

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Drawing;
using System.Data;
using System.IO;
using System.Windows.Forms;
using System.Drawing.Drawing2D;
using SB54_CSAI.Genres;
using SB54_CSAI.MP3Lib;
using System.Text.RegularExpressions;
using SB54_CSAI.DatabaseAccess;

namespace SB54_CSAI.MP3Editor
{
    #region uctMP3Editor CLASS
    /// <summary>
    /// uctCDRipper is a sub class of a <see cref="System.Windows.Forms.UserControl">User
    Control </see>
    /// that provides an MP3 ID3 tag editor, for a single MP3 file.
    /// <br></br>
    /// <br></br>
    /// This code makes use of the following other classes
    /// <list type="bullet">
    /// <item>
    /// <description><see cref="SB54_CSAI.Genres.MThreadSingleton_genres">Genres singleton
    object </see></description>
    /// </item>
    /// <item>
    /// <description><see cref="SB54_CSAI.MP3Lib.MP3Tag">MP3Tag class </see></description>
    /// </item>
    /// </list>
    /// </summary>
    public class uctMP3Editor : System.Windows.Forms.UserControl
    {
        #region Instance fields

        private System.Windows.Forms.Panel oPnlFill;
        private System.Windows.Forms.PictureBox oPicSearch;
        private System.Windows.Forms.LinkLabel oLnkSearch;
        private System.Windows.Forms.ImageList oImgIcons;
        private System.Windows.Forms.ToolTip oToolTip1;
        private System.Windows.Forms.TextBox oTxtTitle;
        private System.Windows.Forms.TextBox oTxtArtist;
        private System.Windows.Forms.TextBox oTxtAlbum;
        private System.Windows.Forms.TextBox oTxtYear;
        private System.Windows.Forms.TextBox oTxtComment;
        private System.Windows.Forms.TextBox oTxtTrackNo;
        private System.Windows.Forms.ComboBox oCmbGenres;
        private System.Windows.Forms.Button oBtnDone;
        private System.Windows.Forms.Label oLblAlbum;
        private System.Windows.Forms.Label oLblYear;
        private System.Windows.Forms.Label oLblComment;
        private System.Windows.Forms.Label oLblTrackNo;
        private System.Windows.Forms.Label oLblGenres;
        private System.Windows.Forms.Label oLblArtist;
        private System.Windows.Forms.Label oLblTitle;
        private System.Windows.Forms.Panel oPnlGroup;
        private System.Windows.Forms.OpenFileDialog oOpnFileDialog;
        private System.ComponentModel.IContainer components;
        private System.Windows.Forms.Panel oPnlBottom;
        private System.Windows.Forms.Label oLblTask;
        private string currentMP3Name;
        private MP3 workingMP3;
        private System.Windows.Forms.Label oLblTrackSmall;
        private System.Windows.Forms.Label oLblTrack;
        private System.Windows.Forms.PictureBox oPicTrack;
        private DBAccess db;
        #endregion
        #region Public Constructor
        /// <summary>
        /// Constructor, simply initalises all GUI components
        /// and a new <see cref="SB54_CSAI.DatabaseAccess.DBAccess">DataBase access object
    
```

```

</see>
    /// </summary>
    public uctMP3Editor()
    {
        // This call is required by the Windows.Forms Form Designer.
        InitializeComponent();
        //create DataBase access object
        db = new DBAccess();
    }
#endregion
#region Public Methods
    /// <summary>
    /// Clean up any resources being used.
    /// </summary>
    /// <param name="disposing">true indicates the object should be disposed</param>
    protected override void Dispose( bool disposing )
    {
        //if being asked to dispose
        if( disposing )
        {
            //clean up the held resources
            if (components != null) { components.Dispose(); }
        }
        //call base class Dispose
        base.Dispose( disposing );
    }
#endregion
#region Component Designer generated code
    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    private void InitializeComponent()
    {
        this.components = new System.ComponentModel.Container();
        System.Resources.ResourceManager resources = new System.Resources.
ResourceManager(typeof(uctMP3Editor));
        this.oPnlFill = new System.Windows.Forms.Panel();
        this.oPicTrack = new System.Windows.Forms.PictureBox();
        this.oLblTrackSmall = new System.Windows.Forms.Label();
        this.oLblTrack = new System.Windows.Forms.Label();
        this.oPnlBottom = new System.Windows.Forms.Panel();
        this.oLblTask = new System.Windows.Forms.Label();
        this.oLnkSearch = new System.Windows.Forms.LinkLabel();
        this.oPicSearch = new System.Windows.Forms.PictureBox();
        this.oPnlGroup = new System.Windows.Forms.Panel();
        this.oLblTitle = new System.Windows.Forms.Label();
        this.oImgIcons = new System.Windows.Forms.ImageList(this.components);
        this.oLblArtist = new System.Windows.Forms.Label();
        this.oLblGenres = new System.Windows.Forms.Label();
        this.oLblTrackNo = new System.Windows.Forms.Label();
        this.oLblComment = new System.Windows.Forms.Label();
        this.oLblYear = new System.Windows.Forms.Label();
        this.oLblAlbum = new System.Windows.Forms.Label();
        this.oBtnDone = new System.Windows.Forms.Button();
        this.oCmbGenres = new System.Windows.Forms.ComboBox();
        this.oTxtTrackNo = new System.Windows.Forms.TextBox();
        this.oTxtComment = new System.Windows.Forms.TextBox();
        this.oTxtYear = new System.Windows.Forms.TextBox();
        this.oTxtAlbum = new System.Windows.Forms.TextBox();
        this.oTxtArtist = new System.Windows.Forms.TextBox();
        this.oTxtTitle = new System.Windows.Forms.TextBox();
        this.oToolTip1 = new System.Windows.Forms.ToolTip(this.components);
        this.oOpnFileDialog = new System.Windows.Forms.OpenFileDialog();
        this.oPnlFill.SuspendLayout();
        this.oPnlBottom.SuspendLayout();
        this.oPnlGroup.SuspendLayout();
        this.SuspendLayout();
        //
        // oPnlFill
        //
        this.oPnlFill.Controls.Add(this.oPicTrack);
        this.oPnlFill.Controls.Add(this.oLblTrackSmall);

```

