

Verificación del correo

Este modulo lo que hace es enviar un código al cliente para verificar el correo a la hora de hacer una compra de un producto.

Autor

- [Peti soluciones productivas](#)

Modelos (herencia)

Se hereda del modelo "sale.order" donde se crearon 3 campos nuevos:

- **codigo:** Código de verificación que se envía por correo.
- **tiempovencimiento:** Campo de fecha con la cual se verifica que el tiempo transcurrido al ingresar el código en la compra este dentro de los 3 minutos disponibles.
- **verificado:** variable que identifica si el correo fue verificado o no.

```
class SaleOrderExtend(models.Model):
    _inherit = 'sale.order'

    codigo = fields.Integer("Codigo de verificación")
    tiempovencimiento = fields.Datetime("tiempo en que vence el codigo")
    verificado = fields.Boolean("Verificado", default=False)
```

También se crearon 3 nuevos metodos:

- **VerificarCodigo(codigo, fecha):** recibe dos variables codigo y fecha, el código es el que ingresa el usuario en pantalla y la fecha es la fecha en la que hace la petición, estos campos se comparan con los campos de la orden para verificar que el código ingresado sea igual al que tiene la orden y que la fecha sea menor a la fecha que se guarda en el campo tiempovencimiento en la orden.

```
def VerificarCodigo(self, codigo, fecha):
    if self.codigo == int(codigo) and fecha <= self.tiempovencimiento:
        self.verificado = True
    else:
        self.verificado = False
```

- **send_code(correo):** Está función envía el código al correo recibido como parámetro.

```
def send_code(self, correo):
    self.codigo = self.GenerarCodigo()
    self.tiempovencimiento = datetime.datetime.now() + timedelta(minutes=3)
    self.partner_id.email = correo
    ctx = {
        'codigo': str(self.codigo).zfill(6),
    }
    template = self.env.ref('doble_autenticacion.email_template_envio_codigo')
    template.sudo().with_context(ctx).send_mail(self.id, force_send=True)
```

- **GenerarCodigo():** Está función genera un código de 6 dígitos y lo retorna.

```
def GenerarCodigo(self):
    numero = random.randint(0, 999999)
```

```

if len(str(numero)) < 6:
    numero = str(numero).zfill(6)

return int(numero)

```

Se agrega un campo en "product.category" llamado servidor_de_correo en donde se guarda el servidor de correo saliente del cual Odoo va a enviar el código al cliente.

```

class ProductCategory(models.Model):
    _inherit = 'product.category'

    servidor_de_correo = fields.Many2one('ir.mail_server', 'Servidor de correo')

```

Controladores

Se crea un nuevo controlador de Odoo que recibe solicitudes HTTP POST a la ruta '/send/code' y devuelve una respuesta JSON.

El método send_code_mail() extrae el correo electrónico de la solicitud POST y obtiene el pedido actual utilizando el método request.website.sale_get_order(). Luego llama al método send_code() del objeto order para enviar un correo electrónico con un código de verificación al correo electrónico proporcionado.

Finalmente, se crea un diccionario data con una clave 'respuesta' que tiene como valor la cadena 'Correo enviado correctamente'. El diccionario se convierte en una cadena JSON utilizando el método json.dumps() y se devuelve como respuesta a la solicitud.

```

class WebsiteSaleExtended(http.Controller):
    @http.route(['/send/code'], type='json', auth="public", method=['POST'], website=True)
    def send_code_mail(self, **kwargs):
        order = request.website.sale_get_order()
        correo = kwargs.get('correo')
        order.send_code(correo)
        data = {'respuesta': 'Correo enviado correctamente'}
        return json.dumps(data)

```

Controlador de Odoo que recibe solicitudes HTTP POST a la ruta '/verificar' y devuelve una respuesta JSON.

