



UNIVERSIDAD DE COLIMA

Facultad de Telemática
Ingeniería de software

Reporte de Examen: App para la administración de una tienda con Xamarin

Programación para Móviles
Maestro: Armando Román Gallardo
Grado y grupo: 5°D
Primera parcial

Integrantes:

- Mariana Guadalupe Fajardo Toscano
- Diego de Jesús Villaseñor Villa
- Jose Enrique Ruvalcaba Segura
- Angel Isaac Bejarano Flores
- Jonathan Solis Ceballos

Miércoles 28 de octubre del 2020

MainPage.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<TabbedPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             xmlns:local="clr-namespace:examen"
             x:Class="examen.MainPage"

             xmlns:android="clr-namespace:Xamarin.Forms.PlatformConfiguration.AndroidSpecific;assembly=Xamarin.Forms.Core"

             android:TabbedPage.ToolbarPlacement="Bottom"
             android:TabbedPage.BarItemColor="DarkOrange"
             android:TabbedPage.BarSelectedItemColor="DarkOrange"
             Title="TiendApp"
             SelectedTabColor="OrangeRed"
             BarTextColor="OrangeRed"
             BarBackgroundColor="White"
             >

    <local:Productos/>

    <local:Clientes />

    <local:Ventas />

</TabbedPage>
```

MainPage.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;

namespace examen
{
    public partial class MainPage : TabbedPage
    {
        public MainPage()
        {
            InitializeComponent();
        }
    }
}
```

```
}  
}
```

App.xaml

```
<?xml version="1.0" encoding="utf-8" ?>  
<Application xmlns="http://xamarin.com/schemas/2014/forms"  
              xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"  
              x:Class="examen.App">  
    <Application.Resources>  
  
    </Application.Resources>  
</Application>
```

App.xaml.cs

```
using System;  
using Xamarin.Forms;  
using Xamarin.Forms.Xaml;  
  
namespace examen  
{  
    public partial class App : Application  
    {  
        public App()  
        {  
            InitializeComponent();  
            //MainPage = new MainPage();  
            MainPage = new NavigationPage( new MainPage() );  
        }  
  
        protected override void OnStart()  
        {  
        }  
  
        protected override void OnSleep()  
        {  
        }  
  
        protected override void OnResume()  
        {  
        }  
    }  
}
```

Clase BaseDeDatos.cs

```
using System;
using System.Collections.Generic;
using System.Text;

namespace examen
{
    class BaseDeDatos
    {
        public static string bdName = "tienda10.db";
    }
}
```

Implementación Cámara

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    android:versionCode="1" android:versionName="1.0"
    package="com.companynome.examen" android:installLocation="auto">
    <uses-sdk android:minSdkVersion="21"
    android:targetSdkVersion="28" />
    <application android:label="examen.Android"
    android:theme="@style/MainTheme">
        <provider android:name="android.support.v4.content.FileProvider"
    android:authorities="${applicationId}.fileprovider"
        android:exported="false"
        android:grantUriPermissions="true">
            <meta-data android:name="android.support.FILE_PROVIDER_PATHS"
        android:resource="@xml/file_paths"></meta-data>
        </provider>
    </application>
    <uses-permission
    android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission
    android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-permission
    android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
</manifest>
```

file_paths.xml (Creado dentro de la carpeta xml, que a su vez fue creada en Resources de Android)

```
<?xml version="1.0" encoding="utf-8"?>
<paths xmlns:android="http://schemas.android.com/apk/res/android">
    <external-files-path name="my_images" path="Pictures" />
    <external-files-path name="my_movies" path="Movies" />
</paths>
```

Productos

Clase Producto.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;
using SQLite;

namespace examen
{
    public class Producto
    {
        [PrimaryKey]
        [AutoIncrement]
        public int Id { get; set; }
        public string Nombre { get; set; }
        public double PreciodeVenta { get; set; }
        public int Cantidad { get; set; }
        public double PreciodeCompra { get; set; }
        public string Foto { get; set; }
    }
}
```

Productos.xaml

```

<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="examen.Productos"
             Title="Productos"
             IconImageSource="products.png"
             BackgroundColor="White"
             >

    <ContentPage.ToolbarItems>
        <ToolbarItem Name="tbiAgregar"
                     Order="Primary"
                     Icon="plus.png"
                     Text="Agregar"
                     Priority="0"
                     Clicked="tbiAgregar_Clicked"
                     />
    </ContentPage.ToolbarItems>

    <ContentPage.Content>

        <StackLayout>

            <ListView x:Name="lvProductos" HasUnevenRows="True"
            Margin="15,5,15,15" ItemTapped="lvProductos_ItemTapped">
                <ListView.ItemTemplate>
                    <DataTemplate>
                        <ViewCell>
                            <Frame BorderColor="#ccc"
                            BackgroundColor="White" Margin="5" CornerRadius="5">
                                <StackLayout Orientation="Horizontal">
                                    <StackLayout
                                    Orientation="Vertical">
                                        <Label Text="Nombre"
                                        FontSize="Micro" FontAttributes="Bold" TextColor="Black"/>
                                        <Label Text="{Binding Nombre}"
                                        FontSize="Large" TextColor="#c90000"></Label>
                                        <Label Text="Cantidad"
                                        FontSize="Micro" FontAttributes="Bold" TextColor="Black"/>
                                        <Label Text="{Binding
                                        Cantidad}" TextColor="Salmon" FontSize="Small"></Label>
                                        <Label Text="Precio de compra"
                                        FontSize="Micro" FontAttributes="Bold" TextColor="Black"/>

