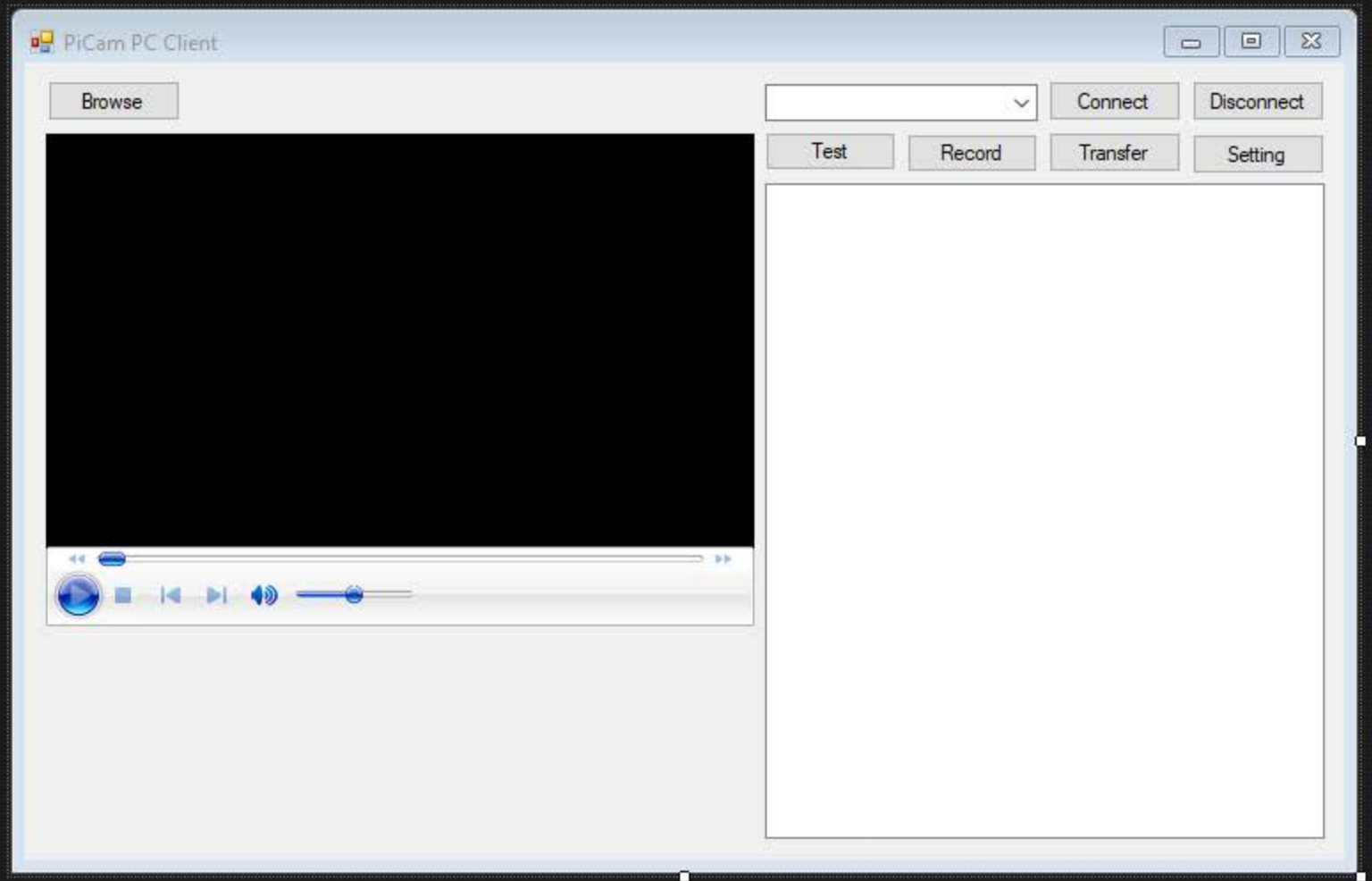

Status_Refresh_Timer

Device_List_Refresh_Timer

Record_Dialog

```
1  namespace PiCamClient
2  {
3      partial class Settings
4      {
5          /// <summary>
6          /// Required designer variable.
7          /// </summary>
8          private System.ComponentModel.IContainer components = null;
9
10         /// <summary>
11         /// Clean up any resources being used.
12         /// </summary>
13         /// <param name="disposing">true if managed resources should be disposed;
               otherwise, false.</param>
14         protected override void Dispose(bool disposing)
15         {
16             if (disposing && (components != null))
17             {
18                 components.Dispose();
19             }
20             base.Dispose(disposing);
21         }
22
23         #region Windows Form Designer generated code
24
25         /// <summary>
26         /// Required method for Designer support - do not modify
27         /// the contents of this method with the code editor.
