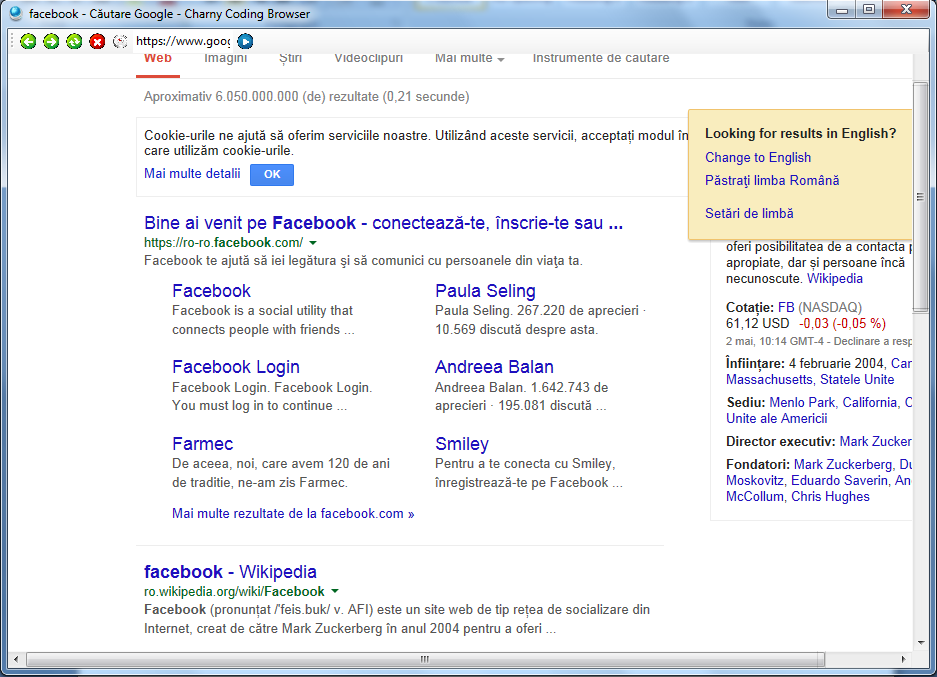
În momentul apăsării click pe cel de-al doilea control de tip PictureBox din interfața inițială se va deschide o nouă fereastră care simulează un browser web, inițial neaccesând nicio pagină (blank).



Se poate schimba pagina de pornire predefinită, aceasta fiind inițial [www.bing.com](http://www.bing.com).



Exemplu de pagină web accesată din interiorul aplicație:



Butonul **Înapoi** oferă posibilitatea întoarcerii pe pagina anterioară.

12.png

Butonul **Următorul** oferă posibilitatea întoarcerii pe pagina de pe care m-am întors pe cea actuală.

13.png

Butonul **Refresh** oferă posibilitatea reîmprospătării paginii actuale.

14.png

Butonul **Stop** oprește toate descărcările de date și blochează pagina în stadiul în care a rămas.

15.png

Butonul **Home** accesează instant pagina prestabilită [www.bing.com](http://www.bing.com).

16.png

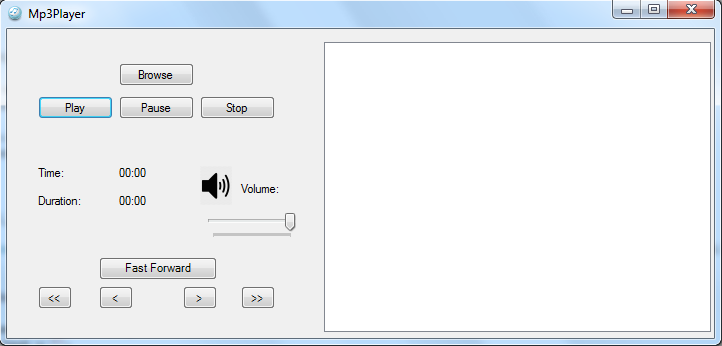
Căsuța de tip **TextBox** permite inserarea adresei paginii web pe care dorim să navigăm.

17.png

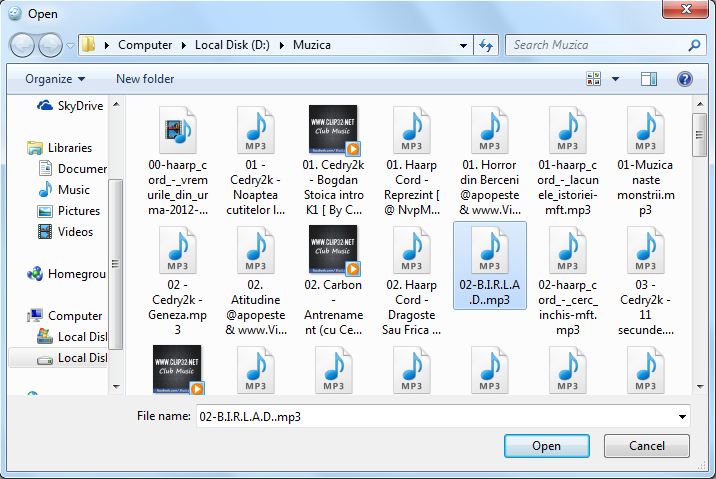
Butonul **Execută**, cu alternativa apăsării tastei “Enter” ne va direcționa către site-ul a cărui adresă se află în căsuța anterioară.