El método verificar() extrae el correo electrónico de la solicitud POST y el código digitado por el cliente.

también obtiene el pedido actual utilizando el método request.website.sale_get_order(). Verifica que el correo ingresado sea igual al del cliente de la orden. Luego llama al método VerificarCodigo() del objeto order para verificar que el código ingresado coincida con el de la orden estas verificaciones se guardan en un diccionario que es retornado como tipo JSON.

```

@http.route(['/verificar'], type='json', auth="public", method=['POST'], website=True)
def verificar(self, **kwargs):
    order = request.website.sale_get_order()
    correo = kwargs.get('correo')
    codigo = kwargs.get('codigo')
    data = {}
    if not order.partner_id.email == correo:
        data['correo'] = 'Correo diferente'
    else:
        data['correo'] = 'Correo igual'

```

```

order.VerificarCodigo(codigo, datetime.datetime.now())
if order.verificado:
    data['respuesta'] = 'Correcto'
else:
    data['respuesta'] = 'Incorrecto'
return json.dumps(data)

```

Vistas

Se agrega a la vista `product.product_category_form_view` el nuevo campo `"servidor_de_correo"` del modelo `"product.category"`

```

<record id="product_category_inherit_form_view" model="ir.ui.view">
    <field name="name">product_category_inherit_form_view</field>
    <field name="model">product.category</field>
    <field name="inherit_id" ref="product.product_category_form_view"/>
    <field name="arch" type="xml">
        <xpath expr="//field[@name='buyer_view']" position="after">
            <field name="servidor_de_correo"/>
        </xpath>
    </field>
</record>

```

Se hereda del template `web_sale_extended.address` para agregar la modal donde se va a ingresar el código para verificar el correo.

```

<template id="address_inherit" inherit_id="web_sale_extended.address">
    <xpath expr="//form[@id='shop']" position="after">
        <div class="modal fade" id="codigo-verificacion-modal" role="dialog">
            <div class="modal-dialog">
                <div class="modal-content">
                    <div class="modal-header modal-header-primary py-3 rounded-top text-light">
                        <h5 class="modal-title" style="color: black;">Verificación del correo</h5>
                        <button type="button" class="close shadow-none" data-dismiss="modal">
                            <span aria-label="Close">&times;</span>
                        </button>
                    </div>
                    <div class="modal-body" style="background-color: white;">
                        <p style="color:black;">
                            Digite el código que fue enviado al correo ingresado en el formulario,
                            este código tiene una vigencia de 3 minutos.
                        <br/>
                        Si no recibio el código o el tiempo expiró puede reenviar el código con el siguiente
                        enlace
                        <t t-if="website_sale_order.main_product_id.categ_id.sponsor_id.id == 5111">
                            <a href="#" id="reenviar_codigo" style="color: #43b12e;">Reenviar código</a>
                        </t>
                        <t t-else="">
                            <a href="#" id="reenviar_codigo" style="color: #0837ba;">Reenviar código</a>
                        </t>
                    </p>
                    <br/>
                    <p id="reenvio" style="color: red;" class="o_hidden">Código reenviado!</p>
                    <br/>
                    <div id="input_code">
                        t-attf-class="form-group #{error.get('') and 'o_has_error' or ''} col-lg-6">
                        <label class="col-form-label" for="birthdate_date" style="color:black;">Código de