28         /// </summary>
29         private void InitializeComponent()
30         {
31             this.components = new System.ComponentModel.Container();
32             this.text_name = new System.Windows.Forms.TextBox();
33             this.text_hostname = new System.Windows.Forms.TextBox();
34             this.text_username = new System.Windows.Forms.TextBox();
35             this.text_password = new System.Windows.Forms.TextBox();
36             this.label_name = new System.Windows.Forms.Label();
37             this.label_host = new System.Windows.Forms.Label();
38             this.label_username = new System.Windows.Forms.Label();
39             this.label_password = new System.Windows.Forms.Label();
40             this.Device_Select = new System.Windows.Forms.ComboBox();
41             this.label_Device = new System.Windows.Forms.Label();
42             this.button_save = new System.Windows.Forms.Button();
43             this.button_exit = new System.Windows.Forms.Button();
44             this.Check_Visibility_Timer = new System.Windows.Forms.Timer
                   (this.components);
45             this.SuspendLayout();
46             //
47             // text_name
48             //
49             this.text_name.Location = new System.Drawing.Point(80, 49);
50             this.text_name.Name = "text_name";
```

```
51              this.text_name.Size = new System.Drawing.Size(100, 20);
52              this.text_name.TabIndex = 0;
53              this.text_name.TextChanged += new System.EventHandler
                  (this.text_name_TextChanged);
54              //
55              // text_hostname
56              //
57              this.text_hostname.Location = new System.Drawing.Point(80, 75);
58              this.text_hostname.Name = "text_hostname";
59              this.text_hostname.Size = new System.Drawing.Size(100, 20);
60              this.text_hostname.TabIndex = 1;
61              this.text_hostname.TextChanged += new System.EventHandler
                  (this.text_hostname_TextChanged);
62              //
63              // text_username
64              //
65              this.text_username.Location = new System.Drawing.Point(80, 101);
66              this.text_username.Name = "text_username";
67              this.text_username.Size = new System.Drawing.Size(100, 20);
68              this.text_username.TabIndex = 2;
69              this.text_username.TextChanged += new System.EventHandler
                  (this.text_username_TextChanged);
70              //
71              // text_password
72              //
73              this.text_password.Location = new System.Drawing.Point(80, 127);
74              this.text_password.Name = "text_password";
75              this.text_password.Size = new System.Drawing.Size(100, 20);
76              this.text_password.TabIndex = 3;
77              this.text_password.TextChanged += new System.EventHandler
                  (this.text_password_TextChanged);
78              //
79              // label_name
80              //
81              this.label_name.AutoSize = true;
82              this.label_name.Location = new System.Drawing.Point(36, 52);
83              this.label_name.Name = "label_name";
84              this.label_name.Size = new System.Drawing.Size(38, 13);
85              this.label_name.TabIndex = 5;
86              this.label_name.Text = "Name:";
87              //
88              // label_host
89              //
90              this.label_host.AutoSize = true;
91              this.label_host.Location = new System.Drawing.Point(14, 78);
92              this.label_host.Name = "label_host";
93              this.label_host.Size = new System.Drawing.Size(60, 13);
94              this.label_host.TabIndex = 6;
95              this.label_host.Text = "HostName:";
96              //
97              // label_username
98              //
```

```
 99              this.label_username.AutoSize = true;
100              this.label_username.Location = new System.Drawing.Point(14, 104);
101              this.label_username.Name = "label_username";
102              this.label_username.Size = new System.Drawing.Size(60, 13);
103              this.label_username.TabIndex = 7;
104              this.label_username.Text = "UserName:";
105              //
106              // label_password
107              //
108              this.label_password.AutoSize = true;
109              this.label_password.Location = new System.Drawing.Point(18, 130);
110              this.label_password.Name = "label_password";
111              this.label_password.Size = new System.Drawing.Size(56, 13);
112              this.label_password.TabIndex = 8;
113              this.label_password.Text = "Password:";
114              //
115              // Device_Select
116              //
117              this.Device_Select.FormattingEnabled = true;
118              this.Device_Select.Location = new System.Drawing.Point(80, 16);
119              this.Device_Select.Margin = new System.Windows.Forms.Padding(2);
120              this.Device_Select.Name = "Device_Select";
121              this.Device_Select.Size = new System.Drawing.Size(100, 21);
122              this.Device_Select.TabIndex = 10;
123              this.Device_Select.SelectedIndexChanged += new System.EventHandler ↵
                   (this.Device_Select_SelectedIndexChanged);
124              //
125              // label_Device
126              //
127              this.label_Device.AutoSize = true;
128              this.label_Device.Location = new System.Drawing.Point(0, 19);
129              this.label_Device.Name = "label_Device";
130              this.label_Device.Size = new System.Drawing.Size(74, 13);
131              this.label_Device.TabIndex = 11;
