

Manual Técnico

Progra - 0092

Contenido

Manual Técnico.....	1
Progra - 0092.....	1
INTRODUCCIÓN	3
Información destacada	3
REQUISITOS DEL SISTEMA	4
Datos estratégicos relevantes	4
INSTALACIÓN Y CONFIGURACIÓN	5
PROGRAMA	6
Conexion	7
Artista	7
LoginUsuario	8
Cancion	9
Crear Playlist	10
Editar usuario	11
Registra usuario	12
Funciones	13
Anexos	14

INTRODUCCIÓN

El siguiente trabajo describe los aspectos técnicos informáticos del sistema de información. El documento familiariza al personal técnico especializado encargado de las actividades de mantenimiento, revisión, solución de problemas, instalación y configuración del sistema.

Información destacada

Este manual es con el fin de orientar y referenciar a la información necesaria para orientar al personal, planteamiento del análisis programación e instalación del sistema.

Este manual es referido a personal con conocimientos en programación avanzada, sobre el uso de estructura de datos, listas (enlazadas, dobles, circularmente doblemente enlazada) y pilas, colas.

Objetivos

Instruir el uso correcto del Sistema de Información, para el acceso oportuno y continuamente adecuado, descripción de los archivos relevantes del sistema, con los que se podrá orientas en la configuración y soporte de este

REQUISITOS DEL SISTEMA

Datos estratégicos relevantes

Este proyecto tiene la finalidad de explicar cómo hacer uso del programa, que ira conforme al usuario, y la finalidad de ser agradable a su uso, para la manipulación del usuario y sus respectivos procedimientos.

Está cargara los datos, en reportes especializados para las operaciones establecidas.

Objetivos:

Dirigido al usuario para su uso correcto durante el proceso del manejo del sistema.

Se ofrece una guía general para la aplicación, y establecer del uso correcto.

Este programa se realizó en Visual Code, lenguajes C#.

Requisitos del sistema:

- Librerías: C#, Visual Studio

CPU	Intel(R) Core 2 Duo
RAM	8 GB Ram
Espacio del disco duro	40 mb
Sistema operativo	Windows 7,8,10

INSTALACIÓN Y CONFIGURACIÓN

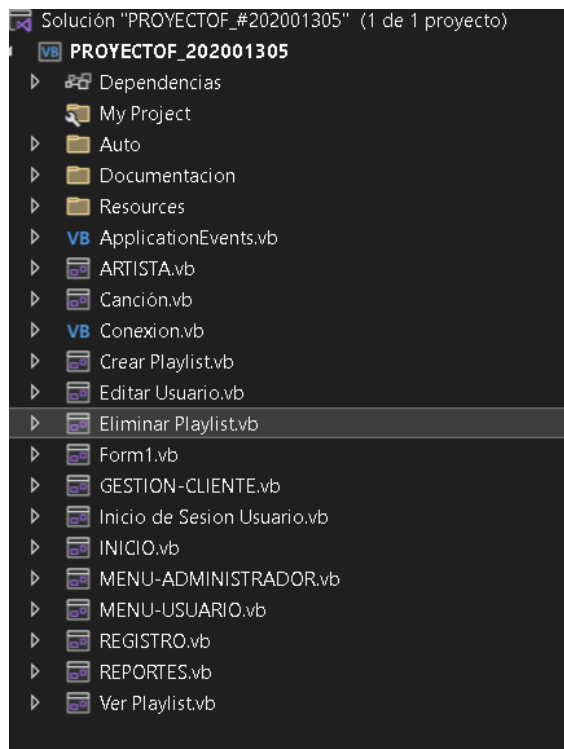
Existen muchos métodos de instalación, sin embargo, a continuación, se describe de forma sencilla y segura la instalación, la que genera un link simbólico.

1. Descomprima el rar, para su uso directo.
2. Instale Xamp

PROGRAMA

Estructura raíz

A continuación, describimos los directorios y archivos más importantes:



En esta ocasión se dividió el proceso en 2 fases diferentes para mantener el orden y un mejor manejo de las estructuras.