```

        this.oPnlFill.Controls.Add(this.oLblTrack);
        this.oPnlFill.Controls.Add(this.oPnlBottom);
        this.oPnlFill.Controls.Add(this.oLnkSearch);
        this.oPnlFill.Controls.Add(this.oPicSearch);
        this.oPnlFill.Controls.Add(this.oPnlGroup);
        this.oPnlFill.Dock = System.Windows.Forms.DockStyle.Fill;
        this.oPnlFill.Location = new System.Drawing.Point(0, 0);
        this.oPnlFill.Name = "oPnlFill";
        this.oPnlFill.Size = new System.Drawing.Size(1008, 688);
        this.oPnlFill.TabIndex = 3;
        this.oPnlFill.Resize += new System.EventHandler(this.oPnlFill_Resize);
        this.oPnlFill.Paint += new System.Windows.Forms.PaintEventHandler(this.oPnlFill_Paint);
        //
        // oPicTrack
        //
        this.oPicTrack.BackColor = System.Drawing.Color.Transparent;
        this.oPicTrack.Image = ((System.Drawing.Image)(resources.GetObject("oPicTrack.
Image")));
        this.oPicTrack.Location = new System.Drawing.Point(16, 8);
        this.oPicTrack.Name = "oPicTrack";
        this.oPicTrack.Size = new System.Drawing.Size(64, 56);
        this.oPicTrack.SizeMode = System.Windows.Forms.PictureBoxSizeModeSizeMode.CenterImage;
        this.oPicTrack.TabIndex = 29;
        this.oPicTrack.TabStop = false;
        //
        // oLblTrackSmall
        //
        this.oLblTrackSmall.BackColor = System.Drawing.Color.Transparent;
        this.oLblTrackSmall.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((System.Byte)
(0)));
        this.oLblTrackSmall.Location = new System.Drawing.Point(80, 24);
        this.oLblTrackSmall.Name = "oLblTrackSmall";
        this.oLblTrackSmall.RightToLeft = System.Windows.Forms.RightToLeft.No;
        this.oLblTrackSmall.Size = new System.Drawing.Size(488, 32);
        this.oLblTrackSmall.TabIndex = 28;
        this.oLblTrackSmall.Text = "No file currently selected";
        this.oLblTrackSmall.TextAlign = System.Drawing.ContentAlignment.MiddleLeft;
        this.oToolTip1.SetToolTip(this.oLblTrackSmall, "Currently selected MP3");
        //
        // oLblTrack
        //
        this.oLblTrack.BackColor = System.Drawing.Color.Transparent;
        this.oLblTrack.Font = new System.Drawing.Font("Microsoft Sans Serif", 11.25F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((System.Byte)(0)));
        this.oLblTrack.Location = new System.Drawing.Point(80, 8);
        this.oLblTrack.Name = "oLblTrack";
        this.oLblTrack.RightToLeft = System.Windows.Forms.RightToLeft.No;
        this.oLblTrack.Size = new System.Drawing.Size(184, 23);
        this.oLblTrack.TabIndex = 27;
        this.oLblTrack.Text = "Currently Selected Track";
        //
        // oPnlBottom
        //
        this.oPnlBottom.BackColor = System.Drawing.Color.White;
        this.oPnlBottom.Controls.Add(this.oLblTask);
        this.oPnlBottom.Dock = System.Windows.Forms.DockStyle.Bottom;
        this.oPnlBottom.ForeColor = System.Drawing.Color.White;
        this.oPnlBottom.Location = new System.Drawing.Point(0, 664);
        this.oPnlBottom.Name = "oPnlBottom";
        this.oPnlBottom.Size = new System.Drawing.Size(1008, 24);
        this.oPnlBottom.TabIndex = 26;
        this.oPnlBottom.Paint += new System.Windows.Forms.PaintEventHandler(this.oPnlBottom_Paint);
        //
        // oLblTask
        //
        this.oLblTask.BackColor = System.Drawing.Color.Transparent;
        this.oLblTask.Font = new System.Drawing.Font("Tahoma", 8.25F, System.Drawing.
FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((System.Byte)(0)));
        this.oLblTask.ForeColor = System.Drawing.Color.White;
        this.oLblTask.Location = new System.Drawing.Point(8, 3);

```