```

```

                                <Label Text="{Binding
PreciodeCompra}" TextColor="Salmon" FontSize="Small"></Label>
                                <Label Text="Precio de venta"
FontSize="Micro" FontAttributes="Bold" TextColor="Black"/>
                                <Label Text="{Binding
PreciodeVenta}" TextColor="Salmon" FontSize="Small"></Label>
                                </StackLayout>
                                <Image Source="{Binding Foto}"
HeightRequest="30" WidthRequest="150" HorizontalOptions="EndAndExpand"
Margin="5"/>
                                </StackLayout>
                            </Frame>
                        </ViewCell>
                    </DataTemplate>
                </ListView.ItemTemplate>
            </ListView>

        </StackLayout>
    </ContentPage.Content>
</ContentPage>

```

Productos.xaml.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;
using Xamarin.Forms.Xaml;
using SQLite;

namespace examen
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class Productos : ContentPage
    {
        public void AbrirBase()
        {
            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);

```

```

        string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);

        // Crea la base de datos si no existe, y crea una conexión
        var db = new SQLiteConnection(rutaDb);
        db.CreateTable<Producto>();
        var todoslosproductos = db.Table<Producto>().ToList();

        lvProductos.ItemsSource = null;
        lvProductos.ItemsSource = todoslosproductos;
    }

    public Productos()
    {
        InitializeComponent();

        lvProductos.IsRefreshing = false;
        this.Appearing += Productos_Appearing;
    }

    private void Productos_Appearing(object sender, EventArgs e)
    {
        AbrirBase();
    }

    // Agregar producto
    async private void tbiAgregar_Clicked(object sender, EventArgs
e)
    {
        //r AgregarProductoPage = new NavigationPage(new
AgregarProducto());
        await Navigation.PushAsync(new AgregarProducto());
        AbrirBase();
    }

    // Detalle del producto
    private void lvProductos_ItemTapped(object sender,
ItemTappedEventArgs e)
    {
        var producto = e.Item as Producto;
        var detailPage = new DetalleProducto();
        detailPage.BindingContext = producto;
    }

```



```

        Navigation.PushAsync(detailPage);
    }
}

```

AgregarProducto.xaml

```

<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="examen.AgregarProducto"
    >

    <ContentPage.ToolbarItems>
        <ToolbarItem Name="tbiGuardar" Order="Primary" Icon="save.png"
            Text="Guardar" Priority="0" Clicked="tbiGuardar_Clicked" />
    </ContentPage.ToolbarItems>

    <ContentPage.Content>
        <ScrollView>
            <StackLayout Margin="35,15,15,15">
                <Label Text="Nombre" FontSize="Medium"/>
                <Entry x:Name="nombre" Visual="Material"/>
                <Label Text="Cantidad" FontSize="Medium" />
                <Entry x:Name="cantidad" Keyboard="Numeric"/>
                <Label Text="Precio de compra" FontSize="Medium" />
                <Entry x:Name="preciodecompra" Keyboard="Numeric" />
                <Label Text="Precio de venta" FontSize="Medium" />
                <Entry x:Name="preciodeventa" Keyboard="Numeric"/>
                <Label Text="Foto URL" FontSize="Medium" />
                <Entry x:Name="foto"/>
                <Button x:Name="TakePhoto" Grid.Row="1" Text="Camara"/>
            </StackLayout>
        </ScrollView>
    </ContentPage.Content>
</ContentPage>

```

AgregarProducto.xaml.cs

```

using System;
using System.Collections.Generic;
using System.Linq;

```

```

using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;
using SQLite;
using Java.Lang;

namespace examen
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class AgregarProducto : ContentPage
    {
        public AgregarProducto()
        {
            InitializeComponent();
            TakePhoto.Clicked += TakePhoto_Clicked;
        }

        private bool addProduct()
        {
            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
            string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);
            // DisplayAlert("Ruta de la base de datos", rutaDb, "ok");
            // Crea la base de datos si no existe, y crea una conexión
            var db = new SQLiteConnection(rutaDb);

            // Crea la tabla si no existe
            db.CreateTable<Producto>();

            if (string.IsNullOrEmpty(nombre.Text) ||
                string.IsNullOrEmpty(foto.Text) ||
                string.IsNullOrEmpty(preciodecompra.Text) ||
                string.IsNullOrEmpty(preciodeventa.Text) ||
                string.IsNullOrEmpty(cantidad.Text))
            {
                DisplayAlert("Error al agregar", "Todos los campos son
obligatorios", "OK");
                return false;
            }
        }
    }
}

```

```

        var registro = new Producto
        {
            Nombre = nombre.Text,
            PreciodeCompra = double.Parse(preciodecompra.Text),
            Cantidad = int.Parse(cantidad.Text),
            PreciodeVenta = double.Parse(preciodeventa.Text),
            Foto = foto.Text
        };

        db.Insert(registro);
        DisplayAlert("Registro agregado", "El registro fue agregado
con exito!", "OK");
        return true;
    }

    private async void TakePhoto_Clicked(object sender, EventArgs
e)
    {
        var photo =
            await
Plugin.Media.CrossMedia.Current.TakePhotoAsync(new
Plugin.Media.Abstractions.StoreCameraMediaOptions());
        if (photo != null)
        {
            foto.Text = photo.Path;
        }
    }

    private void tbiGuardar_Clicked(object sender, EventArgs e)
    {
        if (addProduct())
            Application.Current.MainPage.Navigation.PopAsync();
    }
}
}

```

DetalleProducto.xaml

```

<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="examen.DetalleProducto"

```

