

**Universidad Tecnológica de Santiago (UTESA)**

**Facultad De Arquitectura e Ingeniería**



**Asignatura:**

**SISTEMA Y TECNOLOGIA DE INFORMACION**

**001**

**Presentado a:**

**Tomás F. Ramírez Pérez**

**Presentado por:**

**Oscar Duvernay      2-18-0818**

**Tema:**

**Proyecto Final**

**Santiago, Rep. Dom**

**15/8/2023**

## Introducción al Código Fuente de la Aplicación

Presentaremos el código fuente que forma el núcleo de nuestra aplicación desarrollada en Power Apps. A lo largo de esta sección, destacaremos ejemplos de código que abarcan desde la autenticación de usuarios y la gestión de productos hasta el proceso de compra y la interacción con la base de datos. Este código ha sido diseñado para ofrecer una experiencia de compra en línea fluida y segura para los usuarios, mientras brinda a los propietarios de la tienda herramientas para administrar y mejorar su negocio en línea. Cada fragmento de código se explica detalladamente, y se proporcionan ejemplos de cómo se integra en la aplicación. Desde el registro y autenticación de usuarios hasta la generación de informes de ventas y la gestión del carrito de compras, esta sección te llevará a través de los procesos clave que hacen que nuestra aplicación sea eficiente y efectiva. Esperamos que esta visión general del código fuente te brinde una comprensión sólida de cómo se construye y opera nuestra aplicación en Power Apps.

Código para algunas de las funcionalidades de tu aplicación de Power Apps:

## Registro y Autenticación de Usuarios

// Código para el botón de registro

```
Patch(  
    LoginUsuario,  
    Defaults(LoginUsuario),  
    {  
        Nombre: Text(Nombre1.Text),  
        Usuario: Text(Usuario1.Text),  
        Contraseña: Text(Contraseña1.Text),  
        Direccion: Text(Direccion1.Text)  
    }  
);  
Navigate(RegistroExitosoScreen);
```

// Código para el botón de inicio de sesión

```
If(  

```

```
!IsBlank(Usuario1.Text) && !IsBlank(Contraseña1.Text),  
If(  
    !IsBlank(LookUp(LoginUsuario, Usuario = Usuario1.Text &&  
Contraseña = Contraseña1.Text)),  
    Navigate(InicioExitosoScreen),  
    Notify("Credenciales incorrectas", NotificationType.Error)  
),  
    Notify("Ingrese el usuario y contraseña",  
NotificationType.Error)  
)
```

## Agregar Producto al Carrito

// Código para el botón "Agregar al Carrito"

```
Patch(  
    Carrito,  
    Defaults(Carrito),  
    {  
        IdProducto: DataCardValue2.Text,  
        Precio: Value(DataCardValue19.Text)  
    }  
);  
Notify("Producto agregado al carrito", NotificationType.Success);
```

## Transferir a Comprado

// Código para el botón "Comprar"

```
ForAll(  
    Compra,  
    Collect(  
        Comprado,  
        {  
            Producto: compra,  
            Cantidad: Cantidad,  
            PrecioTotal: preciototal  
        }  
    )  
);  
ClearCollect(Compra, {});  
Navigate(ProductosVentaScreen);
```

## Enviar Sugerencia

// Código para el botón de enviar sugerencia

```
Patch(  
  Sugerencia,  
  Defaults(Sugerencia),  
  {  
    Correo: TextInput4.Text,  
    ASUNTO: TextInput5.Text,  
    MENSAJE: TextInput6.Text  
  }  
);  
Notify("Sugerencia enviada con éxito",  
  NotificationType.Success);  
Navigate(ExceletSugerenciaScreen);
```

## Transferir a Compra

// Código para el botón de transferir a Compra

```
ClearCollect(  
  Compra,  
  AddColumns(  
    Carrito,  
    "Producto", IdProducto,  
    "PrecioTotal", Precio
```

```
)  
);  
Clear(Carrito);  
Navigate(ComprarScreen);
```

## Transferir a Comprado

// Código para el botón de transferir a Comprado

```
ForAll(  
    Compra,  
    Collect(  
        Comprado,  
        {  
            Producto: compra,  
            Cantidad: Cantidad,  
            PrecioTotal: preciototal  
        }  
    )  
);  
ClearCollect(Compra, {});  
Navigate(ProductosVentaScreen);
```