```
this.oLblTask.Name = "oLblTask";
this.oLblTask.RightToLeft = System.Windows.Forms.RightToLeft.No;
this.oLblTask.Size = new System.Drawing.Size(592, 18);
this.oLblTask.TabIndex = 27;
//
// oLnkSearch
//
this.oLnkSearch.BackColor = System.Drawing.Color.Transparent;
this.oLnkSearch.Location = new System.Drawing.Point(104, 104);
this.oLnkSearch.Name = "oLnkSearch";
this.oLnkSearch.RightToLeft = System.Windows.Forms.RightToLeft.No;
this.oLnkSearch.Size = new System.Drawing.Size(216, 23);
this.oLnkSearch.TabIndex = 13;
this.oLnkSearch.TabStop = true;
this.oLnkSearch.Text = "Click Here To Select An MP3 File To Edit";
this.oToolTip1.SetToolTip(this.oLnkSearch, "Select a new MP3, by browsing here")
;
    this.oLnkSearch.LinkClicked += new System.Windows.Forms.
LinkLabelLinkClickedEventHandler(this.oLnkSearch_LinkClicked);
    this.oLnkSearch.MouseEnter += new System.EventHandler(this.
oLnkSearch_MouseEnter);
    this.oLnkSearch.MouseLeave += new System.EventHandler(this.MsgLeave);
//
// oPicSearch
//
this.oPicSearch.BackColor = System.Drawing.Color.Transparent;
this.oPicSearch.Image = ((System.Drawing.Image)(resources.GetObject("oPicSearch.
Image")));
    this.oPicSearch.Location = new System.Drawing.Point(8, 80);
    this.oPicSearch.Name = "oPicSearch";
    this.oPicSearch.Size = new System.Drawing.Size(88, 72);
    this.oPicSearch.TabIndex = 12;
    this.oPicSearch.TabStop = false;
//
// oPnlGroup
//
this.oPnlGroup.BackColor = System.Drawing.Color.Transparent;
this.oPnlGroup.Controls.Add(this.oLblTitle);
this.oPnlGroup.Controls.Add(this.oLblArtist);
this.oPnlGroup.Controls.Add(this.oLblGenres);
this.oPnlGroup.Controls.Add(this.oLblTrackNo);
this.oPnlGroup.Controls.Add(this.oLblComment);
this.oPnlGroup.Controls.Add(this.oLblYear);
this.oPnlGroup.Controls.Add(this.oLblAlbum);
this.oPnlGroup.Controls.Add(this.oBtnDone);
this.oPnlGroup.Controls.Add(this.oCmbGenres);
this.oPnlGroup.Controls.Add(this.oTxtTrackNo);
this.oPnlGroup.Controls.Add(this.oTxtComment);
this.oPnlGroup.Controls.Add(this.oTxtYear);
this.oPnlGroup.Controls.Add(this.oTxtAlbum);
this.oPnlGroup.Controls.Add(this.oTxtArtist);
this.oPnlGroup.Controls.Add(this.oTxtTitle);
this.oPnlGroup.Location = new System.Drawing.Point(312, 144);
this.oPnlGroup.Name = "oPnlGroup";
this.oPnlGroup.Size = new System.Drawing.Size(400, 328);
this.oPnlGroup.TabIndex = 25;
//
// oLblTitle
//
this.oLblTitle.BackColor = System.Drawing.Color.Transparent;
this.oLblTitle.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((System.Byte)(0)));
this.oLblTitle.ForeColor = System.Drawing.Color.Black;
this.oLblTitle.ImageAlign = System.Drawing.ContentAlignment.MiddleLeft;
this.oLblTitle.ImageIndex = 0;
this.oLblTitle.ImageList = this.oImgIcons;
this.oLblTitle.Location = new System.Drawing.Point(16, 0);
this.oLblTitle.Name = "oLblTitle";
this.oLblTitle.Size = new System.Drawing.Size(100, 40);
this.oLblTitle.TabIndex = 37;
this.oLblTitle.Text = "Title";
this.oLblTitle.TextAlign = System.Drawing.ContentAlignment.MiddleRight;
//
```

```
// oImgIcons
//
this.oImgIcons.ColorDepth = System.Windows.Forms.ColorDepth.Depth24Bit;
this.oImgIcons.ImageSize = new System.Drawing.Size(32, 36);
this.oImgIcons.ImageStream = ((System.Windows.Forms.ImageListStreamer)(resources
.GetObject("oImgIcons.ImageStream")));
this.oImgIcons.TransparentColor = System.Drawing.Color.Transparent;
//
// oLblArtist
//
this.oLblArtist.BackColor = System.Drawing.Color.Transparent;
this.oLblArtist.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((System.Byte)(0)));
this.oLblArtist.ForeColor = System.Drawing.Color.Black;
this.oLblArtist.ImageAlign = System.Drawing.ContentAlignment.MiddleLeft;
this.oLblArtist.ImageIndex = 1;
this.oLblArtist.ImageList = this.oImgIcons;
this.oLblArtist.Location = new System.Drawing.Point(16, 40);
this.oLblArtist.Name = "oLblArtist";
this.oLblArtist.Size = new System.Drawing.Size(100, 40);
this.oLblArtist.TabIndex = 36;
this.oLblArtist.Text = "Artist";
this.oLblArtist.TextAlign = System.Drawing.ContentAlignment.MiddleRight;
//
// oLblGenres
//
this.oLblGenres.BackColor = System.Drawing.Color.Transparent;
this.oLblGenres.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((System.Byte)(0)));
this.oLblGenres.ForeColor = System.Drawing.Color.Black;
this.oLblGenres.ImageAlign = System.Drawing.ContentAlignment.MiddleLeft;
this.oLblGenres.ImageIndex = 6;
this.oLblGenres.ImageList = this.oImgIcons;
this.oLblGenres.Location = new System.Drawing.Point(16, 240);
this.oLblGenres.Name = "oLblGenres";
this.oLblGenres.Size = new System.Drawing.Size(100, 40);
this.oLblGenres.TabIndex = 35;
this.oLblGenres.Text = "Genres";
this.oLblGenres.TextAlign = System.Drawing.ContentAlignment.MiddleRight;
//
// oLblTrackNo
//
this.oLblTrackNo.BackColor = System.Drawing.Color.Transparent;
this.oLblTrackNo.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((System.Byte)(0)));
this.oLblTrackNo.ForeColor = System.Drawing.Color.Black;
this.oLblTrackNo.ImageAlign = System.Drawing.ContentAlignment.MiddleLeft;
this.oLblTrackNo.ImageIndex = 5;
this.oLblTrackNo.ImageList = this.oImgIcons;
this.oLblTrackNo.Location = new System.Drawing.Point(16, 200);
this.oLblTrackNo.Name = "oLblTrackNo";
this.oLblTrackNo.Size = new System.Drawing.Size(100, 40);
this.oLblTrackNo.TabIndex = 34;
this.oLblTrackNo.Text = "Track No";
this.oLblTrackNo.TextAlign = System.Drawing.ContentAlignment.MiddleRight;
//
// oLblComment
//
this.oLblComment.BackColor = System.Drawing.Color.Transparent;
this.oLblComment.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((System.Byte)(0)));
this.oLblComment.ForeColor = System.Drawing.Color.Black;
this.oLblComment.ImageAlign = System.Drawing.ContentAlignment.MiddleLeft;
this.oLblComment.ImageIndex = 4;
this.oLblComment.ImageList = this.oImgIcons;
this.oLblComment.Location = new System.Drawing.Point(16, 160);
this.oLblComment.Name = "oLblComment";
this.oLblComment.Size = new System.Drawing.Size(100, 40);
this.oLblComment.TabIndex = 33;
this.oLblComment.Text = "Comment";
this.oLblComment.TextAlign = System.Drawing.ContentAlignment.MiddleRight;
//
// oLblYear
```