```

    >

    <ContentPage.ToolbarItems>
        <ToolbarItem Name="tbiSave" Order="Primary" Icon="listo.png"
Text="Guardar" Priority="0" Clicked="tbiSave_Clicked" />
        <ToolbarItem Name="tbiDelete" Order="Primary" Icon="borrar.png"
Text="Borrar" Priority="1" Clicked="tbiDelete_Clicked" />
    </ContentPage.ToolbarItems>
    <ContentPage.Content>
        <ScrollView>
            <StackLayout>
                <Image Source="{Binding Foto}" />
                <StackLayout Margin="35,15,15,15">
                    <Label x:Name="Id" Text="{Binding Id}"
IsVisible="False"/>
                    <Label Text="Nombre" FontSize="Micro"/>
                    <Entry x:Name="nombre" Text="{Binding Nombre}"
FontSize="Small"/>
                    <Label Text="Cantidad" FontSize="Micro"/>
                    <Entry x:Name="cantidad" Text="{Binding Cantidad}"
Keyboard="Numeric" FontSize="Small"/>
                    <Label Text="Precio de compra" FontSize="Micro"/>
                    <Entry x:Name="preciodecompra" Text="{Binding
PreciodeCompra}" Keyboard="Numeric" FontSize="Small"/>
                    <Label Text="Precio de venta" FontSize="Micro" />
                    <Entry x:Name="preciodeventa" Text="{Binding
PreciodeVenta}" Keyboard="Numeric" FontSize="Small"/>
                    <Label Text="Foto URL" FontSize="Micro"/>
                    <Entry x:Name="foto" Text="{Binding Foto}"
FontSize="Small"/>
                </StackLayout>
            </StackLayout>
        </ScrollView>
    </ContentPage.Content>
</ContentPage>

```

DetalleProducto.xaml.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

```

```

using Xamarin.Forms;
using Xamarin.Forms.Xaml;
using SQLite;

namespace examen
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class DetalleProducto : ContentPage
    {
        public DetalleProducto()
        {
            InitializeComponent();
        }

        private void tbiSave_Clicked(object sender, EventArgs e)
        {
            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
            string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);
            var db = new SQLiteConnection(rutaDb);

            int MiId = int.Parse(Id.Text);
            //DisplayAlert("ID", "" + MiId, "Ok");

            if (string.IsNullOrEmpty(nombre.Text) ||
                string.IsNullOrEmpty(foto.Text) ||
                string.IsNullOrEmpty(preciodecompra.Text) ||
                string.IsNullOrEmpty(preciodeventa.Text) ||
                string.IsNullOrEmpty(cantidad.Text))
            {
                DisplayAlert("Error al actualizar", "Todos los campos
son obligatorios", "OK");
                return;
            }

            var registro = new Producto
            {
                Id = MiId,
                Nombre = nombre.Text,
                PreciodeCompra = double.Parse(preciodecompra.Text),
                Cantidad = int.Parse(cantidad.Text),
            }
        }
    }
}

```

```

        PreciodeVenta = double.Parse(preciodeventa.Text),
        Foto = foto.Text
    };

    db.Table<Producto>();

    db.Update(registro);
    // DisplayAlert("Anuncio", "Registro Actualizado!!", "ok");

    Application.Current.MainPage.Navigation.PopAsync();
}

async private void tbiDelete_Clicked(object sender, EventArgs
e)
{
    string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
    string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);
    var db = new SQLiteConnection(rutaDb);

    int MiId = int.Parse(Id.Text);

    var respuesta = await DisplayAlert("Eliminar producto",
"¿Desea eliminar el producto?", "Si", "No");

    if (respuesta == true)
    {
        db.Delete<Producto>(MiId);
        await
Application.Current.MainPage.Navigation.PopAsync();
    }
}
}
}

```

Cientes

Capturas de pantalla de la interfaz

Clase Cliente.cs

```
using System;
using System.Collections.Generic;
using System.Text;
using SQLite;

namespace examen
{
    class Cliente
    {
        [PrimaryKey]
        [AutoIncrement]
        public int Id { get; set; }
        public string Nombre { get; set; }
        public string Domicilio { get; set; }
        public int Telefono { get; set; }
        public string Correo { get; set; }
        public string Foto { get; set; }
    }
}
```

Cientes.xaml

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
```

```

        x:Class="examen.Clientes"
        Title="Clientes"
        IconImageSource="customer.png"
        BackgroundColor="White"
    >

<ContentPage.ToolbarItems>
    <ToolbarItem Name="tbiAgregar"
        Order="Primary"
        Icon="plus.png"
        Text="Agregar"
        Priority="0"
        Clicked="tbiAgregar_Clicked"
    />
</ContentPage.ToolbarItems>

<ContentPage.Content>

    <StackLayout>

        <ListView x:Name="lvClientes" HasUnevenRows="True" Margin="15,5,15,15"
ItemTapped="lvClientes_ItemTapped">
            <ListView.ItemTemplate>
                <DataTemplate>
                    <ViewCell>
                        <Frame BorderColor="#ccc" BackgroundColor="White" Margin="5"
CornerRadius="5">
                            <StackLayout Orientation="Horizontal">
                                <StackLayout Orientation="Vertical">
                                    <Label Text="Nombre" FontSize="Micro" FontAttributes="Bold"
TextColor="Black"/>
                                    <Label Text="{Binding Nombre}" FontSize="Large"
TextColor="#c90000"/></Label>
                                    <Label Text="Domicilio" FontSize="Micro" FontAttributes="Bold"
TextColor="Black"/>
                                    <Label Text="{Binding Domicilio}" TextColor="Salmon"
FontSize="Small"/></Label>
                                    <Label Text="Telefono" FontSize="Micro" FontAttributes="Bold"
TextColor="Black"/>
                                    <Label Text="{Binding Telefono}" TextColor="Salmon"
FontSize="Small"/></Label>
                                    <Label Text="Correo" FontSize="Micro" FontAttributes="Bold"
TextColor="Black"/>
                                    <Label Text="{Binding Correo}" TextColor="Salmon"
FontSize="Small"/></Label>
                                </StackLayout>
                                <Image Source="{Binding Foto}" HeightRequest="30"
WidthRequest="150" HorizontalOptions="EndAndExpand" Margin="5"/>

```