Conexion

Aquí revisamos la conexión con xamp.

```
Imports MySql.Data.MySqlClient

0 referencias
Module Conexion

    Public connection As New MySqlConnection
    Public comandos As New MySqlCommand

    ' Datos para usuario
    Public UsuarioActual As String
    Public TipoUsuarioActual As String
    Public Id_usuario_date As Integer

    3 referencias
    Public Function conexionBD()

        Try
            connection = New MySqlConnection
            connection.ConnectionString = "server=localhost;username=root;password=;database=proyecto_spofity"
            connection.Open() 'Abrimos la conexion y metodos

            MsgBox("Conexión EXITOSA a Base de Datos!")

            Return connection
        Catch ex As Exception
            MsgBox("Error al conectar Base de Datos")
            Return Nothing ' Retornamos en caso de error
        End Try
    End Function
End Module
```

Artista

Este hace el registro de los artistas en el proceso

```
Conexion.vb  Canción.vb  ARTISTA.vb  ARTISTA.vb [Diseño]
VB PROYECTO_F_202001305  examinarBT  Click

1 Imports System.IO
2 Imports MySql.Data.MySqlClient
3
4 2 referencias
5 Public Class ARTISTA
6     Private imagenSeleccionada As Image
7     Private rutaImagen As String ' Variable para almacenar la ruta de la imagen
8
9     0 referencias
10    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles ButtonRegMenu.Click
11        Me.Hide()
12        MENU_ADMINISTRADOR.Show()
13    End Sub
14
15    0 referencias
16    Private Sub examinarBT_Click(sender As Object, e As EventArgs) Handles examinarBT.Click
17
18        OpenFileDialog1.Filter = "Archivos de imagen|.jpg;*.jpeg;*.png;*.gif;*.bmp"
19
20        If OpenFileDialog1.ShowDialog() = DialogResult.OK Then
21            Try
22                rutaImagen = OpenFileDialog1.FileName
23                rutaImagen = rutaImagen.Replace("\", "/")
24                imagenSeleccionada = Image.FromFile(rutaImagen)
25
26                ' Ajustar la imagen al PictureBox2 manteniendo la relación de aspecto
27                PictureBox2.Image = New Bitmap(imagenSeleccionada, PictureBox2.Width, PictureBox2.Height)
28            Catch ex As Exception
29                ' Manejar errores si la carga de la imagen falla
30                MessageBox.Show("Error al cargar la imagen. Detalles: " & ex.Message, "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
31            End Try
32        End If
33    End Sub
34
35    0 referencias
36    Private Sub ButtonAgregar_Click(sender As Object, e As EventArgs) Handles ButtonAgregar.Click
37        Código para guardar un nuevo artista y la ruta de la imagen en la base de datos
38
39        Try
40            Dim nombreArtista As String = TextBoxNombre.Text
41            Dim nacionalidadArtista As String = TextBoxNacionalidad.Text
42            ' Obtener la fecha seleccionada del DateTimePicker
43            Dim fechaInicio As DateTime = DateTimePicker1.Value
44            ' Extraer el año de la fecha seleccionada
45        End Try
46    End Sub
47
```

LoginUsuario

Este realiza los registros del login y realiza las entradas a nuestro programa, así como revisa si se llena los datos correspondientes, llama a la función de verificarsesion y depende de la respuesta obtenida por la función verificarsesion, ira al formulario administrador o el modelo usuario.

```
0 referencias
Private Sub ButtonLogin_Click(sender As Object, e As EventArgs) Handles ButtonLogin.Click
    Dim Usuario As String
    Dim contraseña As String
    Usuario = TextBoxUsername.Text 'guardamos datos del usuario y contraseña
    contraseña = TextBoxContra.Text

    Try
        If Usuario.Length > 0 AndAlso contraseña.Length > 0 Then
            Dim connection = conexionBD1()
            comandos.Connection = connection
            'Codigo de Login
            'Creamos consulta
            comandos.CommandText = $"SELECT ID_USUARIO, tipo_usuario FROM Usuario WHERE username='{Usuario}' AND password='{contraseña}'"
            Dim lectura As MySqlDataReader = comandos.ExecuteReader()

            If lectura.HasRows Then
                lectura.Read()
                ' Obtenemos el tipo de usuario
                Dim tipo_user As String = lectura.GetString(1)

                ' Obtenemos el ID del usuario
                Dim Id_usuario As Integer = lectura.GetInt32(lectura.GetOrdinal("ID_USUARIO"))

                Select Case tipo_user
                    Case "CL"
                        UsuarioActual = Usuario
                        TipoUsuarioActual = tipo_user
                        Id_usuario_date = Id_usuario ' Guardamos el ID del usuario
                        MENU_USUARIO.Show()
                        Me.Hide()
                    Case "Adm"
                        MENU_ADMINISTRADOR.Show()
                        Me.Hide()
                    Case Else
                        MsgBox("Tipo de usuario no valido" & tipo_user)
                End Select

                comandos.Parameters.Clear()

            Else
                TextBoxUsername.Text = ""
                TextBoxContra.Text = ""
                comandos.Parameters.Clear()
                MsgBox("Usuario o Contraseña Incorrectos")
            End If

            connection.Close()
        End If

    Catch ex As Exception
        MsgBox(ex.Message)
    End Try
End Sub
```


