



PRÁCTICA .NET: TIENDA VIRTUAL

NET- Backend con Tecnologías Propietarias



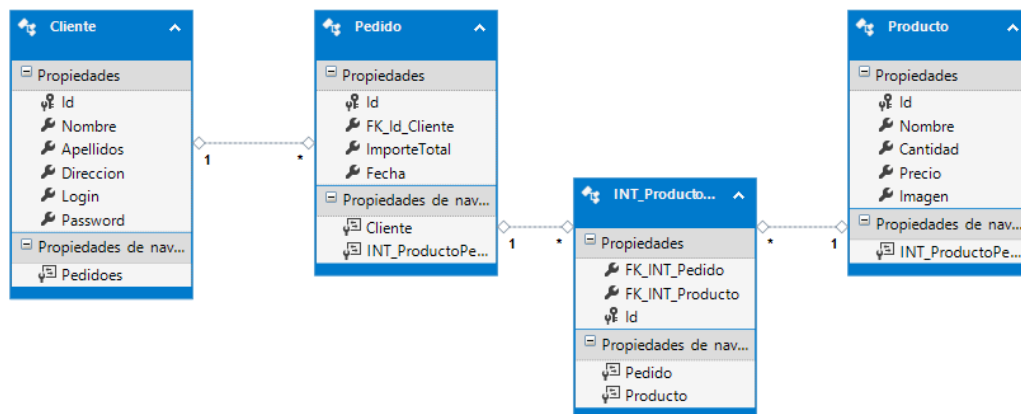
ALBERTO JIMÉNEZ DE LOS GALANES MIRALLES

5 DE MARZO DE 2018

Máster en Ingeniería Web - UPM

1.-Descripción sobre la implementación

La tienda virtual desarrollada se apoya en un archivo de base de datos de Microsoft SQL server llamado "**TiendaAlberto.mdf**", contenido dentro del proyecto, concretamente en la carpeta "**App_Data**". De esta manera se asegura la persistencia de los datos. El modelo de datos que se ha empleado para la tienda virtual es el siguiente:



Compuesto por una serie de entidades y relaciones entre tablas. Donde para modelar la relación entre las tablas **Pedido** y **Producto** (N:M) se optó por una tabla intermedia (**INT_Producto_Pedido**). Esto tiene algunas implicaciones a la hora de realizar ciertas acciones en nuestra tienda virtual. Como puede ser borrar un pedido o borrar un producto o incluso borrar un cliente. Es decir, por ejemplo, para borrar un pedido, se debería borrar primero todos los registros de la tabla intermedia (**INT_Producto_Pedido**) donde se guarda un registro para cada producto del pedido a borrar, y finalmente borrar el pedido que deseábamos. Igualmente, para borrar un cliente (que ya haya realizado al menos un pedido) se deberían borrar primero todos los pedidos de ese cliente, siguiendo el paso anteriormente descrito, y por último el cliente deseado. También, si se desea borrar un producto (contenido posiblemente en n pedidos), se deberían borrar primero todos los registros de la tabla intermedia, después los pedidos y finalmente el producto deseado.

Normalmente, para evitar hacer estos borrados, se toma la decisión de que, en cada tabla de cada una de estas entidades, se crea un campo nuevo denominado "ACTIVADO". Dicho campo de valor booleano, se establece a "FALSE", para evitar romper la integridad referencial. De tal forma que realmente no se elimina ningún registro, y no se tienen en cuenta dichos registros a la hora de consultar la base de datos. No obstante, se ha eliminado de la interfaz gráfica la opción de borrar pedidos, y se deja a criterio del usuario administrador borrar productos y clientes, siempre que no estén referenciados como hemos comentado. Se podrá borrar un producto que no haya sido comprado por nadie y se podrá eliminar un cliente que no haya realizado ningún pedido, en caso contrario se redirigirá a una vista de error.

Respecto a nuestra lógica de negocio está implementada basándonos en la forma de trabajar de .NET, concretamente en el lenguaje de programación C#, siguiendo el patrón modelo vista controlador (MVC), apoyándonos además en un Entity Framework para optimizar nuestra labor de desarrollo. Dicha lógica de negocio está formada por una serie de controladores autogenerados en cierta medida por el IDE de Microsoft, Visual Studio. Cada controlador cuenta con una vista asociada a él, ya que cada vez que se ejecuta una acción del controlador, se visualiza la vista asociada a este.

El proyecto está formado también por una serie de vistas parciales, que son renderizadas en la interfaz gráfica de la tienda virtual cuando proceden. Cuyas vistas se han personalizado con diferentes estilos CSS para proveer a la aplicación un aspecto más profesional y usable.

Además, la aplicación cuenta con una API REST que devuelve JSON, documentación asociada a la API implementada, integración del proyecto con GitHub, el uso de sesiones de usuarios a través del ModelBinder para implementar el carrito de la compra, consultas LINQ sobre nuestra base de datos para realizar ciertas queries, protección de las rutas de la mayoría de los componentes de la aplicación y control de acceso de usuarios con login y password.