```

        </StackLayout>
    </Frame>
</ViewCell>
</DataTemplate>
</ListView.ItemTemplate>
</ListView>

</StackLayout>
</ContentPage.Content>
</ContentPage>

```

Cientes.xaml.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;
using Xamarin.Forms.Xaml;
using SQLite;

namespace examen
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class Cientes : ContentPage
    {
        public void AbrirBase()
        {
            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.LocalApplicationDa
ta);
            string rutaDb = System.IO.Path.Combine(folder, BaseDeDatos.bdName);

            // Crea la base de datos si no existe, y crea una conexión
            var db = new SQLiteConnection(rutaDb);
            db.CreateTable<Cliente>();
            var todoslosclientes = db.Table<Cliente>().ToList();

            lvClientes.ItemsSource = null;
            lvClientes.ItemsSource = todoslosclientes;
        }

        public Cientes()
        {
            InitializeComponent();
            lvClientes.IsRefreshing = false;
            this.Appearing += Cientes_Appearing;

```

```

    }

    private void Clientes_Appearing(object sender, EventArgs e)
    {
        AbrirBase();
    }

    // Agregar producto
    async private void tbiAgregar_Clicked(object sender, EventArgs e)
    {
        //r AgregarProductoPage = new NavigationPage(new AgregarProducto());
        await Navigation.PushAsync(new AgregarCliente());
        AbrirBase();
    }

    // Detalle del producto
    private void lvClientes_ItemTapped(object sender, ItemTappedEventArgs e)
    {
        var cliente = e.Item as Cliente;
        var detailPage = new DetalleCliente();
        detailPage.BindingContext = cliente;
        Navigation.PushAsync(detailPage);
    }
}
}

```

AgregarCliente.xaml

```

<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="examen.AgregarCliente">
    <ContentPage.ToolbarItems>
        <ToolbarItem Name="tbiGuardar" Order="Primary" Icon="save.png" Text="Guardar"
            Priority="0" Clicked="tbiGuardar_Clicked" />
    </ContentPage.ToolbarItems>

    <ContentPage.Content>
        <ScrollView>
            <StackLayout Margin="35,15,15,15">
                <Label Text="Nombre" FontSize="Medium"/>
                <Entry x:Name="nombre" Visual="Material"/>
                <Label Text="Domicilio" FontSize="Medium" />
                <Entry x:Name="domicilio" Visual="Material"/>
                <Label Text="Telefono" FontSize="Medium" />
                <Entry x:Name="telefono" Keyboard="Numeric" />
                <Label Text="Correo" FontSize="Medium" />
                <Entry x:Name="correo" Visual="Material"/>
            </StackLayout>
        </ScrollView>
    </ContentPage.Content>
</ContentPage>

```

```

        <Label Text="Foto URL" FontSize="Medium" />
        <Entry x:Name="foto"/>
    </StackLayout>
</ScrollView>
</ContentPage.Content>
</ContentPage>

```

AgregarCliente.xaml.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;
using SQLite;

namespace examen
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class AgregarCliente : ContentPage
    {
        public AgregarCliente()
        {
            InitializeComponent();
        }

        private void addClient()
        {
            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.LocalApplicationDa
ta);
            string rutaDb = System.IO.Path.Combine(folder, BaseDeDatos.bdName);
            // DisplayAlert("Ruta de la base de datos", rutaDb, "ok");
            // Crea la base de datos si no existe, y crea una conexión
            var db = new SQLiteConnection(rutaDb);

            // Crea la tabla si no existe
            db.CreateTable<Cliente>();

            var registro = new Cliente
            {
                Nombre = nombre.Text,
                Domicilio = domicilio.Text,
                Telefono = int.Parse(telefono.Text),
                Correo = correo.Text,

```

```

        Foto = foto.Text
    };

    db.Insert(registro);
    DisplayAlert("Registro agregado", "El registro fue agregado con exito!", "OK");
}

private void tbiGuardar_Clicked(object sender, EventArgs e)
{
    addClient();
    Application.Current.MainPage.Navigation.PopAsync();
}
}
}

```

DetalleCliente.xaml

```

<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="examen.DetalleCliente"
    >

    <ContentPage.ToolbarItems>
        <ToolbarItem Name="tbiSave" Order="Primary" Icon="listo.png" Text="Guardar"
            Priority="0" Clicked="tbiSave_Clicked" />
        <ToolbarItem Name="tbiDelete" Order="Primary" Icon="borrar.png" Text="Borrar"
            Priority="1" Clicked="tbiDelete_Clicked" />
    </ContentPage.ToolbarItems>
    <ContentPage.Content>
        <ScrollView>
            <StackLayout>
                <Image Source="{Binding Foto}" />
                <StackLayout Margin="35,15,15,15">
                    <Label x:Name="Id" Text="{Binding Id}" IsVisible="False" />
                    <Label Text="Nombre" FontSize="Micro" />
                    <Entry x:Name="nombre" Text="{Binding Nombre}" FontSize="Small" />
                    <Label Text="Domicilio" FontSize="Micro" />
                    <Entry x:Name="domicilio" Text="{Binding Domicilio}" FontSize="Small" />
                    <Label Text="Telefono" FontSize="Micro" />
                    <Entry x:Name="telefono" Text="{Binding Telefono}" Keyboard="Numeric"
                        FontSize="Small" />
                    <Label Text="Correo" FontSize="Micro" />
                    <Entry x:Name="correo" Text="{Binding Correo}" FontSize="Small" />
                    <Label Text="Foto URL" FontSize="Micro" />
                    <Entry x:Name="foto" Text="{Binding Foto}" FontSize="Small" />
                </StackLayout>
            </StackLayout>
        </ScrollView>
    </ContentPage.Content>

```