18.png

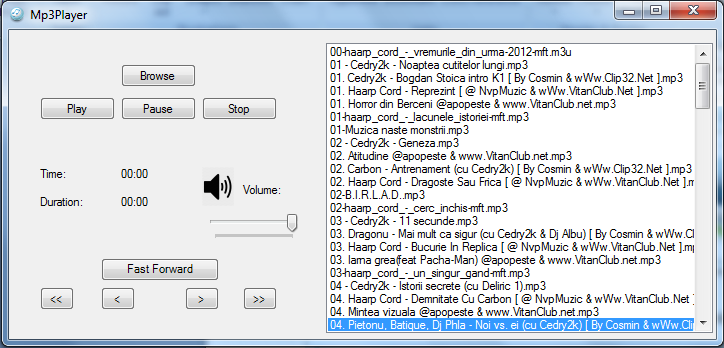
În momentul apăsării pe cel de-al treilea control de tip PictureBox se va deschide o nouă fereastră care va simula un Mp3Player.



În momentul apăsării butonului “Browse” se va deschide o fereastră care permite selectarea unuia sau a mai multor fișiere de tip mp3.

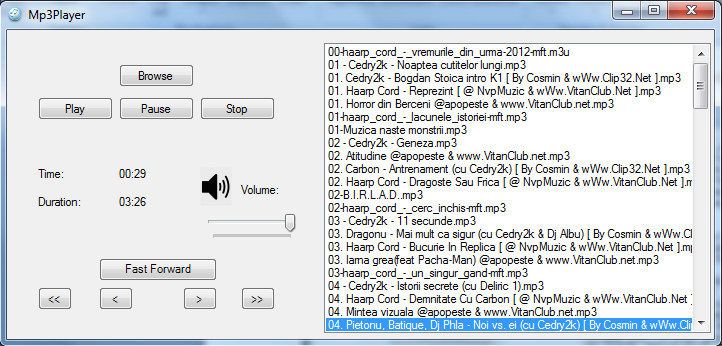


După selectarea acestora, vor fi afișate în interiorul ferestrei principale.



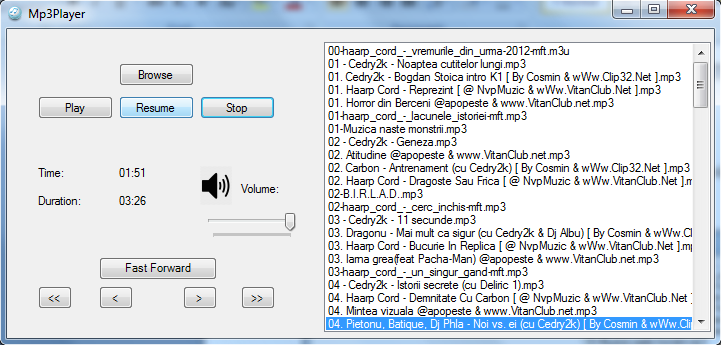
La acționarea unui dublu-click sau a butonului “Play” va începe redarea fișierului mp3 selectat.

Timpul va fi calculat și reținut de un control de tip Timer și va fi afișat corespunzător împreună cu durata totală a melodiei.



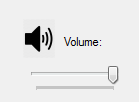
În momentul acționării butonului de pauză, player-ul se va opri automat din redare și butonul va fi înlocuit cu unul de continuare a melodiei din punctul rămas.

Timer-ul se va opri, reținând secunda la care a rămas.

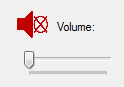
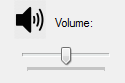


La acționarea butonului “Stop” se va opri redarea melodiei, contorul de secunde revenind la 00:00.

Volumul poate fi ajustat foarte ușor cu ajutorul unui control TrackBar.



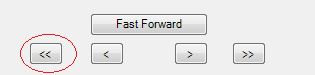
Se poate apăsa chiar pe imagine pentru a trece pe mut. La a doua acționare se revine la volumul la care am rămas.

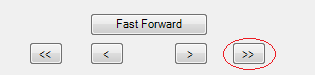
Butonul “Fast Forward” permite redarea fișierului mp3 la o viteză mult mai mare.

****

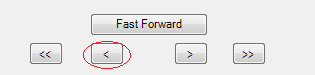
Acest buton va reda automat primul fișier din listă.

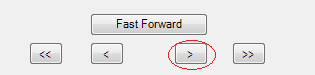


Analog, acesta îl va reda pe ultimul.

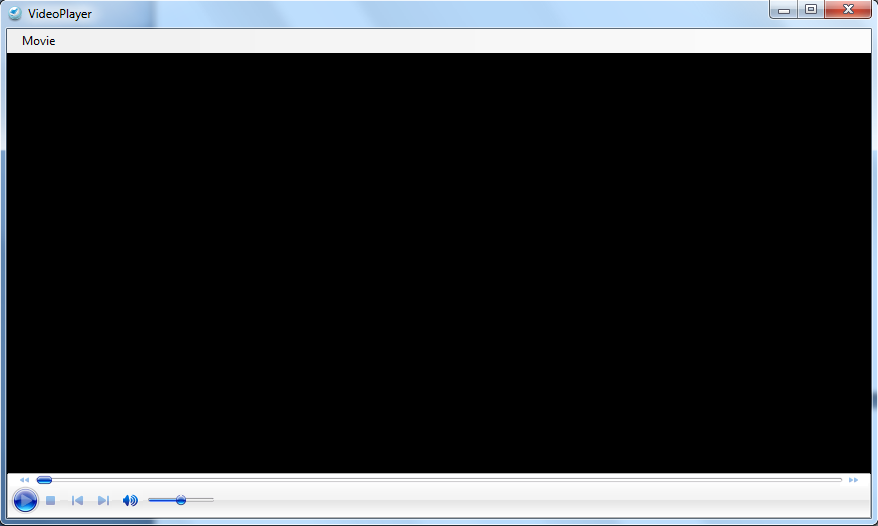


Cele două butoane din mijloc permit redarea melodiei anterioare/ urmatoare.