132              this.label_Device.Text = "Select Device";
133              //
134              // button_save
135              //
136              this.button_save.Enabled = false;
137              this.button_save.Location = new System.Drawing.Point(3, 157);
138              this.button_save.Name = "button_save";
139              this.button_save.Size = new System.Drawing.Size(86, 23);
140              this.button_save.TabIndex = 12;
141              this.button_save.Text = "Save Changes";
142              this.button_save.UseVisualStyleBackColor = true;
143              this.button_save.Click += new System.EventHandler ↵
                   (this.button_save_Click);
144              //
145              // button_exit
146              //
147              this.button_exit.Location = new System.Drawing.Point(95, 157);
148              this.button_exit.Name = "button_exit";
```

```
149             this.button_exit.Size = new System.Drawing.Size(85, 23);
150             this.button_exit.TabIndex = 13;
151             this.button_exit.Text = "Save and Exit";
152             this.button_exit.UseVisualStyleBackColor = true;
153             this.button_exit.Click += new System.EventHandler
                  (this.button_exit_Click);
154             //
155             // Check_Visibility_Timer
156             //
157             this.Check_Visibility_Timer.Enabled = true;
158             this.Check_Visibility_Timer.Interval = 500;
159             this.Check_Visibility_Timer.Tick += new System.EventHandler
                  (this.Check_Visibility_Timer_Tick);
160             //
161             // Settings
162             //
163             this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
164             this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
165             this.ClientSize = new System.Drawing.Size(190, 192);
166             this.ControlBox = false;
167             this.Controls.Add(this.button_exit);
168             this.Controls.Add(this.button_save);
169             this.Controls.Add(this.label_Device);
170             this.Controls.Add(this.Device_Select);
171             this.Controls.Add(this.label_password);
172             this.Controls.Add(this.label_username);
173             this.Controls.Add(this.label_host);
174             this.Controls.Add(this.label_name);
175             this.Controls.Add(this.text_password);
176             this.Controls.Add(this.text_username);
177             this.Controls.Add(this.text_hostname);
178             this.Controls.Add(this.text_name);
179             this.FormBorderStyle =
                  System.Windows.Forms.FormBorderStyle.FixedToolWindow;
180             this.Name = "Settings";
181             this.StartPosition =
                  System.Windows.Forms.FormStartPosition.CenterScreen;
182             this.Text = "Settings";
183             this.TopMost = true;
184             this.ResumeLayout(false);
185             this.PerformLayout();
186 
187         }
188 
189         #endregion
190 
191         private System.Windows.Forms.TextBox text_name;
192         private System.Windows.Forms.TextBox text_hostname;
193         private System.Windows.Forms.TextBox text_username;
194         private System.Windows.Forms.TextBox text_password;
195         private System.Windows.Forms.Label label_name;
196         private System.Windows.Forms.Label label_host;
```

```
197         private System.Windows.Forms.Label label_username;
198         private System.Windows.Forms.Label label_password;
199         private System.Windows.Forms.ComboBox Device_Select;
200         private System.Windows.Forms.Label label_Device;
201         private System.Windows.Forms.Button button_save;
202         private System.Windows.Forms.Button button_exit;
203         private System.Windows.Forms.Timer Check_Visibility_Timer;
204     }
205 }
```

```
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Drawing;
6  using System.Linq;
7  using System.Text;
8  using System.Resources;
9  using System.IO;
10 using System.Threading.Tasks;
11 using System.Windows.Forms;
12 using WinSCP;
13 using WMPLib;
14
15 using PiCamClient.Properties;
16
17 namespace PiCamClient
18 {
19     public partial class Main : Form
20     {
21         const int Device_Count = 10;
22         public SessionOptions[] device_list = new SessionOptions[Device_Count];
23
24         // Transfer Progress List Groups
25         ListViewGroup Initiation_Group = new ListViewGroup("Live Monitor");
26         ListViewGroup Transfer_Group = new ListViewGroup("File Transfer");
27
28         string[] File_List = new string[50];
29         bool file_list_state = false;
30
31         // Winscp Result Declaration
32         CommandExecutionResult execute_result;
33         TransferOptions sftp_option = new TransferOptions();
34         TransferOperationResult sftp_result;
35         Session ssh_session = new Session();
36
37         // Background Stop Flag
38         bool status_check_stop_flag = false;
39
40         //
41         string status = "off";
42         string action = "none";
43         bool recording_status = false;
44         int PiCam_Selection = -1;
45         bool transfer_status = false;
46
47         Settings settings = new Settings();
48
49         public Main()
50         {
51             InitializeComponent();
52         }
```

```
53
54          private void Button_Transfer_Click(object sender, EventArgs e)
55          {
56              action = "transfer";
57          }
58
59          private void SessionFileTransferProgress(object sender ,
              FileTransferProgressEventArgs e)
60          {
61              string _lastFileName = "";
62              int currentindex = 0;
63              // Grab file name
