# Odoo 条码扫码功能 采购订单、销售订单通过扫码增加明细

2018年01月15日

很多人都说从9.0 之后，很多社区版功能被阉割了，比如大家常说的仓库条码扫码模块就没有了。 但是却为我们留下了bcarcode模块，方便我们进行扩展。

由于有需求，需要为采购模块增加条码扫码功能，代码如下：

1.需要在purchase.order.line 增加product\_barcode字段，关联自产品资料的bcarcode:

class PurchaseOrderLine(models.Model):

\_inherit = 'purchase.order.line'

product\_barcode = fields.Char(related='product\_id.barcode')

2.在purchase.order.line 增加一个方法on\_barcode\_scanned，获取扫描枪获取的条码,同时继承扩展barcodes.barcode\_events\_mixin,代码如下:

class PurchaseOrder(models.Model):

\_name = 'purchase.order'

\_inherit = ['purchase.order', 'barcodes.barcode\_events\_mixin']

def \_add\_product(self, product, qty=1.0):

order\_line = self.order\_line.filtered(lambda r: r.product\_id.id == product.id)

if order\_line:

order\_line.product\_qty = qty

else:

product\_lang = product.with\_context({

'lang': self.partner\_id.lang,

'partner\_id': self.partner\_id.id,

})

name = product\_lang.display\_name

if product\_lang.description\_purchase:

name += '\n' + product\_lang.description\_purchase

vals = {

'product\_id': product.id,

'name': name,

'date\_planned': fields.Datetime.now(),

'product\_uom': product.uom\_po\_id.id,

'product\_qty': 1,

'price\_unit': product.standard\_price,

'state': 'draft',

}

self.order\_line += self.order\_line.new(vals)

def on\_barcode\_scanned(self, barcode):

product = self.env['product.product'].search([('barcode', '=', barcode)])

if product:

self.\_add\_product(product)

3.增加js代码，使得前端获取条码：

odoo.define('purchase\_barcode.PurchaseOrderBarcodeHandler', function (require) {

"use strict";

var field\_registry = require('web.field\_registry');

var AbstractField = require('web.AbstractField');

var FormController = require('web.FormController');

FormController.include({

\_barcodePurchaseAddRecordId: function (barcode, activeBarcode) {

if (!activeBarcode.handle) {

return $.Deferred().reject();

}

var record = this.model.get(activeBarcode.handle);

if (record.data.state != 'draft') {

this.do\_warn("采购订单", '只能对草稿状态的单据增加明细');

return $.Deferred().reject();

}

return this.\_barcodeAddX2MQuantity(barcode, activeBarcode);

}

})

var PurchaseOrderBarcodeHandler = AbstractField.extend({

init: function () {

this.\_super.apply(this, arguments);

this.trigger\_up('activeBarcode', {

name: this.name,

fieldName: 'order\_line',

quantity: 'product\_qty',

setQuantityWithKeypress: true,

commands: {

// 'O-CMD.MAIN-MENU': \_.bind(this.do\_action, this, 'stock\_barcode.stock\_barcode\_action\_main\_menu', {clear\_breadcrumbs: true}),

barcode: '\_barcodePurchaseAddRecordId',

}

});

},

});

field\_registry.add('purchaseorder\_barcode\_handler', PurchaseOrderBarcodeHandler);

});

4.扩展purchase order view 部分：

<record id="view\_purchase\_order\_barcode\_inherit\_form" model="ir.ui.view">

<field name="name">purchase.order.form</field>

<field name="model">purchase.order</field>

<field name="inherit\_id" ref="purchase.purchase\_order\_form"/>

<field name="priority" eval="8"/>

<field name="arch" type="xml">

<form position="inside">

<field name="\_barcode\_scanned" widget="purchaseorder\_barcode\_handler"/>

</form>

<xpath expr="//field[@name='order\_line']/tree" position="inside">

<field name="product\_barcode" invisible="1"/>

</xpath>

<xpath expr="//field[@name='order\_line']/kanban" position="inside">

<field name="product\_barcode" invisible="1"/>

</xpath>

<xpath expr="//field[@name='order\_line']//field[@name='product\_qty']" position="attributes">

<attribute name="options">{'barcode\_events': True}</attribute>

<attribute name="widget">field\_float\_scannable</attribute>

</xpath>

<xpath expr="//button[@name='button\_confirm']" position="attributes">

<attribute name="barcode\_trigger">validate</attribute>

</xpath>

</field>

</record>

这样，就可以在界面上直接用扫描枪扫码增加订单明细了。

文章来自：https://www.odooqs.com/blog/1/post/odoo-4