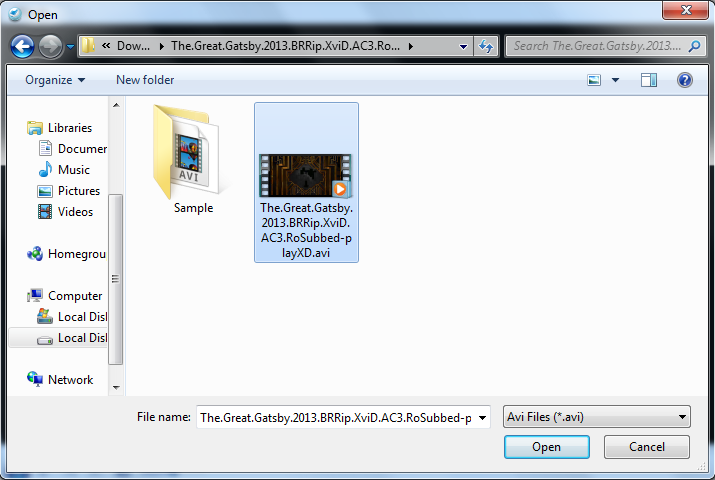




La selectarea celui de-al patrulea control de tip PictureBox se va deschide o fereastră care va simula un player video.



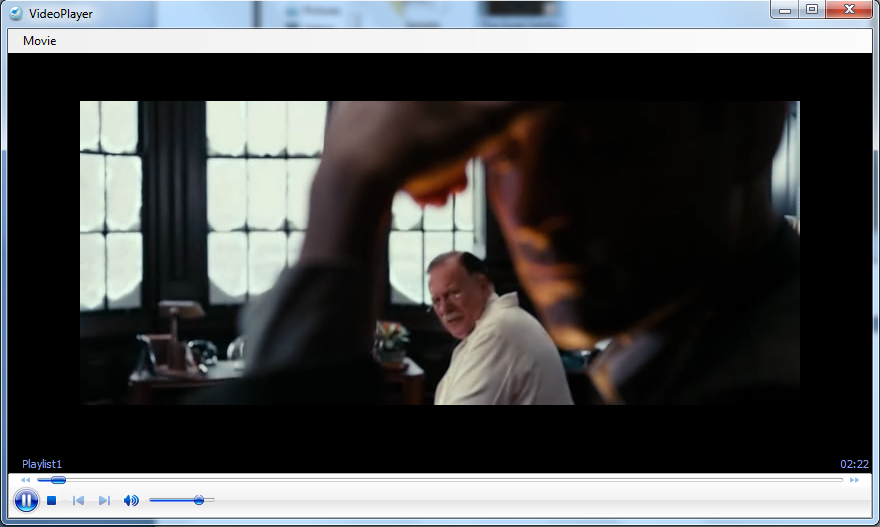
La acționarea butonului “Movie” se va deschide o fereastră de unde vom putea selecta fișierul media dorit.



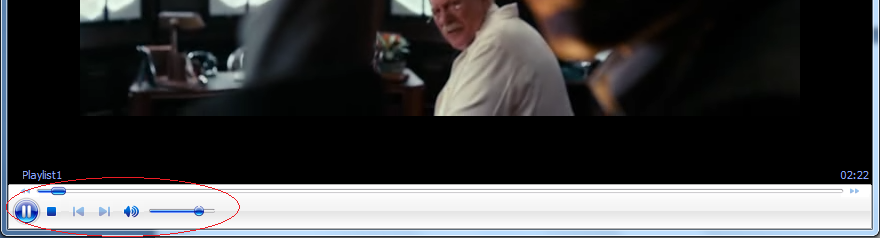
Aplicația permite căutarea exclusivă (filtrarea) a extensiei fișierului dorit.



Exemplu de redare video:



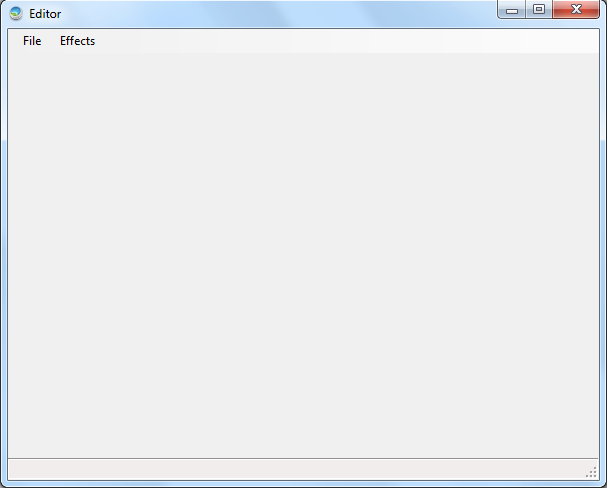
Meniul de butoane este cel specific Windows Media Player, deoarece controlul folosit pentru redare este importat chiar din resursele Windows Media Player.



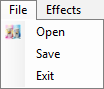
De asemenea, acest video player permite redarea fișierelor mp3.



