

Envío de Email

Envío de emails dende código

Incluimos un botón na vista **"informacion.xml"**, para chamar ao método **"envio_email"**, que definiremos no modelo **"informacion"**

```
<record model="ir.ui.view" id="informacion_form_view">
  <field name="name">informacion.form</field>
  <field name="model">odoo_basico.informacion</field>
  <field name="arch" type="xml">
    <form string="Formulario información">
      <header>
        <button name="ver_contexto" type="object" string="Ver Contexto"/>
        <button name="envio_email" type="object" string="Enviar Email"/>
      </header>
    </form>
  </field>
</record>
```

Menú Nivel 0

Menú Nivel 1

A acción de Información / Rexistro 9

Editar

Crear

Imprimir

Acción

Ver Contexto

Enviar Email

Localización da Data hora

Actualiza Hora a Timezone do Usuario

Actualiza Hora Actual a UTC

Datos persoais

Datos Numéricos

Titulo

A Descrición

¿Autorizado?

Sexo

Data

Mes Castelan

Mes Galego

Data e Hora

Hora UTC

Hora Timezone do Usuario

Hora Actual

Arquivo Adxunto

Foto

Rexistro 9

Descrición Rexistro 9

☒

Home

10/10/2020

mayo


Maio

07/05/2020 14:59:43

12:59:43

14:59:43

19:15:51



Foto

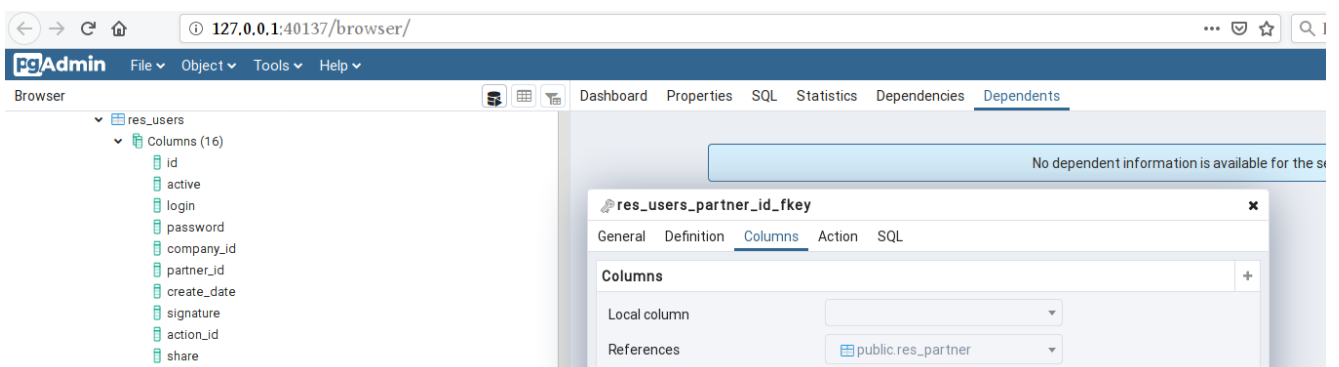
Código para enviar un email

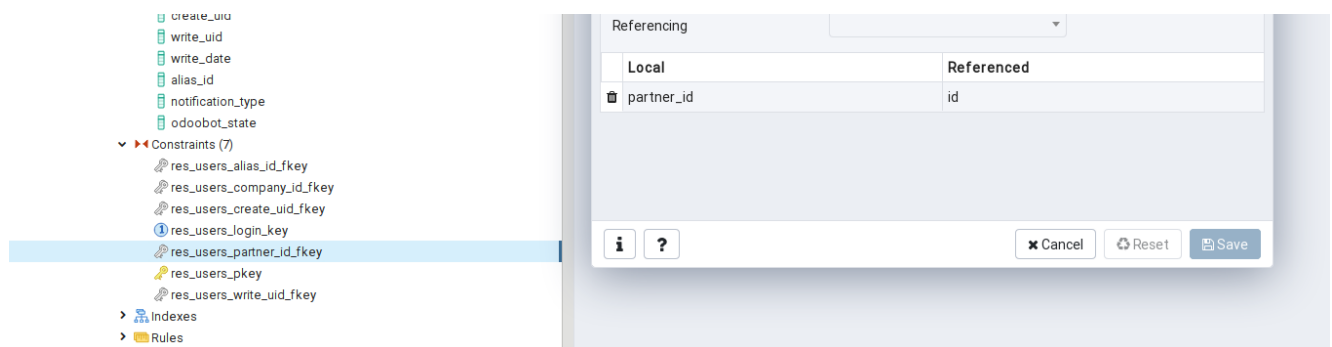
Definimos no arquivo **informacion.py** o método `envio_email` que chamabamos dende a vista "informacion.xml".

```
def envio_email(self):
    meu_usuario = self.env.user
    #mail_de      Odoon pon o email que configuramos en gmail para facer o envio
    mail_reply_to = meu_usuario.partner_id.email # o enderezo email que ten as
    mail_para = 'emaildedestino@servidordedestino.com' # o enderezo email de d
    mail_valores = {
        'subject': 'Aquí iría o asunto do email ',
        'author_id': meu_usuario.id,
        'email_from': mail_reply_to,
        'email_to': mail_para,
        'message_type': 'email',
        'body_html': 'Aquí iría o corpo do email cos datos por exemplo de "%s"
    }
    mail_id = self.env['mail.mail'].create(mail_valores)
    mail_id.sudo().send()
    return True
```