64              string[] filename_array = e.FileName.Split('/');
65
66              // New line for every new file
67              if (_lastFileName != e.FileName)
68              {
69                  foreach(ListViewItem theitem in Transfer_Progress_List.Items)
70                  {
71                      if (theitem.SubItems[0].Text.Split('(')[0] == filename_array
                        [filename_array.Length - 1])
72                      {
73                          currentindex = theitem.Index;
74                          Transfer_Progress_List.Items[currentindex].EnsureVisible
                        ();
75                      }
76                  }
77              }
78
79              // Print transfer progress
80              double progess_percent =  e.FileProgress * 100;
81              // Use "Done" instead of 100%
82              if (progess_percent == 100) Transfer_Progress_List.Items
                  [currentindex].SubItems[1].Text = "Done";
83              else Transfer_Progress_List.Items[currentindex].SubItems[1].Text =
                  progess_percent + "%";
84
85              // Remember a name of the last file reported
86              _lastFileName = e.FileName;
87          }
88
89          private void Main_Load(object sender, EventArgs e)
90          {
91              //Player_1.URL = @"D:\Project49\2019-01-15_16-19-33.ts";
92              //Player_1.Ctlcontrols.play();
93
94              // Media Player Initiation
95              Player_1.Ctlenabled = false;
96              Player_1.uiMode = "none";
97
98              // Transfer Progress List Settings
99              Transfer_Progress_List.Groups.Add(Initiation_Group);
```

```
100             Transfer_Progress_List.Groups.Add(Transfer_Group);
101             Transfer_Progress_List.Sorting = SortOrder.None;
102             Transfer_Progress_List.View = View.Details;
103             Transfer_Progress_List.AllowColumnReorder = false;
104             Transfer_Progress_List.FullRowSelect = true;
105             Transfer_Progress_List.Columns.Add(new ColumnHeader());
106             Transfer_Progress_List.Columns[0].Text = "Name";
107             Transfer_Progress_List.Columns[0].Width = 170;
108             Transfer_Progress_List.Columns.Add(new ColumnHeader());
109             Transfer_Progress_List.Columns[1].Text = "Progress";
110             Transfer_Progress_List.Columns[1].Width = 75;
111             Transfer_Progress_List.Items.Add(new ListViewItem(new string[]
                  { "Remote Device Access", "Unavailable" }, Initiation_Group));
112             Transfer_Progress_List.Items.Add(new ListViewItem(new string[]
                  { "File Tranfer Stream", "Stopped" }, Initiation_Group));
113             Transfer_Progress_List.Items.Add(new ListViewItem(new string[]
                  { "Recording Status", "Stopped" }, Initiation_Group));
114
115             sftp_option.TransferMode = TransferMode.Binary;
116             Device_List_Refresh_Timer.Enabled = true;
117         }
118
119         private void Main_Shown(object sender, EventArgs e)
120         {
121             //Player_1.URL = @"D:\Project49\PiCam_LOGO.png";
122             //Player_1.URL = PiCamClient.Properties.Resources.PiCam_LOGO.png;
123             Player_1.Ctlcontrols.pause();
124             Device_List_Refresh_Timer.Enabled = true;
125         }
126
127         private async void Button_Connect_Click(object sender, EventArgs e)
128         {
129             status_check_stop_flag = false;
130             await Task.Run((Action)status_check_function);
131         }
132
133
134         private void Button_Disconnect_Click(object sender, EventArgs e)
135         {
136             status_check_stop_flag = true;
137         }
138
139         private void status_check_function()
140         {
141             status = "ing";
142             try
143             {
144                 //
                      ============================================================
                      ==================================================
145                 // Select and Establish Connection
146                 ssh_session.Open(device_list[PiCam_Selection]);
```

```
147
148                 status = "on";
149
150                 while (status_check_stop_flag == false)
151                 {
152                     switch(action)
153                     {
154                         //
                        ============================================================
                        ============================================================
155                         // Transfer Recording
156                         case "transfer":
157                             transfer_status = true;
158                             // Read
159
160                             RemoteDirectoryInfo directory =
                        ssh_session.ListDirectory("/home/pi/picam/rec/");
161                             for(int file_list_count = 0; file_list_count <
                        File_List.Length; file_list_count++)
162                             {
163                                 File_List[file_list_count] = "";
164                             }
165                             int file_count = 0;
166                             foreach (RemoteFileInfo FileInfo in directory.Files)
167                             {
168                                 // Grab file name
169                                 string[] filename_array = FileInfo.Name.Split
                        ('.');
170
171                                 // File Type Matching
172                                 if (filename_array[filename_array.Length -
                        1].ToLower() == "ts") // change to TS for PiCam!!!!!!!!!