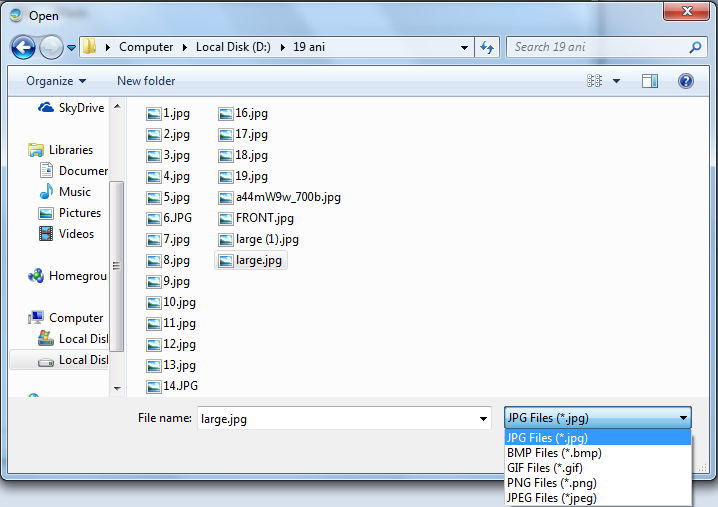
În momentul acționării celui de-al cincilea control de tip PictureBox se va deschide o nouă fereastră care simulează un editor foto.



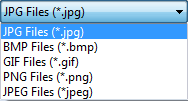
La apăsarea butonului “File” din meniu, se va deschide un submeniu.



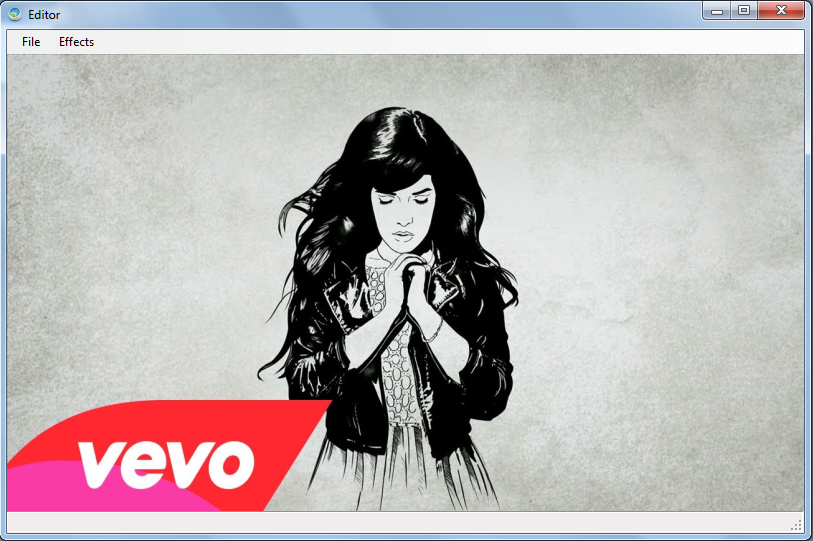
La apăsarea butonului “Open” se va deschide o nouă fereastră care permite selectarea imaginii.



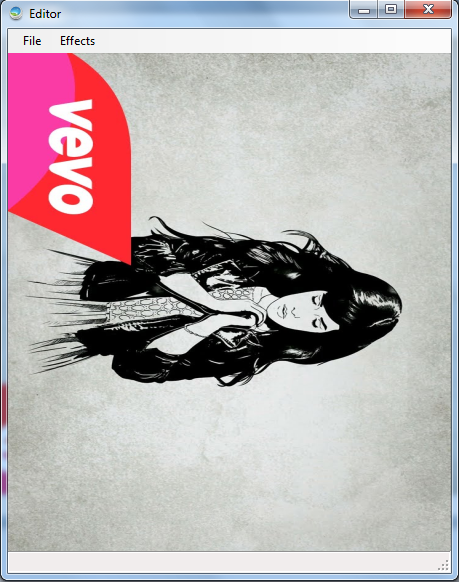
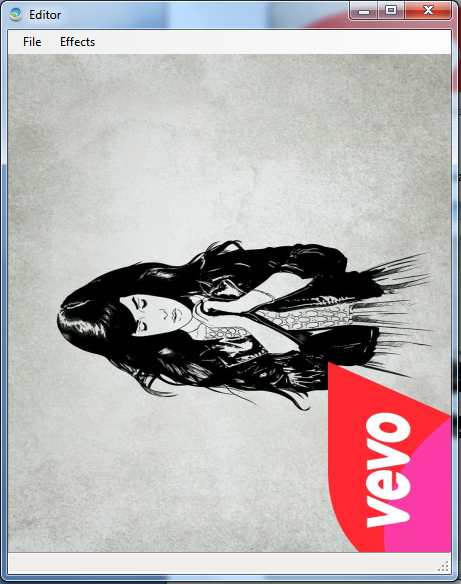
Se permite filtrarea extensiei.