```
//
this.oLblYear.BackColor = System.Drawing.Color.Transparent;
this.oLblYear.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((System.Byte)(0)));
this.oLblYear.ForeColor = System.Drawing.Color.Black;
this.oLblYear.ImageAlign = System.Drawing.ContentAlignment.MiddleLeft;
this.oLblYear.ImageIndex = 3;
this.oLblYear.ImageList = this.oImgIcons;
this.oLblYear.Location = new System.Drawing.Point(16, 120);
this.oLblYear.Name = "oLblYear";
this.oLblYear.Size = new System.Drawing.Size(100, 40);
this.oLblYear.TabIndex = 32;
this.oLblYear.Text = "Year";
this.oLblYear.TextAlign = System.Drawing.ContentAlignment.MiddleRight;
//
// oLblAlbum
//
this.oLblAlbum.BackColor = System.Drawing.Color.Transparent;
this.oLblAlbum.Font = new System.Drawing.Font("Microsoft Sans Serif", 8.25F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((System.Byte)(0)));
this.oLblAlbum.ForeColor = System.Drawing.Color.Black;
this.oLblAlbum.ImageAlign = System.Drawing.ContentAlignment.MiddleLeft;
this.oLblAlbum.ImageIndex = 2;
this.oLblAlbum.ImageList = this.oImgIcons;
this.oLblAlbum.Location = new System.Drawing.Point(16, 80);
this.oLblAlbum.Name = "oLblAlbum";
this.oLblAlbum.Size = new System.Drawing.Size(100, 40);
this.oLblAlbum.TabIndex = 31;
this.oLblAlbum.Text = "Album";
this.oLblAlbum.TextAlign = System.Drawing.ContentAlignment.MiddleRight;
//
// oBtnDone
//
this.oBtnDone.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
this.oBtnDone.Image = ((System.Drawing.Image)(resources.GetObject("oBtnDone.
Image")));
this.oBtnDone.Location = new System.Drawing.Point(128, 280);
this.oBtnDone.Name = "oBtnDone";
this.oBtnDone.Size = new System.Drawing.Size(248, 40);
this.oBtnDone.TabIndex = 29;
this.oBtnDone.TextAlign = System.Drawing.ContentAlignment.BottomCenter;
this.oToolTip1.SetToolTip(this.oBtnDone, "Update the attributes of the current
MP3");
this.oBtnDone.Click += new System.EventHandler(this.oBtnDone_Click);
this.oBtnDone.MouseEnter += new System.EventHandler(this.oBtnDone_MouseEnter);
this.oBtnDone.MouseLeave += new System.EventHandler(this.MsgLeave);
//
// oCmbGenres
//
this.oCmbGenres.ItemHeight = 13;
this.oCmbGenres.Location = new System.Drawing.Point(128, 248);
this.oCmbGenres.Name = "oCmbGenres";
this.oCmbGenres.RightToLeft = System.Windows.Forms.RightToLeft.No;
this.oCmbGenres.Size = new System.Drawing.Size(248, 21);
this.oCmbGenres.TabIndex = 28;
this.oToolTip1.SetToolTip(this.oCmbGenres, "Choose a genre for the current MP3");
;
this.oCmbGenres.MouseEnter += new System.EventHandler(this.
oCmbGenres_MouseEnter);
this.oCmbGenres.MouseLeave += new System.EventHandler(this.MsgLeave);
//
// oTxtTrackNo
//
this.oTxtTrackNo.Location = new System.Drawing.Point(128, 208);
this.oTxtTrackNo.MaxLength = 3;
this.oTxtTrackNo.Name = "oTxtTrackNo";
this.oTxtTrackNo.RightToLeft = System.Windows.Forms.RightToLeft.No;
this.oTxtTrackNo.Size = new System.Drawing.Size(248, 20);
this.oTxtTrackNo.TabIndex = 27;
this.oTxtTrackNo.Text = "";
this.oToolTip1.SetToolTip(this.oTxtTrackNo, "Set the track number (3 characters)
for the current MP3");
this.oTxtTrackNo.MouseEnter += new System.EventHandler(this.
```