```



```

width: 100%;
height: 100%;
overflow: hidden;
outline: 0;
}
.modal-content-custom {
border-radius: 0.3rem;
}
.modal-footer-verificacion {
justify-content: end;
display: flex;
padding: 1rem;
}
.modal-dialog-custom {
max-width: 650px;
}
</style>
<div class="modal fade modal-custom" id="codigo-verificacion-modal" role="dialog">
<div class="modal-dialog modal-dialog-custom">
<div class="modal-content modal-content-custom">
<div class="modal-header modal-header-primary py-3 rounded-top text-light">
<h5 class="modal-title">Verificación del correo</h5>
<button type="button" class="close shadow-none" data-dismiss="modal">
<span aria-label="Close">&times;</span>
</button>
</div>
<div class="modal-body" style="background-color: white;">
<p style="color: black;">
Digite el código que fue enviado al correo ingresado en el formulario,
este código tiene una vigencia de 3 minutos.
<br/>
Si no recibio el código o el tiempo expiró puede reenviar el código con el siguiente
enlace
<a href="#" id="reenviar_codigo" style="color: #59CBE8;">Reenviar código</a>
</p>
<br/>
<p id="reenvio" style="color: red;" class="o_hidden">Código reenviado!</p>
<br/>
<div id="input_code"
t-attf-class="form-group #{error.get("") and 'o_has_error' or ''} col-lg-6">
<label class="col-form-label" for="birthdate_date" style="color: black;">Código de
verificación*
</label>
<input type="number" name="codigo_verificacion" id="codigo_verificacion"
style="background-color: white; border-color: black; color: black;"
tooltip="Ingrese el código que se envió al correo digitado"
t-attf-class="form-control #{error.get('codigo_verificacion') and 'is-invalid' or ''}"
t-att-value="'codigo_verificacion' in checkout and checkout['codigo_verificacion']"/>
<div id="div_warning_code" class="alert alert-warning" role="alert"
style="background-color: #fff3cd !important; border-color: #ffeeba !important; color:
#856404 !important; display: none;">
<div class="d-flex justify-content-end">
<button type="button" id="cerrar" style="border: none;">
x
</button>
</div>

```



```

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

$(function() {
  $("#reenviar_codigo").on('click', function(e){
    e.preventDefault()
    EnviarCodigo(document.querySelector("input[name='email']").value);
    $("#reenvio")[0].classList.remove("o_hidden");
  });

  $("#cerrar").on('click', function(e){
    $("#div_warning_code").hide();
  });

  $("#verificar").on('click', async function(e){
    e.preventDefault();
    $("#reenvio")[0].classList.add("o_hidden");
    if ($("#input[name='codigo_verificacion']")[0].value === ""){
      $("#div_warning_code").show();
    } else {
      if($("#shop").valid()){ //checks if it's valid
        $(this).html('<div><p class="preloader"><span class="spinner-border spinner-border-sm preloader"
role="status" aria-hidden="true" />Cargando...</div>');
        $(this).prop('disabled', true);
      }
      var correo = document.querySelector("input[name='email']").value;
      var codigo = document.querySelector("input[name='codigo_verificacion']").value;
      var data = {'correo': correo, 'codigo': codigo};
      let dic = await ajax.jsonRpc('/verificar', 'call', data)
        .then(function(data) {
          return data
        });
      let diccionario = JSON.parse(dic);
      if(diccionario.correo === 'Correo igual' && diccionario.respuesta === 'Correcto'){
        $("#shop").submit();
      } else if(diccionario.correo === 'Correo diferente' || diccionario.respuesta === 'Incorrecto'){
        $(this).html('<span>Verificar</span><i class="fa fa-chevron-right">');
        $(this).prop('disabled', false);
        $("#div_warning_code").show();
      }
    }
  });

  $("#submit_shop").on('click', function(e){
    e.preventDefault();
    if($("#shop").valid()){ //checks if it's valid
      $(this).html('<div><p class="preloader"><span class="spinner-border spinner-border-sm preloader"
role="status" aria-hidden="true" />Cargando...</div>');
      $(this).prop('disabled', true);
      EnviarCodigo(document.querySelector("input[name='email']").value);
      $("#reenvio")[0].classList.add("o_hidden");
    }
  });

  async function EnviarCodigo(correo){
    var data = {'correo': correo}
    await ajax.jsonRpc('/send/code', 'call', data)

```

```

        .then(function(data) {
            let decode_data = JSON.parse(data);
            if(decode_data['respuesta'] === 'Correo enviado correctamente'){
                $("#codigo-verificacion-modal").modal('show');
                $("#submit_shop").html('<span>Siguiente</span><i class="fa fa-chevron-right"/>');
                $("#submit_shop").prop('disabled', false);
            }
        });
    });
});

```

Configuración

Dirigirse a la categoría de un producto y seleccionar el servidor de correo saliente con el que se va a enviar el mail al cliente.

Al configurar la categoría, cualquier producto que tenga esta categoría el mail del código de verificación será enviado desde el correo saliente asignado a la categoría.