```

        </ScrollView>
    </ContentPage.Content>
</ContentPage>

```

DetalleCliente.xaml.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;
using SQLite;

namespace examen
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class DetalleCliente : ContentPage
    {
        public DetalleCliente()
        {
            InitializeComponent();
        }

        private void tbiSave_Clicked(object sender, EventArgs e)
        {
            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.LocalApplicationDa
ta);

            string rutaDb = System.IO.Path.Combine(folder, BaseDeDatos.bdName);
            var db = new SQLiteConnection(rutaDb);

            int Mild = int.Parse(Id.Text);
            //DisplayAlert("ID", "" + Mild, "Ok");

            var registro = new Cliente
            {
                Id = Mild,
                Nombre = nombre.Text,
                Domicilio = domicilio.Text,
                Telefono = int.Parse(telefono.Text),
                Correo = correo.Text,
                Foto = foto.Text
            };

```

```

        DisplayAlert("ID", Id.Text, "OK");

        db.Table<Cliente>();

        db.Update(registro);
        // DisplayAlert("Anuncio", "Registro Actualizado!!", "ok");

        Application.Current.MainPage.Navigation.PopAsync();
    }

    async private void tbiDelete_Clicked(object sender, EventArgs e)
    {
        string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.LocalApplicationDa
ta);
        string rutaDb = System.IO.Path.Combine(folder, BaseDeDatos.bdName);
        var db = new SQLiteConnection(rutaDb);

        int Mild = int.Parse(Id.Text);

        var respuesta = await DisplayAlert("¡Alerta!", "¿Esta seguro de que lo desea
eliminar?", "Si", "No");

        if (respuesta == true)
        {
            db.Delete<Cliente>(Mild);
            await Application.Current.MainPage.Navigation.PopAsync();
        }
    }
}

```

Ventas

Capturas de pantalla de la interfaz

Clase Venta.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;

```

```

using SQLite;
using Java.Sql;

namespace examen
{
    class Venta
    {
        [PrimaryKey]
        [AutoIncrement]
        public int Id { get; set; }
        public string fecha { get; set; }
        public double precioTotal { get; set; }
        public string cliente { get; set; }
        public string producto { get; set; }
        public int cantidad { get; set; }
    }
}

```

Ventas.xaml

```

<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="examen.Ventas"
    Title="Ventas"
    IconImageSource="bill.png"
    BackgroundColor="White"
    >

    <ContentPage.ToolbarItems>
        <ToolbarItem Name="tbiAgregar"
            Order="Primary"
            Icon="plus.png"
            Text="Agregar"
            Priority="0"
            Clicked="tbiAgregar_Clicked"
            />
    </ContentPage.ToolbarItems>

    <ContentPage.Content>

        <StackLayout>

```

```

        <ListView x:Name="lvVentas" HasUnevenRows="True"
Margin="15,5,15,15" ItemTapped="lvVentas_ItemTapped">
            <ListView.ItemTemplate>
                <DataTemplate>
                    <ViewCell>
                        <Frame BorderColor="#ccc"
BackgroundColor="White" Margin="5" CornerRadius="5">
                            <StackLayout Orientation="Horizontal">
                                <StackLayout
Orientation="Vertical">
                                    <Label x:Name="Id"
Text="{Binding Id}" IsVisible="False"/>
                                    <Label Text="Fecha"
FontSize="Micro" FontAttributes="Bold" TextColor="Black"/>
                                    <Label Text="{Binding fecha}"
TextColor="Salmon" FontSize="Small"></Label>
                                    <Label Text="Precio total"
FontSize="Micro" FontAttributes="Bold" TextColor="Black"/>
                                    <Label Text="{Binding
precioTotal}" FontSize="Large" TextColor="Salmon"></Label>
                                    <Label Text="Cliente"
FontSize="Micro" FontAttributes="Bold" TextColor="Black"/>
                                    <Label Text="{Binding cliente}"
TextColor="Salmon"></Label>
                                    <Label Text="Producto"
FontSize="Micro" FontAttributes="Bold" TextColor="Black"/>
                                    <Label Text="{Binding
producto}" TextColor="Salmon" FontSize="Small"></Label>
                                    <Label Text="Cantidad"
FontSize="Micro" FontAttributes="Bold" TextColor="Black"/>
                                    <Label Text="{Binding
cantidad}" TextColor="Salmon" FontSize="Small"></Label>
                                </StackLayout>
                            </StackLayout>
                        </Frame>
                    </ViewCell>
                </DataTemplate>
            </ListView.ItemTemplate>

```



```

        </ListView>

    </StackLayout>
</ContentPage.Content>
</ContentPage>

```

Ventas.xaml.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;
using SQLite;

namespace examen
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class Ventas : ContentPage
    {
        public void AbrirBase()
        {
            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
            string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);

            // Crea la base de datos si no existe, y crea una conexión
            var db = new SQLiteConnection(rutaDb);
            db.CreateTable<Venta>();
            var ventas = db.Table<Venta>().ToList();

            lvVentas.ItemsSource = null;
            lvVentas.ItemsSource = ventas;
        }

        public Ventas()
        {
            InitializeComponent();

```

```

        lvVentas.IsRefreshing = true;
        this.Appearing += Ventas_Appearing;
    }

    private void Ventas_Appearing(object sender, EventArgs e)
    {
        AbrirBase();
    }

    // Agregar venta
    async private void tbiAgregar_Clicked(object sender, EventArgs
e)
    {
        string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
        string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);

        // Crea la base de datos si no existe, y crea una conexión
        var db = new SQLiteConnection(rutaDb);
        db.CreateTable<Producto>();
        var Productos = db.Table<Producto>().ToList();

        var ventaPage = new AgregarVenta();
        ventaPage.BindingContext = Productos;

        await Navigation.PushAsync(ventaPage);
        AbrirBase();
    }

    private void lvVentas_ItemTapped(object sender,
ItemTappedEventArgs e)
    {
    }
}
}