```
oTxtTrackNo_MouseEnter);
    this.oTxtTrackNo.MouseLeave += new System.EventHandler(this.MsgLeave);
    this.oTxtTrackNo.KeyUp += new System.Windows.Forms.KeyEventHandler(this.
oTxtTrackNo_KeyUp);
    //
    // oTxtComment
    //
    this.oTxtComment.Location = new System.Drawing.Point(128, 168);
    this.oTxtComment.MaxLength = 28;
    this.oTxtComment.Name = "oTxtComment";
    this.oTxtComment.RightToLeft = System.Windows.Forms.RightToLeft.No;
    this.oTxtComment.Size = new System.Drawing.Size(248, 20);
    this.oTxtComment.TabIndex = 26;
    this.oTxtComment.Text = "";
    this.oToolTip1.SetToolTip(this.oTxtComment, "Set a comment (28 characters) for
the current MP3");
    this.oTxtComment.MouseEnter += new System.EventHandler(this.
oTxtComment_MouseEnter);
    this.oTxtComment.MouseLeave += new System.EventHandler(this.MsgLeave);
    //
    // oTxtYear
    //
    this.oTxtYear.Location = new System.Drawing.Point(128, 128);
    this.oTxtYear.MaxLength = 4;
    this.oTxtYear.Name = "oTxtYear";
    this.oTxtYear.RightToLeft = System.Windows.Forms.RightToLeft.No;
    this.oTxtYear.Size = new System.Drawing.Size(248, 20);
    this.oTxtYear.TabIndex = 25;
    this.oTxtYear.Text = "";
    this.oToolTip1.SetToolTip(this.oTxtYear, "Set the year (4 characters) for the
current MP3");
    this.oTxtYear.MouseEnter += new System.EventHandler(this.oTxtYear_MouseEnter);
    this.oTxtYear.MouseLeave += new System.EventHandler(this.MsgLeave);
    this.oTxtYear.KeyUp += new System.Windows.Forms.KeyEventHandler(this.
oTxtYear_KeyUp);
    //
    // oTxtAlbum
    //
    this.oTxtAlbum.Location = new System.Drawing.Point(128, 88);
    this.oTxtAlbum.MaxLength = 30;
    this.oTxtAlbum.Name = "oTxtAlbum";
    this.oTxtAlbum.RightToLeft = System.Windows.Forms.RightToLeft.No;
    this.oTxtAlbum.Size = new System.Drawing.Size(248, 20);
    this.oTxtAlbum.TabIndex = 24;
    this.oTxtAlbum.Text = "";
    this.oToolTip1.SetToolTip(this.oTxtAlbum, "Set the album name (30 characters)
for the current MP3");
    this.oTxtAlbum.MouseEnter += new System.EventHandler(this.oTxtAlbum_MouseEnter);
    this.oTxtAlbum.MouseLeave += new System.EventHandler(this.MsgLeave);
    //
    // oTxtArtist
    //
    this.oTxtArtist.Location = new System.Drawing.Point(128, 48);
    this.oTxtArtist.MaxLength = 30;
    this.oTxtArtist.Name = "oTxtArtist";
    this.oTxtArtist.RightToLeft = System.Windows.Forms.RightToLeft.No;
    this.oTxtArtist.Size = new System.Drawing.Size(248, 20);
    this.oTxtArtist.TabIndex = 16;
    this.oTxtArtist.Text = "";
    this.oToolTip1.SetToolTip(this.oTxtArtist, "Set the artist name (30 characters)
for the current MP3");
    this.oTxtArtist.MouseEnter += new System.EventHandler(this.
oTxtArtist_MouseEnter);
    this.oTxtArtist.MouseLeave += new System.EventHandler(this.MsgLeave);
    //
    // oTxtTitle
    //
    this.oTxtTitle.Location = new System.Drawing.Point(128, 8);
    this.oTxtTitle.MaxLength = 30;
    this.oTxtTitle.Name = "oTxtTitle";
    this.oTxtTitle.RightToLeft = System.Windows.Forms.RightToLeft.No;
    this.oTxtTitle.Size = new System.Drawing.Size(248, 20);
    this.oTxtTitle.TabIndex = 15;
```

```

        this.oTxtTitle.Text = "";
        this.oToolTip1.SetToolTip(this.oTxtTitle, "Set the working title (30 characters)
for the current MP3");
        this.oTxtTitle.MouseEnter += new System.EventHandler(this.oTxtTitle_MouseEnter);
        this.oTxtTitle.MouseLeave += new System.EventHandler(this.MsgLeave);
        //
        // oOpnFileDlg
        //
        this.oOpnFileDlg.Filter = "MP3 Files (*.mp3)|*.mp3";
        this.oOpnFileDlg.InitialDirectory = "c:\\\\";
        this.oOpnFileDlg.RestoreDirectory = true;
        this.oOpnFileDlg.Title = "Please Select An MP3 To Edit";
        //
        // uctMP3Editor
        //
        this.BackColor = System.Drawing.SystemColors.Control;
        this.Controls.Add(this.oPnlFill);
        this.Name = "uctMP3Editor";
        this.RightToLeft = System.Windows.Forms.RightToLeft.Yes;
        this.Size = new System.Drawing.Size(1008, 688);
        this.Resize += new System.EventHandler(this.uctMP3Editor_Resize);
        this.Load += new System.EventHandler(this.uctMP3Editor_Load);
        this.oPnlFill.ResumeLayout(false);
        this.oPnlBottom.ResumeLayout(false);
        this.oPnlGroup.ResumeLayout(false);
        this.ResumeLayout(false);
    }
#endregion
#region Private Methods