Cancion

Se realizará los respectivos registros, así como la actualización modificación y sus respectivos cambios.

```

Conexion.vb  Canción.vb  Canción.vb [Diseño]  REGISTRO.vb [Diseño]
VB PROYECTO_F_202001305  ButtonRegMenu  Click

18
19 1 referencia
20 Private Sub CargarArtistas()
21     ' Conecta a la base de datos.
22     Dim connection = conexionBD1()
23
24     Try
25         ' Crea una consulta para obtener los nombres de los artistas.
26         Dim query As String = "SELECT nombre FROM Artista"
27         Dim command As New MySqlCommand(query, connection)
28         Dim reader As MySqlDataReader = command.ExecuteReader()
29
30         ' Limpia el ComboBox.
31         ComboBox1.Items.Clear()
32
33         ' Agrega los nombres de los artistas al ComboBox.
34         While reader.Read()
35             ComboBox1.Items.Add(reader("nombre").ToString())
36         End While
37     Catch ex As Exception
38         MessageBox.Show("Error al cargar los artistas: " & ex.Message)
39     Finally
40         connection.Close()
41     End Try
42 End Sub
43
44 3 referencias
45 Private Sub CargarTitulosDeCanciones()
46     ' Conecta a la base de datos.
47     Dim connection = conexionBD1()
48
49     Try
50         ' Crea una consulta para obtener los títulos de las canciones.
51         Dim query As String = "SELECT titulo FROM Cancion"
52         Dim command As New MySqlCommand(query, connection)
53         Dim reader As MySqlDataReader = command.ExecuteReader()
54
55         ' Limpia el ComboBox.
56         ComboBox2.Items.Clear()
57
58         ' Agrega los títulos de las canciones al ComboBox.
59         While reader.Read()
60             ComboBox2.Items.Add(reader("titulo").ToString())
61         End While
62     Catch ex As Exception
63         MessageBox.Show("Error al cargar los títulos de las canciones: " & ex.Message)
64     Finally
65         connection.Close()
66     End Try
67 End Sub
68
69 0 referencias
70 Private Sub ButtonAgregar_Click(sender As Object, e As EventArgs) Handles ButtonAgregar.Click
71     ' Verificar si se ha seleccionado un artista en el ComboBox.
72     If ComboBox1.SelectedIndex = -1 Then
73         MsgBox("Por favor, selecciona un artista para crear la canción.")
74     End If
75 End Sub