173                                 {
174                                     int i = 0;
175                                     string[] Byte_Units = { " B", " KB", " MB", "
                         GB", "TB" };
176                                     long FileSize = FileInfo.Length;
177                                     while (FileSize >= 1024)
178                                     {
179                                         FileSize = FileSize / 1024;
180                                         i++;
181                                     }
182                                     string filename = FileInfo.Name + "(" +
                        FileSize + Byte_Units[i] + ")";
183                                     //File_List[file_count] = new string;
184                                     File_List[file_count] = filename;
185                                     file_count++;
186                                 }
187                             }
188                             file_list_state = true;
189                             while (file_list_state) ;
190
```

```
191
192                         // Transfer Files
193                         //sftp_result = ssh_session.PutFiles(@"D:\Downloads
                    \Bla Bla Bla\Pokemon\*", "/home/pi/Pictures/", false,
                    sftp_option);
194                         sftp_result = ssh_session.GetFiles("/home/pi/picam/
                    rec/*.ts", @"D:\Project49\", false, sftp_option);
195                         transfer_status = false;
196                         //sftp_result.Check();
197
198                         MessageBox.Show("Transfer Done", "Notification",
                    MessageBoxButtons.OK);
199                         action = "none";
200                         break;
201                     case "record":
202                         execute_result = ssh_session.ExecuteCommand("sudo
                    touch ~/picam/hooks/start_record");
203                         action = "none";
204                         break;
205                     case "stop":
206                         execute_result = ssh_session.ExecuteCommand("sudo
                    touch ~/picam/hooks/stop_record");
207                         action = "none";
208                         break;
209                     case "test":
210                         execute_result = ssh_session.ExecuteCommand("sudo
                    sh /home/pi/Project49/Code/fivesec.sh");
211                         MessageBox.Show("Test Performed", "Notification",
                    MessageBoxButtons.OK);
212                         action = "none";
213                         break;
214                     //
                    ============================================================
                    ==========================================
215                     case "none":
216                         break;
217                 }
218
219                 //
                    ============================================================
                    ==========================================
220                 // Check Recording Status
221                 execute_result = ssh_session.ExecuteCommand("sudo cat ~/
                    picam/state/record");
222                 if (execute_result.Output == "false") recording_status =
                    false;
223                 else if (execute_result.Output == "true") recording_status =
                    true;
224             }
225             ssh_session.Close();
226             status = "off";
227
```

```
228             }
229             catch (Exception error)
230             {
231                 MessageBox.Show(error.Message, "Error Message",
                      MessageBoxButtons.OK);
232             }
233         }
234 
235         public void Status_Refresh_Timer_Tick(object sender, EventArgs e)
236         {
237             switch(status)
238             {
239                 case "off":
240                     Button_Connect.Enabled = true;
241                     Button_Disconnect.Enabled = false;
242                     Transfer_Progress_List.Items[0].SubItems[1].Text =
                         "Unavailable";
243                     break;
244                 case "ing":
245                     Button_Connect.Enabled = false;
246                     Button_Disconnect.Enabled = false;
247                     Transfer_Progress_List.Items[0].SubItems[1].Text =
                         "Establishing";
248                     break;
249                 case "on":
250                     Button_Connect.Enabled = false;
251                     Button_Disconnect.Enabled = true;
252                     Transfer_Progress_List.Items[0].SubItems[1].Text =
                         "Connected";
253                     break;
254             }
255             if (recording_status)
256             {
257                 if (action == "stop") Button_Record.Enabled = false;
258                 else Button_Record.Enabled = true;
259                 Transfer_Progress_List.Items[2].SubItems[1].Text = "Recording";
260                 Button_Record.Text = "Stop";
261             }
262             else
263             {
264                 if (action == "record") Button_Record.Enabled = false;
265                 else Button_Record.Enabled = true;
266                 Transfer_Progress_List.Items[2].SubItems[1].Text = "Stopped";
267                 Button_Record.Text = "Record";
268             }
269             if(transfer_status)
270             {
271                 Button_Transfer.Enabled = false;
272                 Transfer_Progress_List.Items[1].SubItems[1].Text = "In Progress";
273             }
274             else
275             {
```

```
276                 Button_Transfer.Enabled = true;
277                 Transfer_Progress_List.Items[1].SubItems[1].Text = "Down";
278             }
279             if(file_list_state)
280             {
281                 foreach(ListViewItem item in Transfer_Progress_List.Groups
                      [1].Items)
282                 {
283                     item.Remove();
284                 }
285                 foreach(string filename in File_List)
286                 {
287                     if (!(filename == ""))
288                     {
289                         Transfer_Progress_List.Items.Add(new ListViewItem(new
                        string[] { filename, "Waiting" }, Transfer_Group));
290                         Transfer_Progress_List.Items
                        [Transfer_Progress_List.Items.Count - 1].EnsureVisible();
291                     }
292                 }
293                 file_list_state = false;
294             }
295         }
296
297         private void PiCam_List_SelectedIndexChanged(object sender, EventArgs e)
298         {
299             PiCam_Selection = PiCam_List.SelectedIndex;
300         }
301
302         private void Button_Record_Click(object sender, EventArgs e)
303         {
304             if(recording_status) action = "stop";
305             else action = "record";
306         }
307
308         private void Button_Test_Click(object sender, EventArgs e)
309         {
310             action = "test";
311         }
312
313         public void Button_Setting_Click(object sender, EventArgs e)
314         {
315             settings.Show();
316             Device_List_Refresh_Timer.Enabled = true;
317
318         }
319
320         public void Device_List_Refresh_Timer_Tick(object sender, EventArgs e)
321         {
322             if (!settings.Visible)
323             {
324                 if (File.Exists(@"config.txt"))
```

```
325                 {
326                     try
327                     {
328                         StreamReader reader = new StreamReader(@"config.txt");
329                         for (int i = 0; i < Device_Count; i++)
330                         {
331                             device_list[i] = new SessionOptions();
332                             device_list[i].HostName = "";
333                             device_list[i].UserName = "";
334                             device_list[i].Password = "";
335                         }
336                         PiCam_List.Items.Clear();
337                         int count = 0;
338                         while (true)
339                         {
340                             if (string.Equals(reader.ReadLine(), "----"))
341                             {
342                                 PiCam_List.Items.Add(reader.ReadLine());
343                                 device_list[count].HostName = reader.ReadLine();
344                                 device_list[count].UserName = reader.ReadLine();
345                                 device_list[count].Password = reader.ReadLine();
346                                 device_list[count].Protocol = Protocol.Sftp;
347                                 device_list
                        [count].GiveUpSecurityAndAcceptAnySshHostKey = true;
348                                 count++;
349                             }
350                             else
351                             {
352                                 break;
353                             }
354                         }
355                         reader.Close();
356                     }
357                     catch (Exception wrong)
358                     {
359                         MessageBox.Show(wrong.Message, "Error Message",
                         MessageBoxButtons.OK);