#region MP3 Operations
    /// <summary>
    /// Makes use of the <see cref="SB54_CSAI.Genres.MThreadSingleton_genres">Genres
singleton object </see>
    /// to get the list of Genres. This list of Genres is then added to the Genre
combobox
    ///</summary>
    private void fillGenres()
    {
        //clear combo box
        oCmbGenres.Items.Clear();
        //get the Genres
        ArrayList arGenres = MThreadSingleton_genres.Instance.getGenres();

        //add them to the combobox
        for (int i=0; i < arGenres.Count; i++)
        {
            oCmbGenres.Items.Add((string)arGenres[i]);
        }
        //add the Genres to the database, as they might not exist yet, and may be
required later by
        //other controls, like the Media Library.
        db.createGenres(arGenres);
    }

    /// <summary>
    /// Takes a parameter that represents the filename to read MP3 ID3 tag information
from
    /// and uses the <see cref="SB54_CSAI.MP3Lib.MP3Tag">MP3Tag class </see> to read the
ID3
    /// tag information.
    /// </summary>
    /// <param name="filename">the filename to read MP3 ID3 tag information from</param>
    private void setMP3Details(string filename)
    {
        try
        {
            //start assuming no MP3Tag exists
            bool MP3File = true;
            //check the file is an MP3 file
            FileInfo fFileInfo = new FileInfo(filename);
            if (fFileInfo.Extension.ToLower() != ".mp3")

```

```

    {
        //not MP3
        MP3File = false;
    }
    else
    {
        //Is MP3 so read its TAG
        workingMP3 = new MP3(fFileInfo.DirectoryName, fFileInfo.Name);
        MP3Tag.readMP3Tag (ref workingMP3);
    }
    //only make text boxes enabled if there is an MP3 Tag to read
    oPnlGroup.Enabled = (workingMP3.hasID3Tag && MP3File);
    //set the details for this MP3 file
    oTxtAlbum.Text = workingMP3.id3Album;
    oTxtArtist.Text = workingMP3.id3Artist;
    oTxtComment.Text = workingMP3.id3Comment;
    oCmbGenres.SelectedIndex = (int)workingMP3.id3Genre;
    oTxtTitle.Text = workingMP3.id3Title;
    oTxtTrackNo.Text = ((int)workingMP3.id3TrackNumber).ToString();
    oTxtYear.Text = workingMP3.id3Year;
}
catch (Exception ex)
{
    MessageBox.Show("Problem with setMP3Details uctMP3editor : " +
        "\r\n\r\n\r\n" + ex.Message,
        "File Access ERROR", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
}
}

/// <summary>
/// Checks that a Text box contains numeric data ONLY
/// </summary>
/// <param name="t">The text box to check for numeric data</param>
/// <returns>true if the Text box contains numeric data</returns>
private bool checkNumeric(TextBox t)
{
    if (t.Text != null)
    {
        //check for empty string
        if (! t.Text.Equals(String.Empty) )
        {
            //^ = Matches any single character not in the specified set of
characters.
            //\d = Matches any decimal digit.
            //+ = Specifies one or more matches
            //$ = Specifies that the match must occur at the end of the string
            // before \n at the end of the string, or at the end of the line.
            Regex isNumber = new Regex(@"^\d+$");
            //represents results from a single regular expression
            Match m = isNumber.Match(t.Text);
            if (! m.Success)
            {
                //Not Numeric
                return false;
            }
            else
            {
                //Is Numeric OK
                return true;
            }
        }
    }
    else
    {
        //Empty String, so Not Numeric
        return false;
    }
}
return false;
}

#endregion
#region GUI User interactions

```

```
/// <summary>
/// Allows the user to search the file system for an MP3 file
/// to edit the ID3 tag data for
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oLnkSearch_LinkClicked(object sender, System.Windows.Forms.
LinkLabelLinkClickedEventArgs e)
{
    //if user selected OK from File Dialog
    if(oOpnFileDlg.ShowDialog() == DialogResult.OK)
    {
        try
        {
            //get the filename
            string[] files = oOpnFileDlg.FileNames;
            //only take the 1st one, can only edit 1 at at time
            string file = files[0];
            System.IO.DirectoryInfo d = new System.IO.DirectoryInfo(file);
            //check its actually and MP3
            if (d.Extension.Length == 0 || ! d.Extension.ToLower().Equals(".mp3"))
            {
                //Its not an MP3 file
                MessageBox.Show("The file you selected is not a valid MP3 file " +
                    "\r\n\r\n\r\n" + "Please retry",
                    "File Selection ERROR", MessageBoxButtons.OK,
                    MessageBoxIcon.Error);
                oPnlGroup.Enabled = false;
                oLblTask.Visible = false;
            }
            else
            {
                //Its a valid MP3 file
                currentMP3Name = file;
                oPnlGroup.Enabled = true;
                oLblTask.Visible = true;
                oLblTrackSmall.Text = currentMP3Name;
                setMP3Details(currentMP3Name);
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show("Problem with oLnkSearch_LinkClicked uctMP3editor : " +
                "\r\n\r\n\r\n" + ex.Message,
                "File Access ERROR", MessageBoxButtons.OK,
                MessageBoxIcon.Error);
        }
    }
}

/// <summary>
/// Event handler that occurs when the mouse enters the Title text field
/// Simply sets the status bar text to be Title text field Tooltip
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oTxtTitle_MouseEnter(object sender, System.EventArgs e)
{
    oLblTask.Text = oToolTip1.GetToolTip(oTxtTitle);
}