Podemos ver que mediante `self.env.user` obtemos o id do noso usuario en `res_users`.

```
meu_usuario = self.env.user
```





E mediante a foreign key definida en `res_users` navegamos para acceder ao valor do campo `email` na táboa `res_partner`.

```
mail_de = meu_usuario.partner_id.email # o enderezo email que ten asociado o noso u
```

Vemos que en **mail_valores** imos almacenando os distintos valores necesarios para o envío do email.

E que mediante **self** podemos acceder ao rexistro actual da táboa "odoo_basico_informacion". Así con **self.descripcion** obtemos o valor do campo `descripcion`.

A continuación, accedemos mediante **self.env** ao método **create** do modelo "mail" definido no módulo "mail".

Pasamos como parámetro a variable **mail_valores** que ten os datos necesarios para o envío do email.

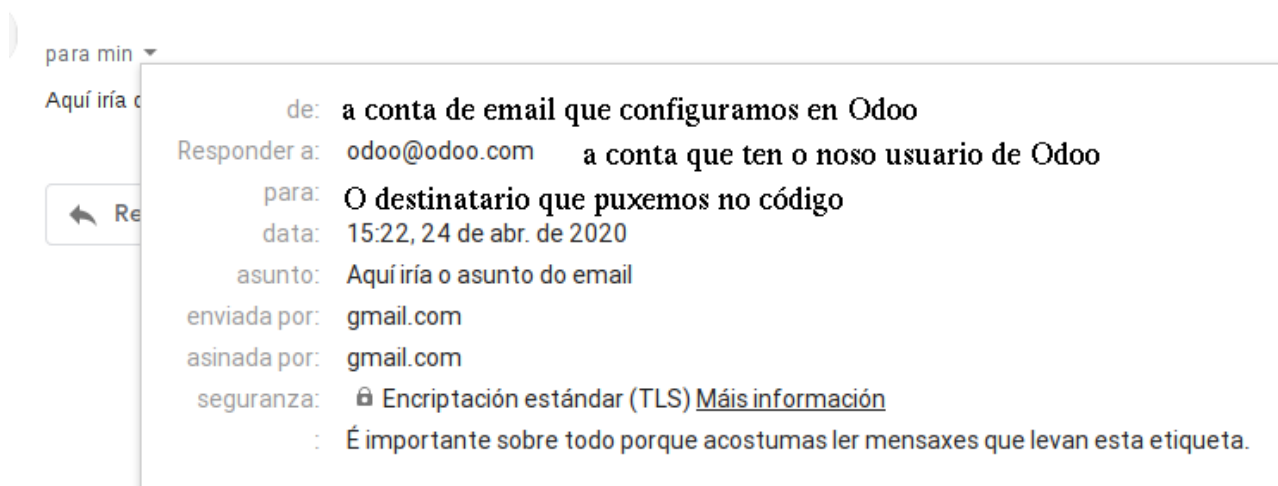
```
mail_id = self.env['mail.mail'].create(mail_valores)
```

E con

```
mail_id.sudo().send()
```

enviamos o email. Usamos `sudo` para ter permisos de acceso ás mensaxes.

Finalmente, vemos no noso correo o que recibiríamos:



En resumo, traballamos con 3 contas de email:

1. A de gmail que configuramos en Odoo para realizar o envío. (Recomendo unha que non sexa a vosa habitual, polo tema da seguridade)
2. A que puxemos no usuario administrador (podedes poñer a vosa), será a conta que reciba as respostas dos emails enviados e a que utilizamos para logearnos en Odoo.
3. A que poñemos no código ou nos datos dos clientes,proveedores... Que será a conta que reciba os emails enviados dende Odoo.