# Modulo: Soft.Services.Test.Catalog

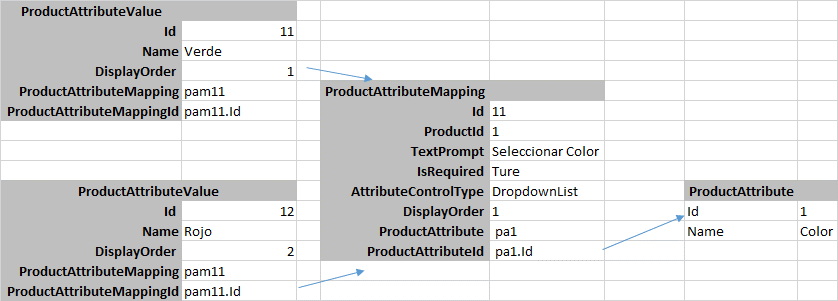
## Clase: ProductAttributeParserTests

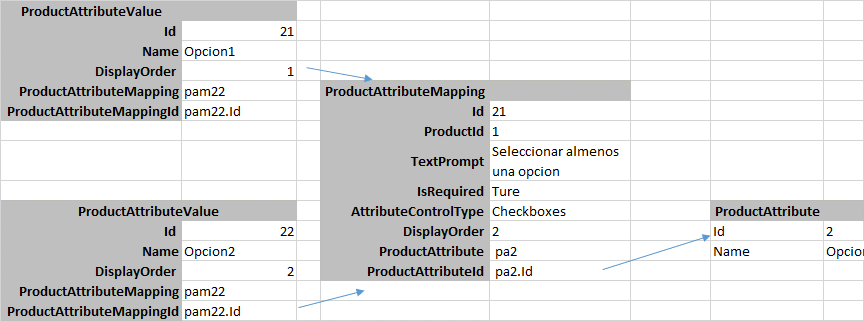
## Resumen

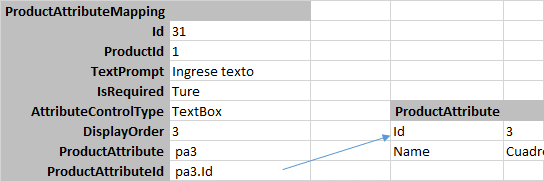
Se busca comprobar el correcto funcionamiento del servicio **PriceFormatterTests** el cual se encarga del manejo de las manedas

|  |  |
| --- | --- |
| Clase | PriceFormatterTests |
| Objetos | IRepository<ProductAttribute> \_productAttributeRepo;  IRepository<ProductAttributeMapping> \_productAttributeMappingRepo;  IRepository<ProductAttributeCombination> \_productAttributeCombinationRepo;  IRepository<ProductAttributeValue> \_productAttributeValueRepo;  IProductAttributeService \_productAttributeService;  IProductAttributeParser \_productAttributeParser;  IEventPublisher \_eventPublisher;  IWorkContext \_workContext;  ICurrencyService \_currencyService;  ILocalizationService \_localizationService;  ITaxService \_taxService;  IPriceFormatter \_priceFormatter;  IPriceCalculationService \_priceCalculationService;  IDownloadService \_downloadService;  IWebHelper \_webHelper;  ShoppingCartSettings \_shoppingCartSettings;  IProductAttributeFormatter \_productAttributeFormatter;  ProductAttribute pa1, pa2, pa3;  ProductAttributeMapping \_pam11, pam21, pam31;  ProductAttributeValue pav11, pav12, pav21, pav22; |
| Creaciones | * \_productAttributeRepo: Ingresamos los atributos de **Color, Option** y **Text** * \_productAttributeMappingRepo: Ingresar el mapping para el **Color** Y **Option** * \_productAttributeCombinationRepo: Se inserta una lisya vacia de ProductAttributeCombination * \_productAttributeValueRepo: Se inserta las opciones para el Mapping **Color** y **Option** * \_eventPublisher: publica el vento de tipo Arg<object> * cacheManager Contiene una cache nula * \_productAttributeService con los atributos:   cacheManager, \_productAttributeRepo, \_productAttributeMappingRepo, \_productAttributeCombinationRepo, \_productAttributeValueRepo,  \_eventPublisher   * \_productAttributeParser se le pasa como argumento al \_productAttributeService * \_priceCalculationService Instamcia * \_workContext con un currentLanguage actual * \_currencyService Instancia * \_localizationService con 4 recursos:   GiftCardAttribute.For.Virtual  GiftCardAttribute.From.Virtual  GiftCardAttribute.For. Physical  GiftCardAttribute.From. Physical   * \_taxService Instancia * \_priceFormatter Instancia * \_downloadService Instancia * \_webHelper Instamcia * \_shoppingCartSettings Instancia * \_productAttributeFormatter con los siguientes parametros   \_workContext, \_productAttributeService, \_productAttributeParser, \_currencyService, \_localizationService, \_taxService, \_priceFormatter, \_downloadService, \_webHelper, \_priceCalculationService, \_shoppingCartSettings |
| Post Condición | La instancia solo debe contener los valores por defectos creados **PriceCalculationService** y con una instancia de un producto de ejemplo creada para su prueba |

**Preparacion de la data**







# Caso de prueba: Poder agregar y parsear un atributo de un producto

**Escenario**:

|  |  |
| --- | --- |
|  |  |

**Prueba**:

\_productAttributeParser.AddProductAttribute

|  |  |  |  |
| --- | --- | --- | --- |
| attributesXml | productAttributeMapping | value | Retorno |
| “” |  | 11 |  |
|  |  | 21 |  |
|  |  | 22 |  |

# Caso de prueba: Poder agregar y parsear los atributos de un giftcard

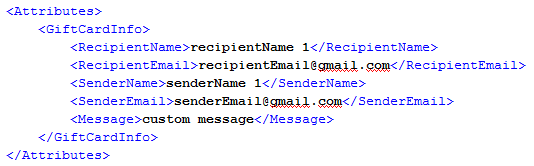
**Escenario**:

* Se crea un giftCard

**Prueba**:

\_productAttributeParser.AddGiftCardAttribute

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| attributesXml | recipientName | recipientEmail | senderName | senderEmail | giftCardMessage |
| “” | recipientName | recipientEmail@gmail.com | senderName 1 | senderEmail@gmail.com | custom message |



\_productAttributeParser.GetGiftCardAttribute

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| attributesXml | recipientName | recipientEmail | senderName | senderEmail | giftCardMessage |
|  | “” | “” | “” | “” | “” |

recipientName.ShouldEqual("recipientName 1");

recipientEmail.ShouldEqual("recipientEmail@gmail.com");

senderName.ShouldEqual("senderName 1");

senderEmail.ShouldEqual("senderEmail@gmail.com");

giftCardMessage.ShouldEqual("custom message");