```

Crear Playlist

```
Public Class Crear_Playlist
    Private rutaCancion As String
    Private imagenRuta As String
    ' Agregar variables para mantener los lectores de datos.
    Private readerCanciones As MySqlDataReader
    Private readerPlaylists As MySqlDataReader

    0 referencias
    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles ButtonRegMenu.Click
        Me.Hide()
        MENU_USUARIO.Show()
    End Sub

    0 referencias
    Private Sub Button3_Click_1(sender As Object, e As EventArgs) Handles Button3.Click
        OpenFileDialog1.Filter = "Imágenes|*.jpg;*.jpeg;*.png;*.gif;*.bmp" ' Ajusta las extensiones de imagen permitidas
        If OpenFileDialog1.ShowDialog() = DialogResult.OK Then
            ' Obtener la ruta de la imagen seleccionada y almacenarla en la variable
            imagenRuta = OpenFileDialog1.FileName

            ' Mostrar la ruta en una etiqueta
            LabelRutaImagen.Text = imagenRuta

            ' Cargar y mostrar la imagen en un PictureBox
            PictureBox2.Image = Image.FromFile(imagenRuta)
            PictureBox2.SizeMode = PictureBoxSizeMode.Zoom
            PictureBox2.Anchor = AnchorStyles.Top Or AnchorStyles.Bottom
        End If
    End Sub

    0 referencias
    Private Sub Crear_Playlist_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        CargarTitulosDeCanciones()
        CargarPlaylists()
    End Sub

    1 referencia
    Private Sub CargarTitulosDeCanciones()
        ' Conecta a la base de datos.
        Dim connection = conexionBD1()

        Try
            ' Cerrar el lector de datos anterior si existe.
            If readerCanciones IsNot Nothing AndAlso Not readerCanciones.IsClosed Then
                readerCanciones.Close()
            End If

            ' Crea una consulta para obtener los títulos de las canciones.
            Dim query As String = "SELECT titulo FROM Canciones"
```

Editar usuario

Se hizo la actualización del usuario y eliminación.

```
Imports MySql.Data.MySqlClient

1 referencia
Public Class Editor_Usuario
0 referencias
    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles ButtonRegMenu.Click
        Me.Hide()
        MENU_USUARIO.Show()
    End Sub

0 referencias
    Private Sub Editar_Usuario_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        ' Realiza una consulta para obtener los datos del usuario actual
        Dim connection = conexionBD1()
        Dim query As String = $"SELECT * FROM Usuario WHERE username = '{UsuarioActual}'"
        Dim command As New MySqlCommand(query, connection)
        Dim reader As MySqlDataReader = command.ExecuteReader()

        If reader.Read() Then
            TextBoxNombre.Text = reader("nombre").ToString()
            TextBoxPassword.Text = reader("password_").ToString()
        End If

        connection.Close()
    End Sub

0 referencias
    Private Sub ButtonActualizar_Click(sender As Object, e As EventArgs) Handles ButtonActualizar.Click
        Dim connection = conexionBD1()

        ' Recopila los datos del usuario desde los TextBox
        Dim nuevoNombre As String = TextBoxNombre.Text
        Dim nuevoPassword As String = TextBoxPassword.Text
        Dim confirmarPassword As String = TextBoxPaswordconf.Text

        ' Verificar si las contraseñas coinciden
        If nuevoPassword = confirmarPassword Then
            ' Realiza una consulta para actualizar los datos del usuario
            Dim query As String = $"UPDATE Usuario SET nombre = '{nuevoNombre}', password_ = '{nuevoPassword}' WHERE username = '{Usua"
            Dim command As New MySqlCommand(query, connection)

            Try
                command.ExecuteNonQuery()
                MsgBox("Datos del usuario actualizados correctamente.")
            Catch ex As Exception
                MsgBox("Error al actualizar los datos del usuario: " & ex.Message)
            Finally
                connection.Close()
            End Try
        Else
            MsgBox("Las contraseñas no coinciden.")
        End If
    End Sub
End Class
```

Registra usuario

Aquí se realizó los registros de los usuarios, así como el ingreso a la base de datos.

```
Imports System.Windows.Forms.VisualStyles.VisualStyleElement.StartPanel
Imports MySql.Data.MySqlClient

1 referencia
Public Class REGISTRO
    0 referencias
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles ButtonRegistrarse.Click
        Try
            If TextBoxIngContraseña.Text = TextBoxConfContraseña.Text Then
                Dim conection = conexionBD1()

                'Codigo Registro Usuario / clientes solamente
                Dim codigoInsertar = $"INSERT INTO Usuario (nombre, username, password_, tipo_usuario)
                $"VALUES ('{TextBoxIngresarNombre.Text}', '{TextBoxUsarname.Text}', " &
                $"'{TextBoxIngContraseña.Text}', 'CL')"
                Dim lectura As MySqlDataReader
                comandos = New MySqlCommand(codigoInsertar, conection)
                lectura = comandos.ExecuteReader
                MsgBox("Usuario registrado correctamente :D")
                conexionBD1.Close()
                'Codigo para dirigirse a la ventana Iniccion de Sesion Usuario
                Me.Hide()
                Inicio_de_Sesion_Usuario.Show()
            Else
                MsgBox("Contraseñas no coinciden revisa por favor...")
            End If
        Catch ex As Exception
            MsgBox("Error en la insercion del cliente..." & ex.Message)
        End Try
    End Sub

    0 referencias
    Private Sub Button1_Click_1(sender As Object, e As EventArgs) Handles Button1.Click
        TextBoxIngresarNombre.Text = ""
        TextBoxUsarname.Text = ""
        TextBoxIngContraseña.Text = ""
        TextBoxConfContraseña.Text = ""
    End Sub

    0 referencias
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        Me.Hide()
        Inicio_de_Sesion_Usuario.Show()
    End Sub
End Class
```