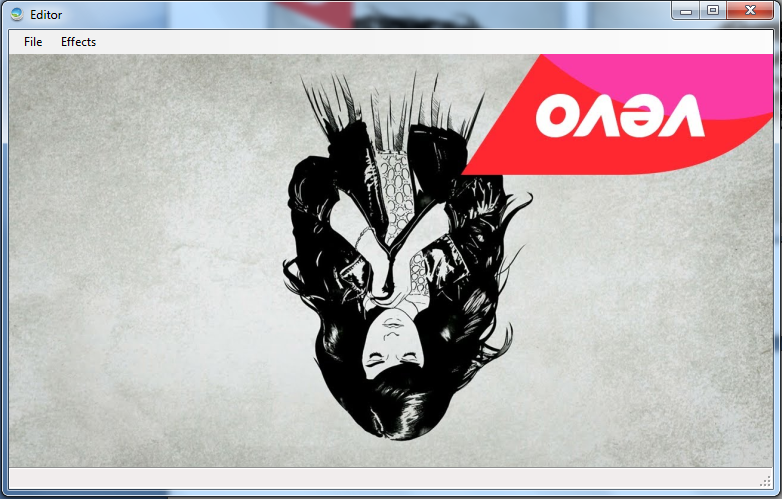
După selectarea fișierului acesta va apărea în fereastra activă.



Din submeniul “Effects” se poate selecta rotirea la 90 de grade in orice directie.

De asemenea, imaginea se poate roti si la 180 de grade.



Butonul “GrayScale” va transforma imaginea actuală într-una alb-negru (considerăm imaginea folosită inițial).

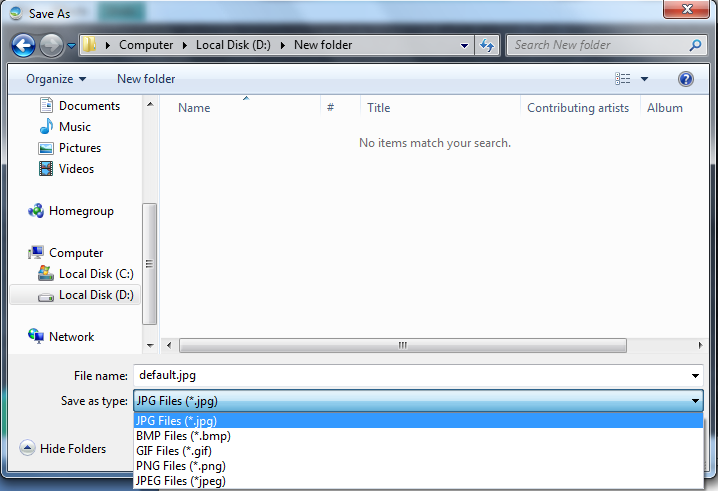


Un buton “Undo” va apărea în meniu, iar la selectarea acestuia se poate revoca ultima acțiune.

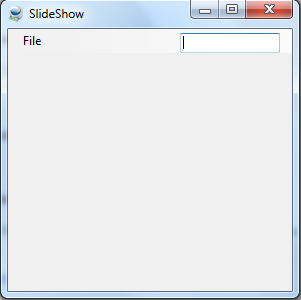
Culorile pot, de asemenea, fi inversate dând click pe butonul “Invert Colors”.



În orice moment al editării se poate salva noua imagine din “File-Save” sub extensia dorită.

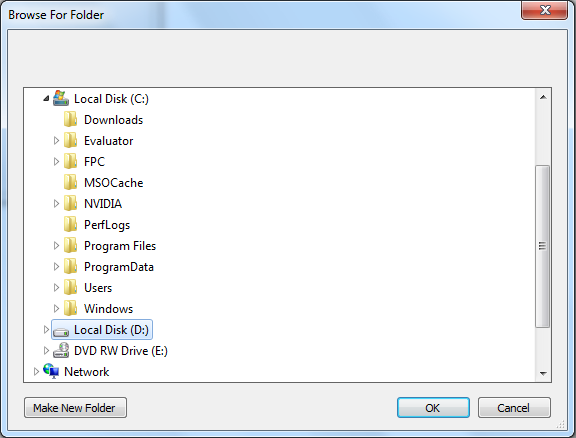


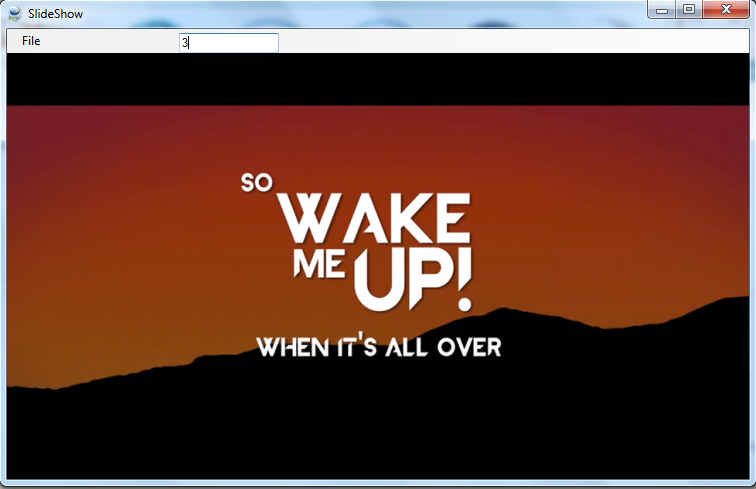
La acționarea unui click asupra ultimului control PictureBox se va deschide o nouă fereastră care permite afișarea pozelor dintr-un folder selectat sub forma unor diapozitive.



În căsuța selectată predefinit se va introduce o valoare întreagă care reprezintă durata dintre diapozitive. (aceasta se poate modifica în timpul afișării, adică în timp real!!!)

Apăsând “File-Choose Folder” se va deschide o fereastră care va permite selectarea unui folder din care dorim să selectăm pozele.







Fereastra se va redimensiona automat în funcție de dimensiunile fotografiei afișate.