```

AgregarVenta.xaml

```

<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

```

```

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
x:Class="examen.AgregarVenta">

<ContentPage.ToolbarItems>
    <ToolbarItem Name="tbiGuardar" Order="Primary" Icon="save.png"
Text="Guardar" Priority="0" Clicked="tbiGuardar_Clicked" />
</ContentPage.ToolbarItems>

<ContentPage.Content>
    <ScrollView>
        <StackLayout Margin="35,15,15,15">
            <Label Text="Fecha" FontSize="Medium"/>
            <DatePicker x:Name="fecha" Visual="Material"
IsEnabled="False"/>

            <Label Text="Cliente" FontSize="Medium" />
            <Picker x:Name="pkrClientes" Title="Seleccionar
cliente" SelectedIndexChanged="pkrClientes_SelectedIndexChanged"/>

            <Label Text="Producto" FontSize="Medium" />
            <Picker x:Name="pkrProductos" Title="Seleccionar
producto" SelectedIndexChanged="pkrProductos_SelectedIndexChanged"/>

            <Label Text="Cantidad" FontSize="Medium" />
            <Entry x:Name="cantidad" Keyboard="Numeric"
TextChanged="cantidad_TextChanged"/>

            <Label Text="Precio de total" FontSize="Medium" />
            <Entry x:Name="preciototal" Keyboard="Numeric"
IsEnabled="False" FontSize="Large" TextColor="Salmon"/>
        </StackLayout>
    </ScrollView>
</ContentPage.Content>
</ContentPage>

```

AgregarVenta.xaml.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;

```

```

using Xamarin.Forms.Xaml;
using SQLite;

namespace examen
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class AgregarVenta : ContentPage
    {
        int selectedIndex = -1;
        double cantidadProducto = 1;
        string clienteNombre = "";
        string productoNombre = "";
        int idProducto = -1;
        int existenciasActuales = -1;
        string NombreActualProducto = "";
        double PreciodeVentaActualProducto = 0;
        double PreciodeCompraActualProducto = 0;
        string FotoActualProducto = "";

        public void AbrirBase()
        {
            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
            string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);

            // Crea la base de datos si no existe, y crea una conexión
            var db = new SQLiteConnection(rutaDb);

            //var result = db.Query<Producto>("SELECT Nombre from
Producto");

            // Jalar PRODUCTOS
            db.CreateTable<Producto>();
            var results = db.Table<Producto>().ToList();

            var products = new List<string>();

            foreach (var producto in results)
            {
                if (producto != null)
                    products.Add(producto.Nombre.ToString());
            }
        }
    }
}

```

```

    }

    pkrProductos.ItemsSource = products;

    // Jalar CLIENTES
    var db2 = new SQLiteConnection(rutaDb);

    db2.CreateTable<Cliente>();
    var results2 = db2.Table<Cliente>().ToList();

    var clients = new List<string>();

    foreach (var client in results2)
    {
        if (client != null)
            clients.Add(client.Nombre.ToString());
    }

    pkrClientes.ItemsSource = clients;
}

public AgregarVenta()
{
    InitializeComponent();
    cantidad.Text = "1";
    preciotal.Text = "0";
    this.Appearing += AgregarVenta_Appearing;
}

private void AgregarVenta_Appearing(object sender, EventArgs e)
{
    AbrirBase();
}

private bool addClient()
{
    string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
    string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);
    // DisplayAlert("Ruta de la base de datos", rutaDb, "ok");

```

```

// Crea la base de datos si no existe, y crea una conexión
var db = new SQLiteConnection(rutaDb);

// Crea la tabla si no existe
db.CreateTable<Venta>();

if (string.IsNullOrEmpty(clienteNombre) ||
    string.IsNullOrEmpty(productoNombre) ||
    string.IsNullOrEmpty(cantidad.Text) ||
    string.IsNullOrEmpty(preciototal.Text) ||

string.IsNullOrEmpty(fecha.Date.ToShortDateString()))
{
    DisplayAlert("Error al agregar", "Todos los campos son
obligatorios", "OK");
    return false;
}

var registro = new Venta
{
    fecha = fecha.Date.ToShortDateString(),
    cliente = clienteNombre,
    producto = productoNombre,
    cantidad = int.Parse(cantidad.Text),
    precioTotal = double.Parse(preciototal.Text)
};

db.Insert(registro);
DisplayAlert("Registro agregado", "El registro fue agregado
con exito!", "OK");
return true;
}

private void tbiGuardar_Clicked(object sender, EventArgs e)
{
    if (addClient())
    {
        updateExistencias();
        Application.Current.MainPage.Navigation.PopAsync();
    }
}

```

```

        private void updateExistencias()
        {
            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
            string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);
            var db = new SQLiteConnection(rutaDb);

            if (string.IsNullOrEmpty(clienteNombre) ||
                string.IsNullOrEmpty(productoNombre) ||
                string.IsNullOrEmpty(cantidad.Text) ||
                string.IsNullOrEmpty(preciototal.Text) ||

string.IsNullOrEmpty(fecha.Date.ToShortDateString()))
            {
                DisplayAlert("Error al agregar", "Todos los campos son
obligatorios", "OK");
                return;
            }

            var registro = new Producto
            {
                Id = idProducto,
                Nombre = NombreActualProducto,
                PreciodeVenta = PreciodeVentaActualProducto,
                Cantidad = existenciasActuales -
int.Parse(cantidad.Text),
                PreciodeCompra = PreciodeCompraActualProducto,
                Foto = FotoActualProducto
            };

            db.Table<Producto>();

            db.Update(registro);
            // DisplayAlert("Anuncio", "Registro Actualizado!!", "ok");
        }

        private void pkrProductos_SelectedIndexChanged(object sender,
EventArgs e)
        {
            var picker = (Picker)sender;
            selectedIndex = picker.SelectedIndex;
        }
    }

```