360                     }
361                 }
362                 else
363                 {
364                     MessageBox.Show("Possible setting file missing or broken!
                     Please restore or create new.", "Setting File Missing",
                     MessageBoxButtons.OK);
365                 }
366                 if (PiCam_List.SelectedIndex == -1 && !(PiCam_List.Items.Count ==
                    0))
367                 {
368                     PiCam_List.SelectedIndex = 0;
369                 }
370                 Device_List_Refresh_Timer.Enabled = false;
371             }
```

```
372         }
373
374         private void Button_Browse_Click(object sender, EventArgs e)
375         {
376             Record_Dialog.Filter = "TS Files (*.ts)|*.ts|All Files (*.*)|*.*";
377             Record_Dialog.CheckFileExists = true;
378             Record_Dialog.Multiselect = false;
379             Player_1.settings.mute = false;
380             Player_1.settings.setMode("loop", true);
381             //Player_1.settings.volume = 12;
382             if (Record_Dialog.ShowDialog() != DialogResult.OK) return;
383             Player_1.URL = Record_Dialog.FileName;
384             Player_1.Ctlcontrols.play();
385         }
386     }
387 }
```

Check_Visibility_Timer

```
1  namespace PiCamClient
2  {
3      partial class Settings
4      {
5          /// <summary>
6          /// Required designer variable.
7          /// </summary>
8          private System.ComponentModel.IContainer components = null;
9
10         /// <summary>
11         /// Clean up any resources being used.
12         /// </summary>
13         /// <param name="disposing">true if managed resources should be disposed;
               otherwise, false.</param>
14         protected override void Dispose(bool disposing)
15         {
16             if (disposing && (components != null))
17             {
18                 components.Dispose();
19             }
20             base.Dispose(disposing);
21         }
22
23         #region Windows Form Designer generated code
24
25         /// <summary>
26         /// Required method for Designer support - do not modify
27         /// the contents of this method with the code editor.
28         /// </summary>
29         private void InitializeComponent()
30         {
31             this.components = new System.ComponentModel.Container();
32             this.text_name = new System.Windows.Forms.TextBox();
33             this.text_hostname = new System.Windows.Forms.TextBox();
34             this.text_username = new System.Windows.Forms.TextBox();
35             this.text_password = new System.Windows.Forms.TextBox();
36             this.label_name = new System.Windows.Forms.Label();
37             this.label_host = new System.Windows.Forms.Label();
38             this.label_username = new System.Windows.Forms.Label();
39             this.label_password = new System.Windows.Forms.Label();
40             this.Device_Select = new System.Windows.Forms.ComboBox();
41             this.label_Device = new System.Windows.Forms.Label();
42             this.button_save = new System.Windows.Forms.Button();
43             this.button_exit = new System.Windows.Forms.Button();
44             this.Check_Visibility_Timer = new System.Windows.Forms.Timer
                   (this.components);
45             this.SuspendLayout();
46             //
47             // text_name
48             //
49             this.text_name.Location = new System.Drawing.Point(80, 49);
50             this.text_name.Name = "text_name";
```

```
51              this.text_name.Size = new System.Drawing.Size(100, 20);
52              this.text_name.TabIndex = 0;
53              this.text_name.TextChanged += new System.EventHandler
                  (this.text_name_TextChanged);
54              //
55              // text_hostname
56              //
57              this.text_hostname.Location = new System.Drawing.Point(80, 75);
58              this.text_hostname.Name = "text_hostname";
59              this.text_hostname.Size = new System.Drawing.Size(100, 20);
60              this.text_hostname.TabIndex = 1;
61              this.text_hostname.TextChanged += new System.EventHandler
                  (this.text_hostname_TextChanged);
62              //
63              // text_username
64              //
65              this.text_username.Location = new System.Drawing.Point(80, 101);
66              this.text_username.Name = "text_username";
67              this.text_username.Size = new System.Drawing.Size(100, 20);
68              this.text_username.TabIndex = 2;
69              this.text_username.TextChanged += new System.EventHandler
                  (this.text_username_TextChanged);
70              //
71              // text_password
72              //
73              this.text_password.Location = new System.Drawing.Point(80, 127);
74              this.text_password.Name = "text_password";
75              this.text_password.Size = new System.Drawing.Size(100, 20);
76              this.text_password.TabIndex = 3;
77              this.text_password.TextChanged += new System.EventHandler
                  (this.text_password_TextChanged);
78              //
79              // label_name
80              //
81              this.label_name.AutoSize = true;
82              this.label_name.Location = new System.Drawing.Point(36, 52);
83              this.label_name.Name = "label_name";
84              this.label_name.Size = new System.Drawing.Size(38, 13);
85              this.label_name.TabIndex = 5;
86              this.label_name.Text = "Name:";
87              //
88              // label_host
89              //
90              this.label_host.AutoSize = true;
91              this.label_host.Location = new System.Drawing.Point(14, 78);
92              this.label_host.Name = "label_host";
93              this.label_host.Size = new System.Drawing.Size(60, 13);
94              this.label_host.TabIndex = 6;
95              this.label_host.Text = "HostName:";
96              //
97              // label_username
98              //
```

```
 99              this.label_username.AutoSize = true;