**III. LISTING**

**MainR.cs**

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;

namespace ManageRStudio

{

public partial class MainR : Form

{

public MainR()

{

InitializeComponent();

}

private void pictureBox1\_Click(object sender, EventArgs e)

{

Form Alarma = new Alarm();

Alarma.Show();

}

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

{

}

private void MainR\_Resize(object sender, EventArgs e)

{

if (FormWindowState.Minimized == this.WindowState)

{

notifyIcon1.Visible = true;

notifyIcon1.ShowBalloonTip(500);

this.Hide();

}

else if (FormWindowState.Normal == this.WindowState)

{

notifyIcon1.Visible = false;

}

}

private void notifyIcon1\_MouseDoubleClick(object sender, MouseEventArgs e)

{

this.Show();

this.WindowState = FormWindowState.Normal;

}

private void pictureBox2\_Click(object sender, EventArgs e)

{

Form Video = new VideoPlayer();

Video.Show();

}

private void pictureBox3\_Click(object sender, EventArgs e)

{

Form Editor = new Editor();

Editor.Show();

}

private void pictureBox4\_Click(object sender, EventArgs e)

{

Form Web = new BrowserWeb();

Web.Show();

}

private void pictureBox5\_Click(object sender, EventArgs e)

{

Form mp3 = new Mp3Player();

mp3.Show();

}

private void pictureBox6\_Click(object sender, EventArgs e)

{

Form SlideS = new SlideShow();

SlideS.Show();

}

}

}

**Alarm.cs**

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;

namespace ManageRStudio

{

public partial class Alarm : Form

{

public Alarm()

{

InitializeComponent();

timer1.Start();

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

string song;

private void button1\_Click(object sender, EventArgs e)

{

OpenFileDialog open = new OpenFileDialog();

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

{

song = open.FileName;

textBox1.Text = open.SafeFileName;

}

}

private void timer1\_Tick(object sender, EventArgs e)

{

label1.Text = DateTime.Now.Hour.ToString("00") + ":" + DateTime.Now.Minute.ToString("00") + ":" + DateTime.Now.Second.ToString("00");

}

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

{

label2.Text = maskedTextBox1.Text;

maskedTextBox1.Text = "";

timer2.Start();

startButton.Visible = false;

stopButton.Visible = true;

}

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

{

axWindowsMediaPlayer1.Ctlcontrols.stop();

timer2.Stop();

startButton.Visible = true;

stopButton.Visible = false;

label2.Text = "00:00:00";

}

private void timer2\_Tick(object sender, EventArgs e)

{

if (label1.Text == label2.Text)

axWindowsMediaPlayer1.URL = song;

}

}

}

**BrowserWeb.cs**

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;

namespace ManageRStudio

{

public partial class BrowserWeb : Form

{

public BrowserWeb()

{

InitializeComponent();

}

private void toolStripButton6\_Click(object sender, EventArgs e)

{

webBrowser1.Navigate(toolStripTextBox1.Text);

}

private void toolStripButton2\_Click(object sender, EventArgs e)

{

webBrowser1.GoForward();

}

private void toolStripButton1\_Click(object sender, EventArgs e)

{

webBrowser1.GoBack();

}

private void toolStripButton3\_Click(object sender, EventArgs e)

{

webBrowser1.Refresh();

}

private void toolStripButton4\_Click(object sender, EventArgs e)

{

webBrowser1.Stop();

}

private void toolStripButton5\_Click(object sender, EventArgs e)

{

webBrowser1.GoHome();

}

private void webBrowser1\_DocumentCompleted(object sender, WebBrowserDocumentCompletedEventArgs e)

{

this.Text = webBrowser1.Document.Title + " - Charny Coding Browser";

toolStripTextBox1.Text = webBrowser1.Document.Url.ToString();

}

private void toolStripTextBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == 13)

{

webBrowser1.Navigate(toolStripTextBox1.Text);

}

}

}

}

**Editor.cs**

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;

namespace ManageRStudio

{

public partial class Editor : Form

{

public Editor()

{

InitializeComponent();

}

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

{

}

Image ImgUndo;

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

{

OpenFileDialog open = new OpenFileDialog();

open.Filter = "JPG Files|\*.jpg|BMP Files|\*.bmp|GIF Files|\*.gif|PNG Files|\*.png|JPEG Files|\*jpeg";

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

{

statusStrip1.Text = open.SafeFileName;

pictureBox1.ImageLocation = open.FileName;

}

}

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

{

SaveFileDialog save = new SaveFileDialog();

save.Filter = "JPG Files|\*.jpg|BMP Files|\*.bmp|GIF Files|\*.gif|PNG Files|\*.png|JPEG Files|\*jpeg";

Size size = new Size(pictureBox1.Width, pictureBox1.Height);

Bitmap bmp = new Bitmap(pictureBox1.Image, size);

save.FileName = statusStrip1.Text;

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

{

bmp.Save(save.FileName);

}

}

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

{

Close();

}

private void rotate90ToolStripMenuItem\_Click(object sender, EventArgs e)

{

pictureBox1.Image.RotateFlip(RotateFlipType.Rotate90FlipNone);

pictureBox1.Refresh();

}

private void rotate180ToolStripMenuItem\_Click(object sender, EventArgs e)

{

pictureBox1.Image.RotateFlip(RotateFlipType.Rotate180FlipNone);

pictureBox1.Refresh();

}

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

{

Size size = new Size(pictureBox1.Width, pictureBox1.Height);

Bitmap bmp = new Bitmap(pictureBox1.Image, size);

Color color;

