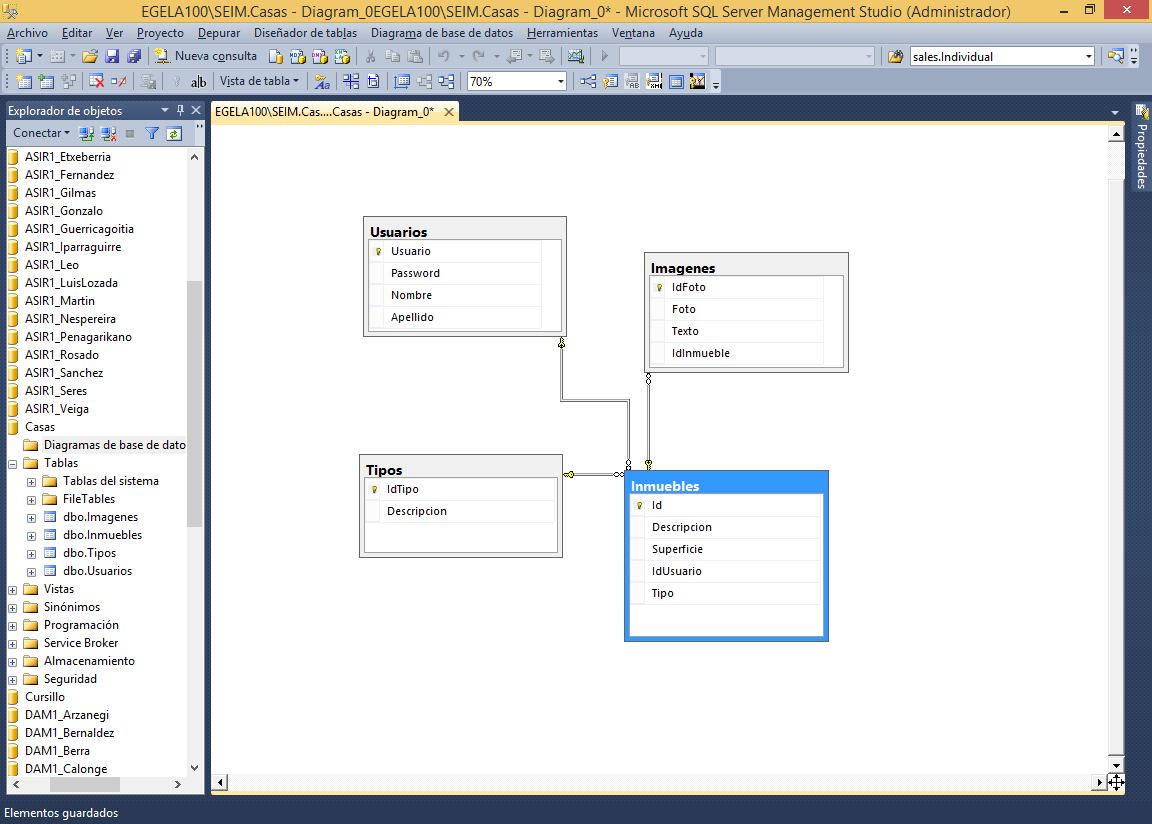
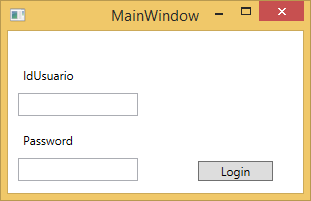
Se quiere implementar una aplicación para mostrar el catálogo de inmuebles a los usuarios de una inmobiliaria.

La base de datos es la siguiente:



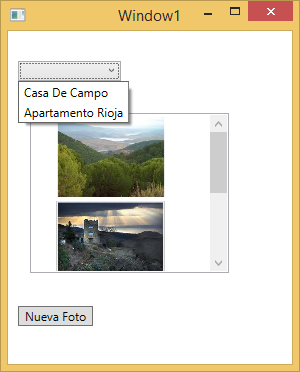
Donde usuarios son los usuarios registrados en la inmobiliaria, tipos los tipos de inmueble a la venta, inmuebles los inmuebles a la venta e imágenes almacena las imágenes de los inmuebles. Un inmueble puede tener más de una imagen.

1. Al cargarse la aplicación mostrará la siguiente ventana:



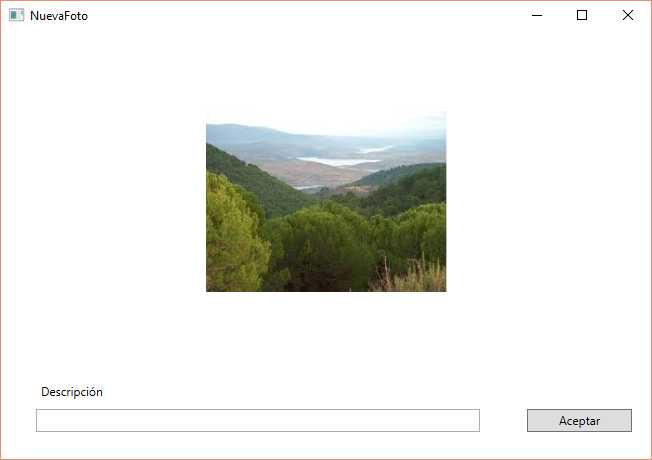
Si el usuario y el password son correctos pasará al formulario principal de la aplicación, si no la aplicación terminará.