```

        if (selectedIndex != -1)
        {

            productoNombre =
(string)picker.ItemsSource[selectedIndex];

            string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
            string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);

            // Crea la base de datos si no existe, y crea una
conexión

            var db = new SQLiteConnection(rutaDb);

            //var result = db.Query<Producto>("SELECT Nombre from
Producto");

            // Jalar PRODUCTOS
            db.CreateTable<Producto>();
            var results = db.Table<Producto>().ToList();

            //var products = new List<string>();

            //Obtener precio total
            var precioProducto =
double.Parse(results[selectedIndex].PreciodeVenta.ToString());
            idProducto =
int.Parse(results[selectedIndex].Id.ToString());
            existenciasActuales =
int.Parse(results[selectedIndex].Cantidad.ToString());
            NombreActualProducto =
results[selectedIndex].Nombre.ToString();
            FotoActualProducto =
results[selectedIndex].Foto.ToString();
            PreciodeVentaActualProducto =
double.Parse(results[selectedIndex].PreciodeVenta.ToString());
            PreciodeCompraActualProducto =
double.Parse(results[selectedIndex].PreciodeCompra.ToString());

```



```

        if (string.IsNullOrEmpty(cantidad.Text) ||
string.IsNullOrEmpty(cantidad.Text))
        {
            cantidadProducto = 1;
        }
        else
        {
            cantidadProducto = int.Parse(cantidad.Text);
        }
        preciototal.Text = Convert.ToString(precioProducto *
cantidadProducto);
    }
}

private void cantidad_TextChanged(object sender,
TextChangedEventArgs e)
{
    if (selectedIndex != -1)
    {
        string folder =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.Local
ApplicationData);
        string rutaDb = System.IO.Path.Combine(folder,
BaseDeDatos.bdName);

        // Crea la base de datos si no existe, y crea una
conexión

        var db = new SQLiteConnection(rutaDb);

        //var result = db.Query<Producto>("SELECT Nombre from
Producto");

        // Jalar PRODUCTOS
        db.CreateTable<Producto>();
        var results = db.Table<Producto>().ToList();

        //var products = new List<string>();

        //Obtener precio total
        var precioProducto =
double.Parse(results[selectedIndex].PreciodeVenta.ToString());
        if (string.IsNullOrEmpty(cantidad.Text) ||
string.IsNullOrEmpty(cantidad.Text))

```

```

        {
            return;
        }

        cantidadProducto = double.Parse(cantidad.Text);
        preciototal.Text = Convert.ToString(precioProducto *
cantidadProducto);
    }
}

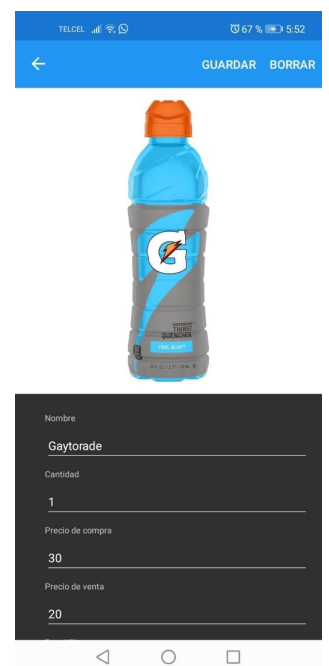
private void pkrClientes_SelectedIndexChanged(object sender,
EventArgs e)
{
    var picker = (Picker)sender;
    selectedIndex = picker.SelectedIndex;

    if (selectedIndex != -1)
    {
        clienteNombre =
(string)picker.ItemsSource[selectedIndex];
    }
}
}
}