## Agregar Producto al Carrito

### // Código para el botón "Agregar al Carrito"

```
Patch(  
    Carrito,  
    Defaults(Carrito),  
    {  
        IdProducto: DataCardValue2.Text,  
        Precio: Value(DataCardValue19.Text),  
        Cantidad: 1 // Por ejemplo, se agrega una unidad del  
producto  
    }  
);  
Notify("Producto agregado al carrito", NotificationType.Success);
```

## Proceso de Pago

### // Código para el botón de pago

```
SubmitForm(FormularioPago);  
If(FormularioPago.Valid,  
    ClearCollect(
```



```
Compra,
AddColumns(
    Carrito,
    "Producto", IdProducto,
    "PrecioTotal", Precio * Cantidad // Precio total por la
cantidad en el carrito
)
);
Clear(Carrito);
Navigate(ConfirmacionCompraScreen);
);
```

## Transferir a Compra y Limpiar Carrito

// Código para el botón de compra

```
ForAll(
    Carrito,
    Collect(
        Compra,
        {
            Producto: IdProducto,
            Cantidad: Cantidad,
            PrecioTotal: Precio * Cantidad
        }
    )
);
```

```
)  
);  
Clear(Carrito);  
Navigate(CompraExitosaScreen);
```

## Transferir a Rechazado

// Código para el botón de rechazar

```
ForAll(  
    Compra,  
    Collect(  
        Rechazado,  
        {  
            producto: Producto,  
            cantidad: Cantidad,  
            preciototal: PrecioTotal  
        }  
    )  
);  
Clear(Compra);  
Notify("Productos rechazados agregados",  
    NotificationType.Success);  
Navigate(RechazoExitosoScreen);
```

## Transferir a Comprado con Detalles

// Código para el botón "Comprar" con detalles

```
ForAll(  
  Compra,  
  Collect(  
    Comprado,  
    {  
      Producto: compra.Producto,  
      Cantidad: compra.Cantidad,  
      PrecioTotal: compra.PrecioTotal  
    }  
  )  
);  
ClearCollect(Compra, {});  
Navigate(ProductosVentaScreen);
```

## Borrar Carrito

// Código para el botón "Vaciar Carrito"

```
Clear(Carrito);
```

```
Notify("Carrito vaciado", NotificationType.Information);
```

## Ver Detalles del Producto

// Código para el botón "Detalles" en la lista de productos

```
Navigate(DetallesProductoScreen, Transition.None);
```

## Notificación al Agregar al Carrito

// Código para el botón "Agregar al Carrito" con notificación

```
Patch(
```

```
    Carrito,
```

```
    Defaults(Carrito),
```

```
{
```

```
    IdProducto: DataCardValue2.Text,
```

```
    Precio: Value(DataCardValue19.Text),
```

```
    Cantidad: 1
```

```
}
```

```
);
```

```
Notify("Producto agregado al carrito", NotificationType.Success);
```

## Notificación al Comprar

// Código para el botón de compra con notificación

```
ForAll(  
    Carrito,  
    Collect(  
        Compra,  
        {  
            Producto: IdProducto,  
            Cantidad: Cantidad,  
            PrecioTotal: Precio * Cantidad  
        }  
    )  
);  
Clear(Carrito);  
Notify("¡Compra realizada con éxito!", NotificationType.Success);  
Navigate(CompraExitosaScreen);
```

# BASE DE DATOS

Para esta Base de datos se usó Sharepoint, una muy buena utilidad en uso de Datos:

L

LoginProyectos

Bloc de notas

Páginas

LoginUsuario

Productos

Carrito

Compra

Sugerencia

Comprado

Rechazado

Contenido del sitio

Papelera de reciclaje

+ Nuevo

Editar en vista de cuadrícula

Compartir

Exportar

...

LoginUsuario

Nombre	Usuario	Contraseña	Direccion
Oscar Duvernay	Oscar	123	Moca

Bloc de notas

Páginas

LoginUsuario

Productos

Carrito

Compra

Sugerencia

Comprado

Rechazado

Contenido del sitio

Papelera de reciclaje

+ Nuevo

Editar en vista de cuadrícula

Compartir

Exportar

...

Productos

Nombre	Descripcion	Precio	Stock
iphone11	Producto	\$400,00	10
Apple SE 2023	Producto	\$339,00	3
Cubo Rubik	Objeto de Estrategia	\$20,00	0
Dell I4 20GB	Pc de Alta Gama	\$500,00	7

Bloc de notas

Páginas

LoginUsuario

Productos

Carrito

Compra

Sugerencia

Comprado

+ Nuevo

 Editar en vista de cuadrícula

 Compartir

 Exportar

## Compra

Producto 	Cantidad 	PrecioTotal 
iphone	1	\$400