ImgUndo = pictureBox1.Image;

undoToolStripMenuItem.Visible = true;

undoToolStripMenuItem.BackColor = Color.DarkCyan;

for (int height = 0; height < size.Height; ++height)

for (int width = 0; width < size.Width; ++width)

{

color = bmp.GetPixel(width, height);

int GrayScale = Convert.ToInt32(color.R \* 0.3 + color.G \* 0.59 + color.B \* 0.11);

bmp.SetPixel(width, height, Color.FromArgb(GrayScale, GrayScale, GrayScale));

}

pictureBox1.Image = bmp;

}

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

{

pictureBox1.Image = ImgUndo;

undoToolStripMenuItem.Visible = false;

}

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

{

Size size = new Size(pictureBox1.Width, pictureBox1.Height);

Bitmap bmp = new Bitmap(pictureBox1.Image, size);

Color color;

ImgUndo = pictureBox1.Image;

undoToolStripMenuItem.Visible = true;

undoToolStripMenuItem.BackColor = Color.DarkCyan;

for (int height = 0; height < size.Height; ++height)

for (int width = 0; width < size.Width; ++width)

{

color = bmp.GetPixel(width, height);

bmp.SetPixel(width, height, Color.FromArgb(255 - color.R, 255 - color.G, 255 - color.B));

}

pictureBox1.Image = bmp;

}

}

}

**Mp3Player.cs**

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;

namespace ManageRStudio

{

public partial class Mp3Player : Form

{

public Mp3Player()

{

InitializeComponent();

}

string[] muzica = new string[1000];

int ct;

int volum;

private void Mp3Player\_Load(object sender, EventArgs e)

{

trackBar1.Value = 100;

}

private void button4\_Click(object sender, EventArgs e)

{

OpenFileDialog open = new OpenFileDialog();

open.Multiselect = true;

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

{

for (int i = 0; i < open.FileNames.LongLength; ++i)

if (listBox1.Items.Contains(open.FileNames.GetValue(i)) == false)

{

listBox1.Items.Add(open.SafeFileNames.GetValue(i));

muzica[ct] = open.FileNames.GetValue(i).ToString();

++ct;

}

if (listBox1.SelectedIndex < 0) listBox1.SetSelected(0, true);

}

}

private void button1\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.URL = muzica[listBox1.SelectedIndex];

timer1.Start();

}

private void button9\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.play();

axWindowsMediaPlayer1.Ctlcontrols.currentPosition = Convert.ToDouble(label3.Text.Replace(":", ""));

button9.Visible = false;

button2.Visible = true;

}

private void button2\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.pause();

axWindowsMediaPlayer1.Ctlcontrols.currentPosition = Convert.ToDouble(label3.Text.Replace(":", ""));

button9.Visible = true;

button2.Visible = false;

}

private void button3\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.stop();

}

private void button10\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.fastForward();

}

private void button5\_Click(object sender, EventArgs e)

{

listBox1.SetSelected(0, true);

axWindowsMediaPlayer1.URL = muzica[listBox1.SelectedIndex];

}

private void button6\_Click(object sender, EventArgs e)

{

if (listBox1.SelectedIndex != 0)

{

listBox1.SetSelected(listBox1.SelectedIndex - 1, true);

axWindowsMediaPlayer1.URL = muzica[listBox1.SelectedIndex];

}

}

private void button8\_Click(object sender, EventArgs e)

{

if (listBox1.SelectedIndex != listBox1.Items.Count - 1)

{

listBox1.SetSelected(listBox1.SelectedIndex + 1, true);

axWindowsMediaPlayer1.URL = muzica[listBox1.SelectedIndex];

}

}

private void button7\_Click(object sender, EventArgs e)

{

listBox1.SetSelected(listBox1.Items.Count - 1, true);

axWindowsMediaPlayer1.URL = muzica[listBox1.SelectedIndex];

}

private void trackBar1\_Scroll(object sender, EventArgs e)

{

axWindowsMediaPlayer1.settings.volume = trackBar1.Value;

}

private void pictureBox2\_Click(object sender, EventArgs e)

{

trackBar1.Value = 0;

pictureBox1.Visible = true;

pictureBox2.Visible = false;

volum = axWindowsMediaPlayer1.settings.volume;

axWindowsMediaPlayer1.settings.volume = trackBar1.Value;

}

private void pictureBox1\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.settings.volume = volum;

trackBar1.Value = axWindowsMediaPlayer1.settings.volume;

pictureBox1.Visible = false;

pictureBox2.Visible = true;

}

private void timer1\_Tick(object sender, EventArgs e)

{

label4.Text = axWindowsMediaPlayer1.currentMedia.durationString;

label3.Text = axWindowsMediaPlayer1.Ctlcontrols.currentPositionString;

}

private void listBox1\_DoubleClick(object sender, EventArgs e)

{

axWindowsMediaPlayer1.URL = muzica[listBox1.SelectedIndex];

timer1.Start();

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void listBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

if(Keys.Space == ModifierKeys)

if (axWindowsMediaPlayer1.playState == WMPLib.WMPPlayState.wmppsPlaying)

{

axWindowsMediaPlayer1.Ctlcontrols.pause();

axWindowsMediaPlayer1.Ctlcontrols.currentPosition = Convert.ToDouble(label3.Text.Replace(":", ""));

button9.Visible = true;

button2.Visible = false;

}

else

{

axWindowsMediaPlayer1.Ctlcontrols.play();

axWindowsMediaPlayer1.Ctlcontrols.currentPosition = Convert.ToDouble(label3.Text.Replace(":", ""));

button9.Visible = false;

button2.Visible = true;