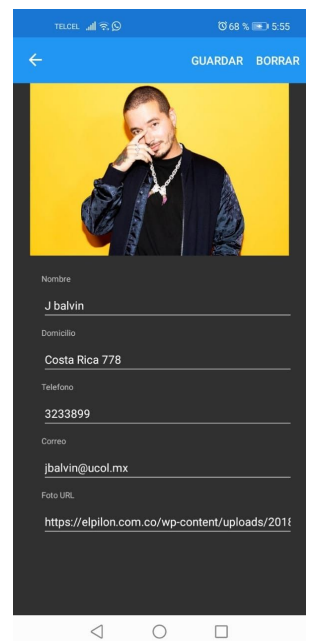
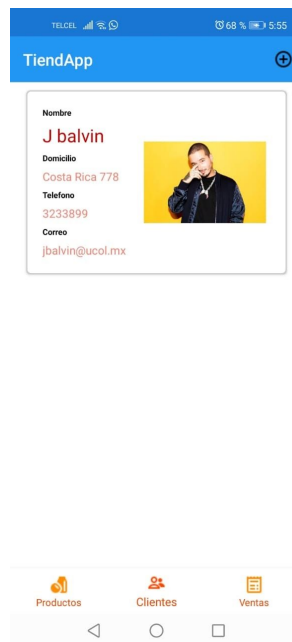
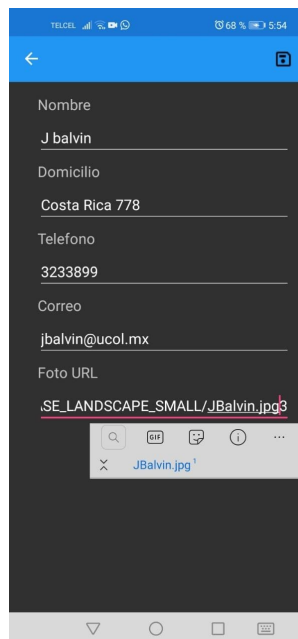
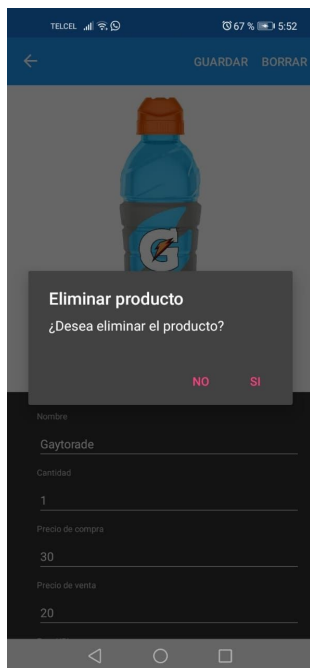
```

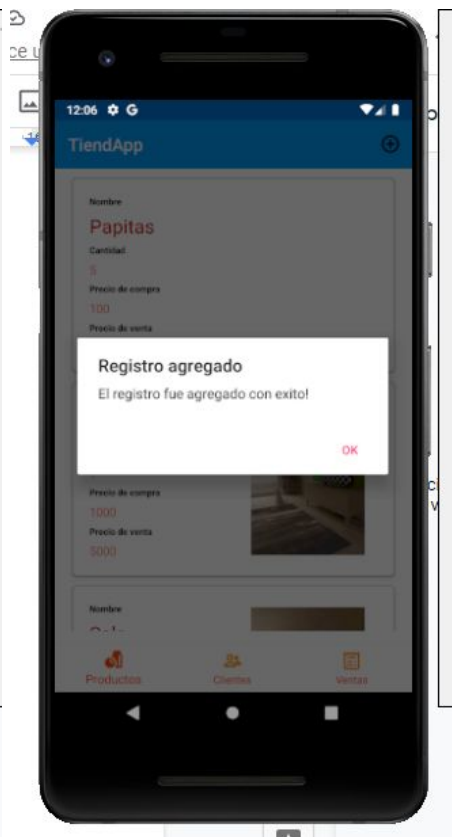
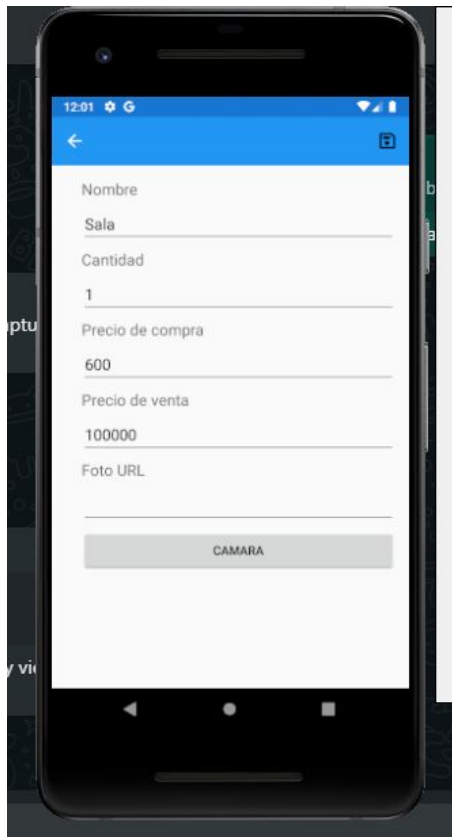
Capturas de pantalla de la interfaz

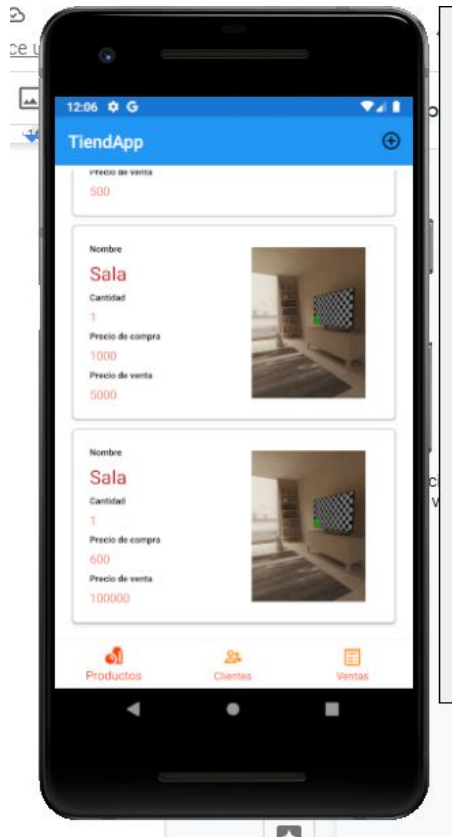
Producto



Cliente







Ventas

TELCEL 69 % 5:56

←

Fecha
28/10/2020

Cliente
J balvin

Producto

Seleccionar producto

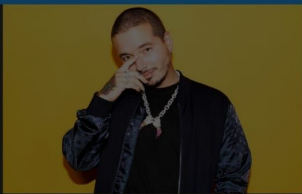
Gaytorade

CANCELAR

0

TELCEL 68 % 5:55

← GUARDAR BORRAR



¡Alerta!

¿Esta seguro de que lo desea eliminar?

NO SI

3233899

Correo
jbalvin@ucol.mx

Foto URL
<https://elpilon.com.co/wp-content/uploads/2016/10/jbalvin.jpg>

←

Fecha
28/10/2020

Cliente
Seleccionar cliente

Producto
Seleccionar producto

Cantidad
1

Precio de total
0