100              this.label_username.Location = new System.Drawing.Point(14, 104);
101              this.label_username.Name = "label_username";
102              this.label_username.Size = new System.Drawing.Size(60, 13);
103              this.label_username.TabIndex = 7;
104              this.label_username.Text = "UserName:";
105              //
106              // label_password
107              //
108              this.label_password.AutoSize = true;
109              this.label_password.Location = new System.Drawing.Point(18, 130);
110              this.label_password.Name = "label_password";
111              this.label_password.Size = new System.Drawing.Size(56, 13);
112              this.label_password.TabIndex = 8;
113              this.label_password.Text = "Password:";
114              //
115              // Device_Select
116              //
117              this.Device_Select.FormattingEnabled = true;
118              this.Device_Select.Location = new System.Drawing.Point(80, 16);
119              this.Device_Select.Margin = new System.Windows.Forms.Padding(2);
120              this.Device_Select.Name = "Device_Select";
121              this.Device_Select.Size = new System.Drawing.Size(100, 21);
122              this.Device_Select.TabIndex = 10;
123              this.Device_Select.SelectedIndexChanged += new System.EventHandler ↵
                   (this.Device_Select_SelectedIndexChanged);
124              //
125              // label_Device
126              //
127              this.label_Device.AutoSize = true;
128              this.label_Device.Location = new System.Drawing.Point(0, 19);
129              this.label_Device.Name = "label_Device";
130              this.label_Device.Size = new System.Drawing.Size(74, 13);
131              this.label_Device.TabIndex = 11;
132              this.label_Device.Text = "Select Device";
133              //
134              // button_save
135              //
136              this.button_save.Enabled = false;
137              this.button_save.Location = new System.Drawing.Point(3, 157);
138              this.button_save.Name = "button_save";
139              this.button_save.Size = new System.Drawing.Size(86, 23);
140              this.button_save.TabIndex = 12;
141              this.button_save.Text = "Save Changes";
142              this.button_save.UseVisualStyleBackColor = true;
143              this.button_save.Click += new System.EventHandler ↵
                   (this.button_save_Click);
144              //
145              // button_exit
146              //
147              this.button_exit.Location = new System.Drawing.Point(95, 157);
148              this.button_exit.Name = "button_exit";
```

```
149             this.button_exit.Size = new System.Drawing.Size(85, 23);
150             this.button_exit.TabIndex = 13;
151             this.button_exit.Text = "Save and Exit";
152             this.button_exit.UseVisualStyleBackColor = true;
153             this.button_exit.Click += new System.EventHandler
                  (this.button_exit_Click);
154             //
155             // Check_Visibility_Timer
156             //
157             this.Check_Visibility_Timer.Enabled = true;
158             this.Check_Visibility_Timer.Interval = 500;
159             this.Check_Visibility_Timer.Tick += new System.EventHandler
                  (this.Check_Visibility_Timer_Tick);
160             //
161             // Settings
162             //
163             this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
164             this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
165             this.ClientSize = new System.Drawing.Size(190, 192);
166             this.ControlBox = false;
167             this.Controls.Add(this.button_exit);
168             this.Controls.Add(this.button_save);
169             this.Controls.Add(this.label_Device);
170             this.Controls.Add(this.Device_Select);
171             this.Controls.Add(this.label_password);
172             this.Controls.Add(this.label_username);
173             this.Controls.Add(this.label_host);
174             this.Controls.Add(this.label_name);
175             this.Controls.Add(this.text_password);
176             this.Controls.Add(this.text_username);
177             this.Controls.Add(this.text_hostname);
178             this.Controls.Add(this.text_name);
179             this.FormBorderStyle =
                  System.Windows.Forms.FormBorderStyle.FixedToolWindow;
180             this.Name = "Settings";
181             this.StartPosition =
                  System.Windows.Forms.FormStartPosition.CenterScreen;
182             this.Text = "Settings";
183             this.TopMost = true;
184             this.ResumeLayout(false);
185             this.PerformLayout();
186 
187         }
188 
189         #endregion
190 
191         private System.Windows.Forms.TextBox text_name;
192         private System.Windows.Forms.TextBox text_hostname;
193         private System.Windows.Forms.TextBox text_username;
194         private System.Windows.Forms.TextBox text_password;
195         private System.Windows.Forms.Label label_name;
196         private System.Windows.Forms.Label label_host;
```

```
197         private System.Windows.Forms.Label label_username;
198         private System.Windows.Forms.Label label_password;
199         private System.Windows.Forms.ComboBox Device_Select;
200         private System.Windows.Forms.Label label_Device;
201         private System.Windows.Forms.Button button_save;
202         private System.Windows.Forms.Button button_exit;
203         private System.Windows.Forms.Timer Check_Visibility_Timer;
204     }
205 }
```

```
 1  using System;
 2  using System.IO;
 3  using System.Collections.Generic;
 4  using System.ComponentModel;
 5  using System.Data;
 6  using System.Drawing;
 7  using System.Linq;
 8  using System.Text;
 9  using System.Threading.Tasks;
10  using System.Windows.Forms;
11  using WinSCP;
12
13  namespace PiCamClient
14  {
15      public partial class Settings : Form
16      {
17
18          string[,] device_list = new string[10, 4];
19          int i = 0;
20          int j = 0;