}

}

private void listBox1\_KeyDown(object sender, KeyEventArgs e)

{

if (Keys.Space == ModifierKeys)

if (axWindowsMediaPlayer1.playState == WMPLib.WMPPlayState.wmppsPlaying)

{

axWindowsMediaPlayer1.Ctlcontrols.pause();

axWindowsMediaPlayer1.Ctlcontrols.currentPosition = Convert.ToDouble(label3.Text.Replace(":", ""));

button9.Visible = true;

button2.Visible = false;

}

else

{

axWindowsMediaPlayer1.Ctlcontrols.play();

axWindowsMediaPlayer1.Ctlcontrols.currentPosition = Convert.ToDouble(label3.Text.Replace(":", ""));

button9.Visible = false;

button2.Visible = true;

}

}

}

}

**SlideShow.cs**

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;

namespace ManageRStudio

{

public partial class SlideShow : Form

{

int filemax = 0;

int filecount = 0;

public SlideShow()

{

InitializeComponent();

}

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

{

FolderBrowserDialog dlg = new FolderBrowserDialog();

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

{

filemax = 0;

panel1.Controls.Clear();

foreach (string a in System.IO.Directory.GetFiles(dlg.SelectedPath))

{

if ((a.EndsWith(".jpg")) || (a.EndsWith(".png")) || (a.EndsWith(".gif")) || (a.EndsWith(".bmp")) || (a.EndsWith(".jpeg")))

{

PictureBox pb = new PictureBox();

Image im = Image.FromFile(a);

pb.Image = im;

panel1.Controls.Add(pb);

filemax++;

}

}

timer1.Enabled = true;

timer1.Start();

timer1.Interval = Convert.ToInt32(textBox1.Text.ToString())\*1000;

timer1.Tick += new EventHandler(timer1\_Tick);

}

}

private void timer1\_Tick(object sender, EventArgs e)

{

if (filecount == filemax) { filecount = 0; }

PictureBox p = (PictureBox)panel1.Controls[filecount];

Size s = new Size(p.Image.Size.Width/3, p.Image.Size.Height/3);

pictureBox1.Image = p.Image;

Application.OpenForms["SlideShow"].Size = s;

filecount++;

timer1.Interval = Convert.ToInt32(textBox1.Text.ToString()) \* 1000;

}

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

{

}

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

{

}

}

}

**VideoPlayer.cs**

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;

namespace ManageRStudio

{

public partial class VideoPlayer : Form

{

public VideoPlayer()

{

InitializeComponent();

}

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

{

OpenFileDialog open = new OpenFileDialog();

open.Filter = "Mp3 Files|\*.mp3|Avi Files|\*.avi|Mpeg Files|\*.mpeg";

open.ShowDialog();

axWindowsMediaPlayer1.URL = open.FileName;

}

}

}

**IV. CONCLUZII**

Tehnologia se dezvoltă zi de zi și, în același timp, nevoile și dorințele oamenilor se diversifică. Timpul nu ne mai oferă aceleași limite la care, parcă, nu era posibil să ajungem și fiecare persoană își dorește rapiditate și eficiență.

Lucrând la această aplicație utilitară am conștientizat cu adevărat importanța aplicațiilor media (foto/ video) în rutina zilnică a fiecărui deținător de calculator. La început nu știam foarte multe despre C# / Windows Forms, însă, pe parcurs, am învățat să lucrez cu diferite interfețe de programare, astfel înțelegând mult mai bine codul. Realizez acest lucru atunci când mă uit la primele versiuni ale aplicației, care sunt extreme de sărace și cu mult mai multe erori în comparație cu varianta finală. Acest lucru mă încurajează să continui aprofundarea studiului creării aplicațiilor Windows și a limbajelor de programare diverse.

Programul a fost creat astfel încât informația să fie clară și concisă, iar grafica este foarte pretențioasă. Cu siguranță, orice utilizator poate folosi acest utilitar cu ușurință, indiferent de nivelul de cunoștințe TIC pe care le are.

Acest proiect reprezintă un bun mod de informare asupra importanței tehnologiei în viața fiecărui om. Pentru viitor aş vrea să adaug o secțiune dedicată dorințelor utilizatorilor. De asemenea, aș dori și să îi creez o platformă online pentru a reduce dificultatea căutării aplicației.

**V. BIBLIOGRAFIE**

* *C# .NET Web Developer’s Guide,* Turtschi Adrian, DotThatCom.com, Werry Jasson, Hack Greg, Albahari Joseph, Nandu Saurabh, Lee WiMeng; Syngress Publishing, 2002
* *Programming C#,*  Jesse Liberty; O’Reilly, 2001
* *Visual C#,* Lucian Sasu, 30.05.2005
* *Programare în Visual C# 2008 Express Edition,* Constantin Gălățan, Susana Gălățan; Editura L&S Info-Mat
* *Beginning C# 2008 From Novice to Professional Second Edition,* Christian Gross, Apress Publishing 2008
* Webografie:
  + <http://en.wikipedia.org/wiki/C_Sharp_(programming_language)>
  + <http://www.tutorialspoint.com/csharp/csharp_tutorial.pdf>
  + <http://www.filelist.ro>
  + <https://www.ghielectronics.com/downloads/FEZ/Beginners%20guide%20to%20NETMF.pdf>
  + <http://www.techotopia.com/index.php/C_Sharp_Essentials>

**VI. ANEXĂ CD/ DVD**