1. Una vez cargado mostrará un formulario como el siguiente:



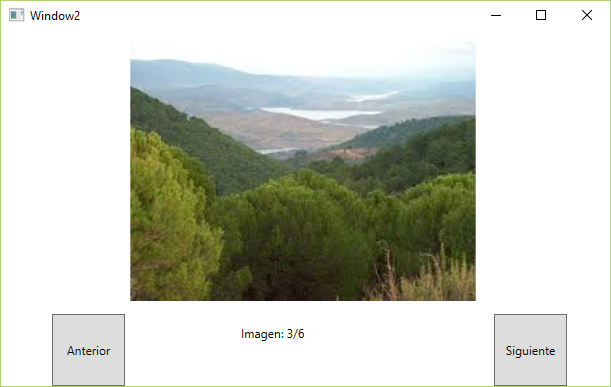
* + 1. El combobox muestre la descripción y el tipo de todos los inmuebles del usuario actual.
    2. La listbox muestra todas las fotos agregadas para el inmueble actual.
    3. Al pulsar nueva foto saldrá el formulario del ejercicio 3.
    4. Al hacer doble click en la listbox saldrá el formulario del ejercicio 4.

1. Al pulsar una nueva foto, saldrá el formulario siguiente, con el que se dará de alta una nueva foto para el inmueble introducido.



Donde al hacer doble click en la imagen se desplegará el cuadro de diálogo para elegir fotografía y al hacer click en Aceptar dará de alta una nueva fotografía para el inmueble actual y se descargará el formulario.

1. Al pulsar doble click en la listbox de la ventana principal aparecerá el siguiente formulario.



De manera que:

* 1. La foto que se visualice será exactamente la que hayamos seleccionado en la listbox aunque vamos a tener que poder navegar todas las imágenes del inmueble mediante anterior y siguiente.
  2. La etiqueta mostrará en qué imagen estamos y cuántas imágenes hay para el inmueble actual (en la figura 3 / 6).
  3. El botón anterior visualizará la imagen anterior, controlar no pasarnos de la posición 0.
  4. El botón siguiente visualizará la imagen siguiente, controlar no pasarnos del total de imágenes.

# Modelo para el viewModel del MainWindow.

Public Class MainViewModel

Private id, pwd As String

Private modelo As DaniDBEntities

Public Sub New(model As BD\_DaniEntities)

''asigna el parámetro a la variable privada

End Sub

Public Property IdUsuario As String

''Obtiene o establece el id de usuario

End Property

Public Property Password As String

''Obtiene o establece el pwd

End Property

Public ReadOnly Property Usuario As Usuarios

Get

''Obtiene el usuario actual

End Get

End Property

Public Sub Validar ()

''Comprueba si existe usuario y password e inicializa usuario Actual

End sub

End Class

Public Class FotoWindowViewModel

Implements INotifyPropertyChanged

Dim inm As Inmuebles

Dim modelo As BD\_DaniEntities

Dim imagen As Imagenes

Public Event PropertyChanged As PropertyChangedEventHandler Implements INotifyPropertyChanged.PropertyChanged

Public Sub New(i As Inmuebles, m As BD\_DaniEntities)

''Inicializa las variables Miebro

End Sub

Public Property Foto As Byte()

''Devuelve o establece la imagen del Inmueble inm. En la parte set

''notifica cambio de propiedad

End Property

Property Descripcion As String

''Devuelve o establece la descripción del Inmueble inm. En la parte set

''notifica cambio de propiedad

End Property

Private Function CalcularId()

''Devuelve el id del siguiente inmueble a insertar (el máximo de los ''existentes o 1 si no hay

End Function

Public Sub Agregar()

''Agrega el nuevo inmueble y guarda los cambios.

End Sub

End Class

Public Class VerFotosViewModel

Implements INotifyPropertyChanged

Dim InmuebleActual As Inmuebles

Dim pos As Integer

Public Sub New(Inmueble As Inmuebles, posicion As Integer)

''Inicializa los miembros con los parámetros

End Sub

Public ReadOnly Property Foto As Byte()

''Devuelve la imagen del inmueble de la posición actual

End Property

Public ReadOnly Property PosicionActual As Integer

''Devuelve la posición actual

End Property

Public ReadOnly Property TotalFotos As Integer

''Devuelve el contador de fotos para el inmueble actual

End Property

Public Event PropertyChanged As PropertyChangedEventHandler Implements

INotifyPropertyChanged.PropertyChanged

Public Sub Avanzar()

''Si no estamos en la última foto del inmueble actual avanza a la

''siguiente y notifica el cambio.

End Sub

Public Sub Retroceder()

''Si no estamos en la última foto del inmueble actual retrocede

''a la anterior y notifica el cambio

End Sub

End Class