1. Products

Variants:

Modelo para el nombre de la variante : [\_o.option\_id.name\_]

Separador del modelo para el nombre de la variante : "El texto que quieras como valor fijo, por ejemplo : REF "

Generador de código : [\_o.option\_id.code\_]

o bien..

[\_o.option\_id.name\_]

En este caso, si tienes una dimensión que sea "tamaño" y tienes varias dadas de alta y en el campo del codigo de la dimensión tienes esto :

100 x 20

100 x 25

100 x 30

y por otra parte, si tienes otra dimensión que se llame color con dos registros y en el campo codigo tienes esto :

v

a

te creará se productos con estos códigos :

REF 100 X 20 v

REF 100 X 25 v

REF 100 X 30 v

REF 100 X 20 a

REF 100 X 25 a

REF 100 X 30 a

Si cambias el valor del campo y pones [\_o.option\_id.name\_] te pondrá el valor del campo name de la dimensión, por ejemplo, en el caso del color v "verde" y a"Azul" si así lo has dado de alta.

Modules:

product\_variant\_multi, product\_variant\_multi\_advanced, product\_variant\_selection, sale\_product\_variant\_selection..

Q

Me gustaría saber si conocéis algún vertical de Retail para OpenERP.

He probado con el product\_variant para la creación de tallas y colores. Este módulo crea tantos productos cómo colores y tallas hay, pero no es capaz de generar un único productos en el cual pueda meter cantidades por colores y tallas. (Que es lo más intuitivo)

A.

Sobre esto, siento comunicarte que la única forma (lo estructures, como lo estructures, da igual) de gestionar el stock por tallas y colores, es teniendo un "producto/referencia" por cada talla color que es lo que hace product\_variant. Ahora bien, otra cosa es que des usabilidad al usuario y le montes una pantalla donde pueda realizar la compra/venta como dices tú... definiendo una referencia genérica (plantilla) e incluyendo cantidades por cada talla/color, en una especie de grid variable en función de los atributos definidos para esa plantilla.

Esto es cambiarle la vista al usuario, pero internamente, al final el sistema debe hacer una línea de pedido por cantidad producto/talla/color y realizar movimientos de stock, tanto de entrada como de salida para ese producto concreto a fin de controlar el stock de forma individual.

Es más, si en algún momento, una vez arrancado se te ocurre definir otro atributo... sería necesario "mover" el stock de un producto al nuevo y todas las referencias anteriores no te valdrían ya.

Montar las pantallas "amigables" para el usuario tanto en compras como en ventas tiene su aquel porque es un grid dinámico pero se puede hacer perfectamente.

Que hi ha per la V8:

OpenERP Version 8 includes a new way to handle product attributes, multi-level variants and a configurator for the eCommerce. The proposed solution is super simple but also extremely powerful as it handles nearly all business cases easily.

This talk is about how we redesigned attributes and variants in OpenERP version 8 and how you can use them in the eCommerce to handle product variants. Several issues have been fixed by this new approach:

Easier user interface:

no more template and variants menu, one single and efficient view: the product catalog

the menu/user interface does not change if you activate variants or not. Easier to understand for end-users.

You can easily mix products with or without variants

Better approach to views management: in version 7, we had views for Products (product.product) and templates (product.template). Every module should have inherited both object's views. This led to inconcistencies in product and template views. In our proposed solution, templates are the only object you need to define views.

Consolidation of several concepts into a single one:

product attributes

variants

multi-level variants

product configurator

manufacturing attributes

Improved eCommerce:

Product configurator

Multiple types of variants (selection box, color boxes, radio buttons)

Search by attributes (color: red)

Cleaning of fields between products and templates:

on which level is the photo field, the cost price, etc.

multi-variants for manufacturing support in your BoM

1. Presupuestos

Si hay que entrar lineas sin un producto claro, mejor definir un poducto genérico al que cambiar la descripción incluirlo en el presupuesto.

1. Localización Española

- Instalación:

sudo git clone https://github.com/OCA/l10n-spain.git