/// <summary>
/// Event handler that occurs when the mouse enters the Artist text field
/// Simply sets the status bar text to be Artist text field Tooltip
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oTxtArtist_MouseEnter(object sender, System.EventArgs e)
{
    oLblTask.Text = oToolTip1.GetToolTip(oTxtArtist);
}
```

```
/// <summary>
/// Event handler that occurs when the mouse enters the Album text field
/// Simply sets the status bar text to be Album text field Tooltip
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oTxtAlbum_MouseEnter(object sender, System.EventArgs e)
{
    oLblTask.Text = oToolTip1.GetToolTip(oTxtAlbum);
}

/// <summary>
/// Event handler that occurs when the mouse enters the Year text field
/// Simply sets the status bar text to be Year text field Tooltip
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oTxtYear_MouseEnter(object sender, System.EventArgs e)
{
    oLblTask.Text = oToolTip1.GetToolTip(oTxtYear);
}

/// <summary>
/// Event handler that occurs when the mouse enters the Comment text field
/// Simply sets the status bar text to be Comment text field Tooltip
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oTxtComment_MouseEnter(object sender, System.EventArgs e)
{
    oLblTask.Text = oToolTip1.GetToolTip(oTxtComment);
}

/// <summary>
/// Event handler that occurs when the mouse enters the Track No text field
/// Simply sets the status bar text to be Track No text field Tooltip
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oTxtTrackNo_MouseEnter(object sender, System.EventArgs e)
{
    oLblTask.Text = oToolTip1.GetToolTip(oTxtTrackNo);
}

/// <summary>
/// Event handler that occurs when the mouse enters the Genre combo box
/// Simply sets the status bar text to be Genre combo box Tooltip
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oCmbGenres_MouseEnter(object sender, System.EventArgs e)
{
    oLblTask.Text = oToolTip1.GetToolTip(oCmbGenres);
}

/// <summary>
/// Event handler that occurs when the mouse enters the Done button area
/// Simply sets the status bar text to be Done button Tooltip
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oBtnDone_MouseEnter(object sender, System.EventArgs e)
{
    oLblTask.Text = oToolTip1.GetToolTip(oBtnDone);
}

/// <summary>
/// Event handler that occurs when the mouse enters the Search Link area
/// Simply sets the status bar text to be Search Link Tooltip
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oLnkSearch_MouseEnter(object sender, System.EventArgs e)
```

```

    {
        oLblTask.Text = oToolTip1.GetToolTip(oLnkSearch);
    }

    /// <summary>
    /// Event handler that occurs when the key up occurs for the Year text field
    /// A call to the checkNumeric() method is made with the current Year text field
    /// contents
    /// </summary>
    /// <param name="sender">The source object that raised the event</param>
    /// <param name="e">The event arguments</param>
    private void oTxtYear_KeyUp(object sender, System.Windows.Forms.KeyEventArgs e)
    {
        //check year is numeric
        if (!checkNumeric(oTxtYear) && !oTxtYear.Text.Equals(string.Empty))
        {
            MessageBox.Show("Year must be numeric, please retry",
                "Data Entry ERROR", MessageBoxButtons.OK,
                MessageBoxIcon.Error);
            oTxtYear.Text = "";
        }
    }

    /// <summary>
    /// Event handler that occurs when the key up occurs for the Track No text field
    /// The Track No text field contents is parsed to n integer and compared to
    /// a track limit between 0-255, if it is outside these limits a default value
    /// of 255 will be given to the Track No text field
    /// </summary>
    /// <param name="sender">The source object that raised the event</param>
    /// <param name="e">The event arguments</param>
    private void oTxtTrackNo_KeyUp(object sender, System.Windows.Forms.KeyEventArgs e)
    {
        //only woory about non empty fields
        if (!oTxtTrackNo.Text.Equals(string.Empty))
        {
            //check year is numeric
            if (!checkNumeric(oTxtTrackNo))
            {
                MessageBox.Show("Track No must be numeric, please retry",
                    "Data Entry ERROR", MessageBoxButtons.OK,
                    MessageBoxIcon.Error);
                oTxtTrackNo.Text = "";
                return;
            }

            //check its range, if not ok, use default of 255 for Track No
            if (Int32.Parse(oTxtTrackNo.Text) > 255)
            {
                MessageBox.Show("Track Number Must be between 0-255" +
                    "\r\n\r\n\r\n" + "So has been clamped to 255",
                    "Data Entry ERROR", MessageBoxButtons.OK,
                    MessageBoxIcon.Error);
                oTxtTrackNo.Text = "255";
            }
        }
    }

    /// <summary>
    /// Event handler that occurs when the key up occurs for the Done button is clicked
    /// by the user. The current MP3 details are updated using the current GUI field
    values.
    /// The <see cref="SB54_CSAI.DatabaseAccess.DBAccess">DataBase access object </see>
    /// </summary> is also used to update the MP3, if it exists.
    /// <param name="sender">The source object that raised the event</param>
    /// <param name="e">The event arguments</param>
    private void oBtnDone_Click(object sender, System.EventArgs e)
    {
        try

```