21          int last_index = 0;
22          bool last_visible = false;
23
24          public Settings()
25          {
26              InitializeComponent();
27          }
28
29          private void Device_Select_SelectedIndexChanged(object sender, EventArgs
              e)
30          {
31              if(button_save.Enabled)
32              {
33                  if(!(Device_Select.SelectedIndex == last_index))
34                  {
35                      if (MessageBox.Show("Do you want to discard the changes you
                        made?", "Change Unsaved", MessageBoxButtons.YesNo) ==
                        DialogResult.No)
36                      {
37                          Device_Select.SelectedIndex = last_index;
38                      }
39                      else
40                      {
41                          last_index = Device_Select.SelectedIndex;
42                          text_name.Text = device_list[Device_Select.SelectedIndex,
                         0];
43                          text_hostname.Text = device_list
                        [Device_Select.SelectedIndex, 1];
44                          text_username.Text = device_list
                        [Device_Select.SelectedIndex, 2];
45                          text_password.Text = device_list
                        [Device_Select.SelectedIndex, 3];
```

```
46                          button_save.Enabled = false;
47                      }
48                  }
49              }
50              else
51              {
52                  last_index = Device_Select.SelectedIndex;
53                  text_name.Text = device_list[Device_Select.SelectedIndex, 0];
54                  text_hostname.Text = device_list[Device_Select.SelectedIndex, 1];
55                  text_username.Text = device_list[Device_Select.SelectedIndex, 2];
56                  text_password.Text = device_list[Device_Select.SelectedIndex, 3];
57                  button_save.Enabled = false;
58              }
59          }
60
61          private void text_name_TextChanged(object sender, EventArgs e)
62          {
63              button_save.Enabled = true;
64          }
65
66          private void text_hostname_TextChanged(object sender, EventArgs e)
67          {
68              button_save.Enabled = true;
69          }
70
71          private void text_username_TextChanged(object sender, EventArgs e)
72          {
73              button_save.Enabled = true;
74          }
75
76          private void text_password_TextChanged(object sender, EventArgs e)
77          {
78              button_save.Enabled = true;
79          }
80
81          private void text_fingerprint_TextChanged(object sender, EventArgs e)
82          {
83              button_save.Enabled = true;
84          }
85
86          private void button_save_Click(object sender, EventArgs e)
87          {
88              device_list[Device_Select.SelectedIndex, 0] = text_name.Text;
89              device_list[Device_Select.SelectedIndex, 1] = text_hostname.Text;
90              device_list[Device_Select.SelectedIndex, 2] = text_username.Text;
91              device_list[Device_Select.SelectedIndex, 3] = text_password.Text;
92              button_save.Enabled = false;
93          }
94
95          private void button_exit_Click(object sender, EventArgs e)
96          {
97              StreamWriter writer = new StreamWriter(@"config.txt");
```

```
 98              for (j = 0; j < 10; j++)
 99              {
100                  if(device_list[j, 0] != null)
101                  {
102                      writer.WriteLine("----");
103                      for (i = 0; i < 4; i++)
104                      {
105                          writer.WriteLine(device_list[j, i]);
106                      }
107                  }
108              }
109              writer.Close();
110              last_visible = false;
111              this.Hide();
112          }
113
114          private void Check_Visibility_Timer_Tick(object sender, EventArgs e)
115          {
116              if(this.Visible && !last_visible)
117              {
118                  last_visible = true;
119                  if (File.Exists(@"config.txt"))
120                  {
121                      try
122                      {
123                          StreamReader reader = new StreamReader(@"config.txt");
124                          i = 0;
125                          Device_Select.Items.Clear();
126                          while (true)
127                          {
128                              if (string.Equals(reader.ReadLine(), "----"))
129                              {
130                                  device_list[i, 0] = reader.ReadLine();
131                                  Device_Select.Items.Add(device_list[i, 0]);
132                                  device_list[i, 1] = reader.ReadLine();
133                                  device_list[i, 2] = reader.ReadLine();
134                                  device_list[i, 3] = reader.ReadLine();
135                                  i++;
136                              }
137                              else
138                              {
139                                  break;
140                              }
141                          }
142                          reader.Close();
143                      }
144                      catch (Exception haha)
145                      {
146                          MessageBox.Show(haha.Message, "Error Message",
                         MessageBoxButtons.OK);
147                      }
148                  }
```

```
149                 else
150                 {
151                     MessageBox.Show("Possible setting file missing or broken!
                        Please restore or create new.", "Setting File Missing",
                        MessageBoxButtons.OK);
152                 }
153                 if (Device_Select.SelectedIndex == -1 && !
                      (Device_Select.Items.Count == 0))
154                 {
155                     Device_Select.SelectedIndex = 0;
156                 }
157             }
158         }
159     }
160 }
161
```