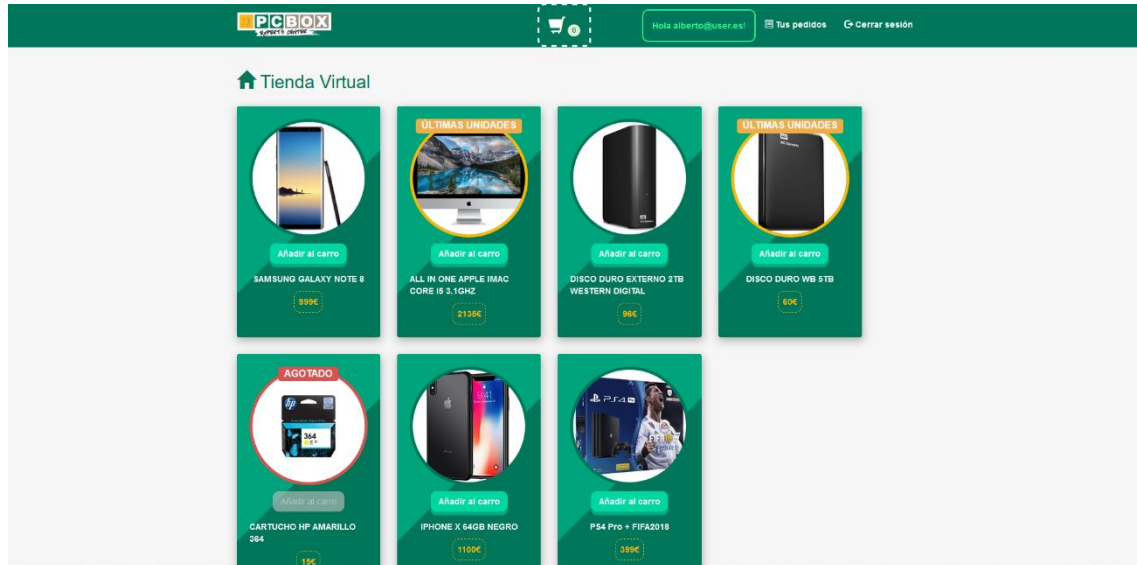
GitHub: <https://github.com/AlbertoNitro/TiendaVirtual-.Net>

La aplicación funcionalmente, se ha enfocado bajo dos puntos de vista. El primero de ellos, bajo el punto de vista de un usuario cliente de la tienda virtual. Desde el cual un cliente puede comprar productos y realizar un pedido, así como ver sus pedidos realizados anteriormente. El segundo punto de vista, teniendo en cuenta a un usuario administrador. Desde el cual se pueden gestionar tanto productos, pedidos y clientes, siempre respetando la integridad referencial de nuestra base de datos como hemos comentado más arriba. Al contemplar estos dos puntos de vista para dos usuarios diferentes de la aplicación se han implementado vistas diferentes para cada uno de ellos.

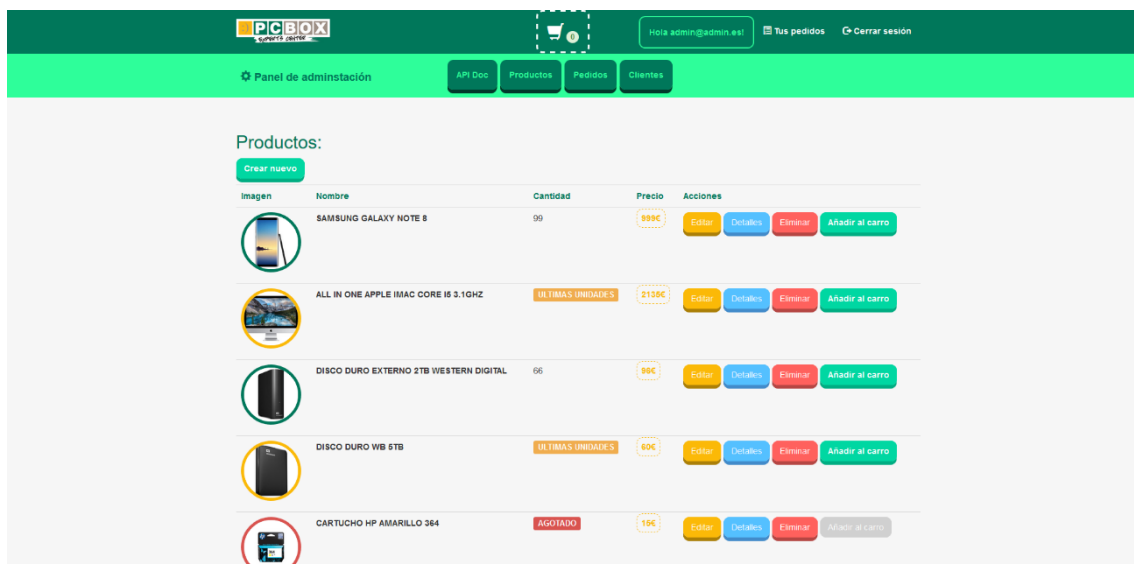
Para facilitar las pruebas sobre la tienda virtual, se puede acceder como usuario de esta con los siguientes clientes ya creados. También es posible crear nuevos usuarios clientes, no así de administrador.

Login	Password	
admin@admin.es	Admin.1234	Usuario Admin
luis@user.es	User.1234	
alberto@user.es	User.1234	Usuario Cliente
marina@user.es	User.1234	
ana@user.es	User.1234	
sergio@user.es	User.1234	

Vista de clientes:



Vista de administrador:



Finalmente recalcar que, si un producto está en stock de más de 10 unidades, su borde de imagen sera verde. Si su stock está entre 10 y 1 unidades, el borde de su imagen sera naranja junto con un aviso “ÚLTIMAS UNIDADES”. En último, caso si un producto no está en stock aparecera en rojo, mensaje “AGOTADO” y no se permitirá comprarlo mediante su botón deshabilitado.