Funciones

Funcion	Descripción	Paquete	Clase
conexionBD()	Realiza la conexión así a la base de datos	MysqlCliente	Conexión
conexionBD1	Se usa como clase extra para la conexión y devolución de la conexión abierta	MysqlCliente	Conexión
CargarArtistas	Esta carga los artistas en un combo box	Form	Cancion
CargarTitulosDeCanciones	Carga los títulos de las canciones respectivas en un combo box	Form	Cancion
LimpiarControles	Limpia los controles	Form	Cancion
CargarPlaylists	Carga los títulos de las playlist a un combo box	Form	Eliminarplaylist
CargarTitulosDeCanciones	Cargar los títulos en un combo box	Form	Crear Playlist
CargarPlaylists	Cargar las playlist en un combo box	Form	Crear Playlist
LlenarComboBoxClientes	Cargar los datos en un combo box de todos los clientes existentes	Form	Artista

Anexos

Auto	17/10/2023 22:54	Carpeta de archivos	
bin	5/9/2023 11:17	Carpeta de archivos	
Documentacion	20/10/2023 01:54	Carpeta de archivos	
My Project	16/10/2023 22:11	Carpeta de archivos	
obj	16/10/2023 13:05	Carpeta de archivos	
Resources	8/9/2023 17:01	Carpeta de archivos	
ApplicationEvents.vb	5/9/2023 11:17	Visual Basic Sourc...	2 KB
ARTISTA.Designer.vb	17/10/2023 22:59	Visual Basic Sourc...	15 KB
ARTISTA.resx	17/10/2023 22:59	Microsoft .NET Ma...	14 KB
ARTISTA.vb	18/10/2023 23:04	Visual Basic Sourc...	10 KB
Canción.Designer.vb	18/10/2023 00:39	Visual Basic Sourc...	16 KB
Canción.resx	18/10/2023 00:39	Microsoft .NET Ma...	3 KB
Canción.vb	18/10/2023 23:03	Visual Basic Sourc...	14 KB
Conexion.vb	20/10/2023 00:32	Visual Basic Sourc...	2 KB
Crear Playlist.Designer.vb	20/10/2023 01:36	Visual Basic Sourc...	14 KB
Crear Playlist.resx	20/10/2023 01:36	Microsoft .NET Ma...	14 KB
Crear Playlist.vb	20/10/2023 01:36	Visual Basic Sourc...	8 KB
Editar Usuario.Designer.vb	20/10/2023 00:33	Visual Basic Sourc...	10 KB
Editar Usuario.resx	20/10/2023 00:33	Microsoft .NET Ma...	3 KB
Editar Usuario.vb	20/10/2023 00:33	Visual Basic Sourc...	4 KB
Eliminar Playlist.Designer.vb	20/10/2023 01:18	Visual Basic Sourc...	7 KB
Eliminar Playlist.resx	20/10/2023 01:18	Microsoft .NET Ma...	3 KB
Eliminar Playlist.vb	20/10/2023 01:18	Visual Basic Sourc...	3 KB
Form1.Designer.vb	16/10/2023 15:47	Visual Basic Sourc...	7 KB
Form1.resx	16/10/2023 15:47	Microsoft .NET Ma...	108 KB
Form1.vb	16/10/2023 15:47	Visual Basic Sourc...	1 KB