```

    {
        //set the details for this MP3 file
        workingMP3.id3Album = (!oTxtAlbum.Text.Equals(String.Empty) ? oTxtAlbum.Text
: "");
        workingMP3.id3Artist = (!oTxtArtist.Text.Equals(String.Empty) ? oTxtArtist.
Text : "");
        workingMP3.id3Comment = (!oTxtComment.Text.Equals(String.Empty) ?
oTxtComment.Text : "");
        workingMP3.id3Genre = (oCmbGenres.SelectedIndex != -1 ? (Byte)oCmbGenres.
SelectedIndex : new Byte());
        workingMP3.id3Title = (!oTxtTitle.Text.Equals(String.Empty) ? oTxtTitle.Text
: "");
        workingMP3.id3TrackNumber = (!oTxtTrackNo.Text.Equals(String.Empty) ? (Byte)
Byte.Parse(oTxtTrackNo.Text) : new Byte());
        workingMP3.id3Year = (!oTxtYear.Text.Equals(String.Empty) ? oTxtYear.Text :
"");

        //update the MP3Tag with new information
        MP3Tag.updateMP3Tag(ref workingMP3);
        //see if the MP3 exists in the database, if it does, update all its details
with these
        //new details. Database class looks after these operations
        db.updateMP3(ref workingMP3);

        //clear the details
        oTxtAlbum.Text = "";
        oTxtArtist.Text = "";
        oTxtComment.Text = "";
        oCmbGenres.SelectedIndex = -1;
        oTxtTitle.Text = "";
        oTxtTrackNo.Text = "";
        oTxtYear.Text = "";
        oPnlGroup.Enabled = false;
        oLblTask.Visible = false;
        MessageBox.Show("You have sucessfully updated the Mp3 details for file\r\n\r\n
\r\n" +
            oLblTrackSmall.Text, "MP3 Update Confirm", MessageBoxButtons.OK,
            MessageBoxIcon.Information);
    }
    catch (Exception ex)
    {
        MessageBox.Show("Problem with File Access in setMP3Details uctMP3editor : "
+
            "\r\n\r\n\r\n\r\n" + ex.Message,
            "File Access ERROR", MessageBoxButtons.OK,
            MessageBoxIcon.Error);
    }
}
#endregion
#region GUI look and feel methods

/// <summary>
/// Event handler that occurs when the main uctMP3Editor paints
/// Simply paints the main uctMP3Editor panel using a colorBlend
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oPnlFill_Paint(object sender, System.Windows.Forms.PaintEventArgs e)
{
    try
    {
        base.OnPaint(e);

        Color[] myColors =
        {
            Color.Gainsboro,
            Color.White,
            Color.White };
        float[] myPositions = {0.0f,0.4f,1.0f};
        ColorBlend myBlend = new ColorBlend();
        myBlend.Colors = myColors;
        myBlend.Positions = myPositions;

        Rectangle r1 = new Rectangle(0,0,oPnlFill.Bounds.Width,oPnlFill.Bounds.

```

```
Height);
    LinearGradientBrush lgBrush2 = new LinearGradientBrush(r1,Color.Gainsboro,
Color.White,45.0f,true);
    lgBrush2.InterpolationColors = myBlend;
    e.Graphics.FillRectangle(lgBrush2,0,0,oPnlFill.Bounds.Width,oPnlFill.Bounds.
Height);
    // Create font and brush.
    Font drawFont = new Font("Arial", 14,FontStyle.Bold, GraphicsUnit.Pixel);
    SolidBrush drawBrush = new SolidBrush(Color.White);
}
//nothing to do occurs when, going from Minimised to maxmise and rectange to
draw is effectievely
//0 Wide 0 High, which causes an exception. Cant do anything about it.
catch(Exception ex)
{
}
}

/// <summary>
/// Event handler that occurs when the main uctMP3Editor panel resizes.
/// Simply used to refresh the main uctMP3Editor panel area.
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oPnlFill_Resize(object sender, System.EventArgs e)
{
    oPnlFill.Invalidate(oPnlFill.Bounds);
    this.Refresh();
}

/// <summary>
/// Event handler that occurs when the main uctMP3Editor panel resizes.
/// Simply reposition all the relevant GUI components based on the new
/// client window size.
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void uctMP3Editor_Resize(object sender, System.EventArgs e)
{
    oPnlGroup.Left = (oPnlFill.Width/2 - oPnlGroup.Width/2);
}

/// <summary>
/// Event handler that occurs when the main uctMP3Editor_Load panel loads.
/// Call the fillGenres() method, and start in disabled statem user must
/// pick an MP3 to enable GUI controls
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void uctMP3Editor_Load(object sender, System.EventArgs e)
{
    //call the fillGenres() method
    fillGenres();

    //start in disabled statem user must pick an MP3 to enable GUI controls
    oPnlGroup.Enabled = false;
    oLblTask.Visible = false;
}

/// <summary>
/// Used as a generic event delegate hanlser that all the GUI components make use of
.
/// Simply clears the text within the bottom panel oPnlBottom
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void MsgLeave(object sender, System.EventArgs e)
{
    oLblTask.Text=" ";
}
```

```
/// <summary>
/// Event handler that occurs when the main oPnlBottom paints
/// Simply paints the panel hatched
/// </summary>
/// <param name="sender">The source object that raised the event</param>
/// <param name="e">The event arguments</param>
private void oPnlBottom_Paint(object sender, System.Windows.Forms.PaintEventArgs e)
{
    base.OnPaint(e);
    //draw the rectangle
    Graphics g = e.Graphics;
    Pen penBlack = new Pen(Color.Black);
    Rectangle r = new Rectangle(0,0,oPnlBottom.Size.Width,oPnlBottom.Size.Height);
    g.DrawRectangle(penBlack,r);
    Brush divBrush = new HatchBrush(HatchStyle.Divot,Color.Gray,Color.Black);
    g.FillRectangle(divBrush,r);
}

#endregion

#endregion
}
#endregion
}
```