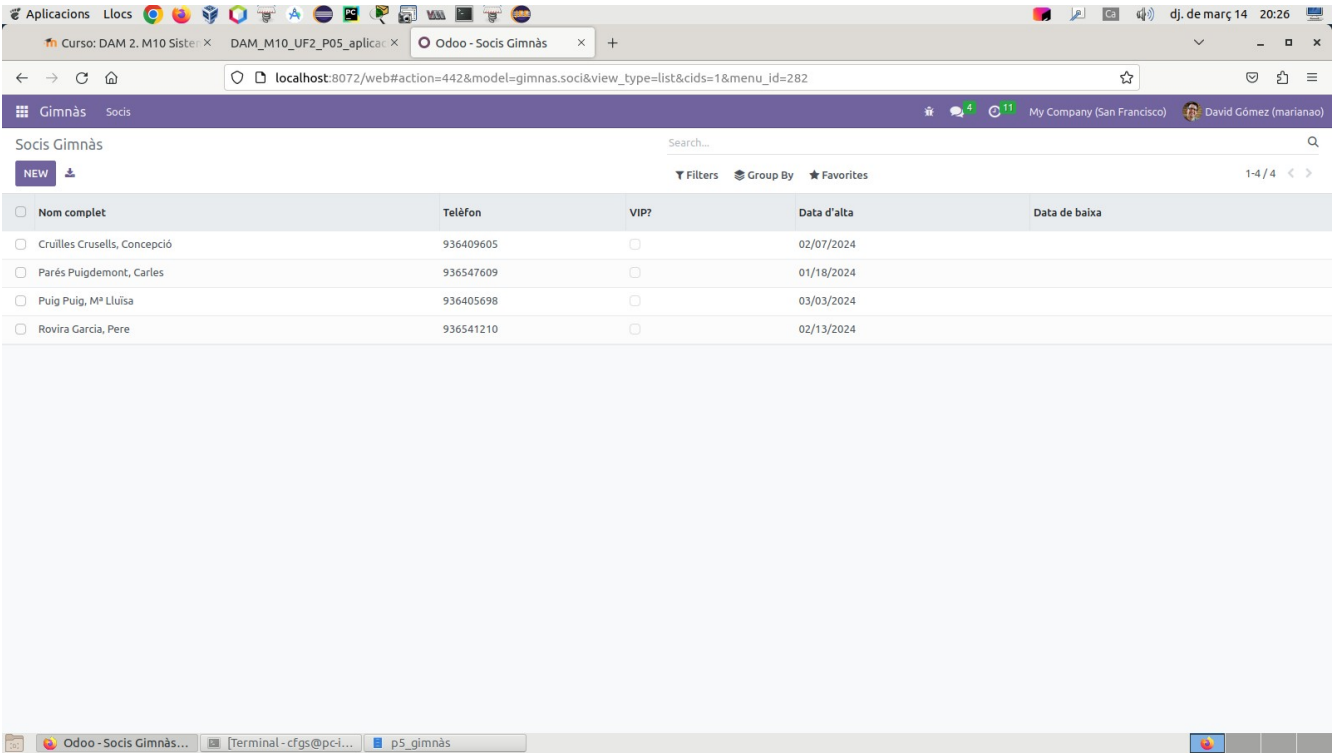
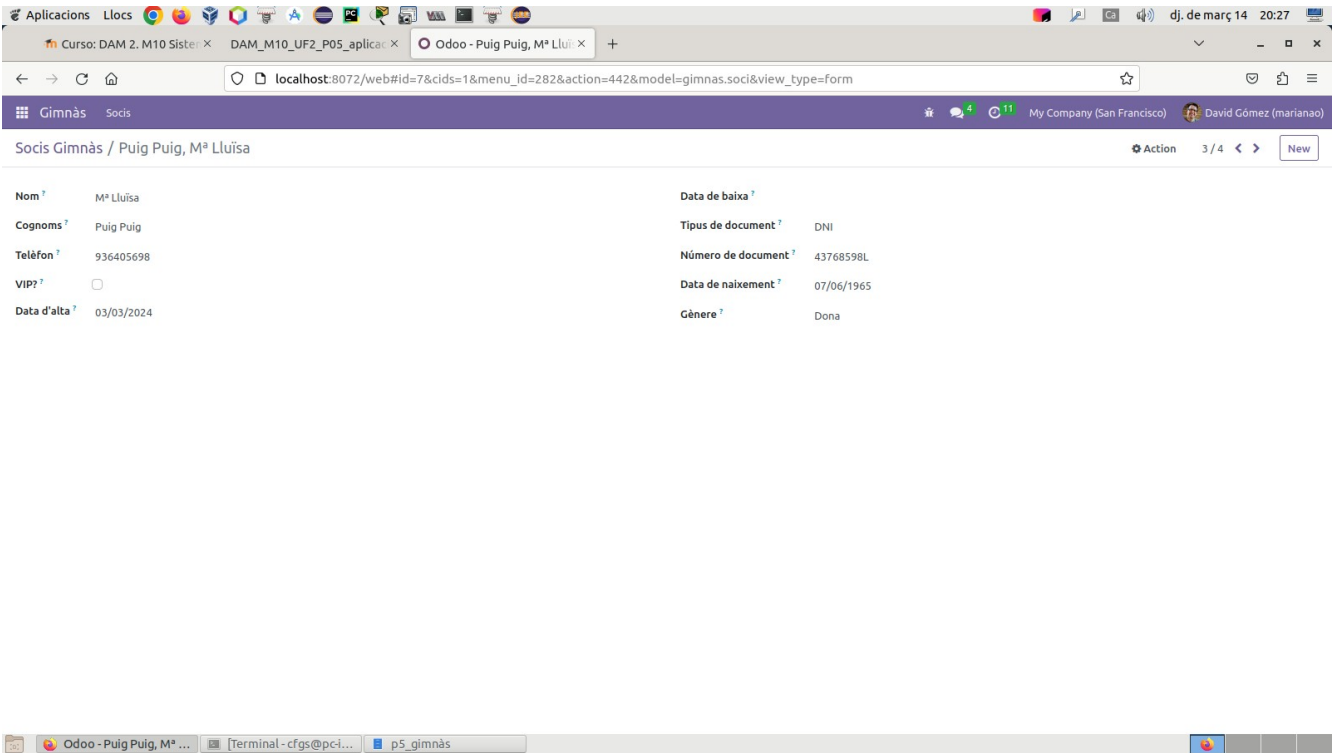


Aplicació d'un Gimnàs

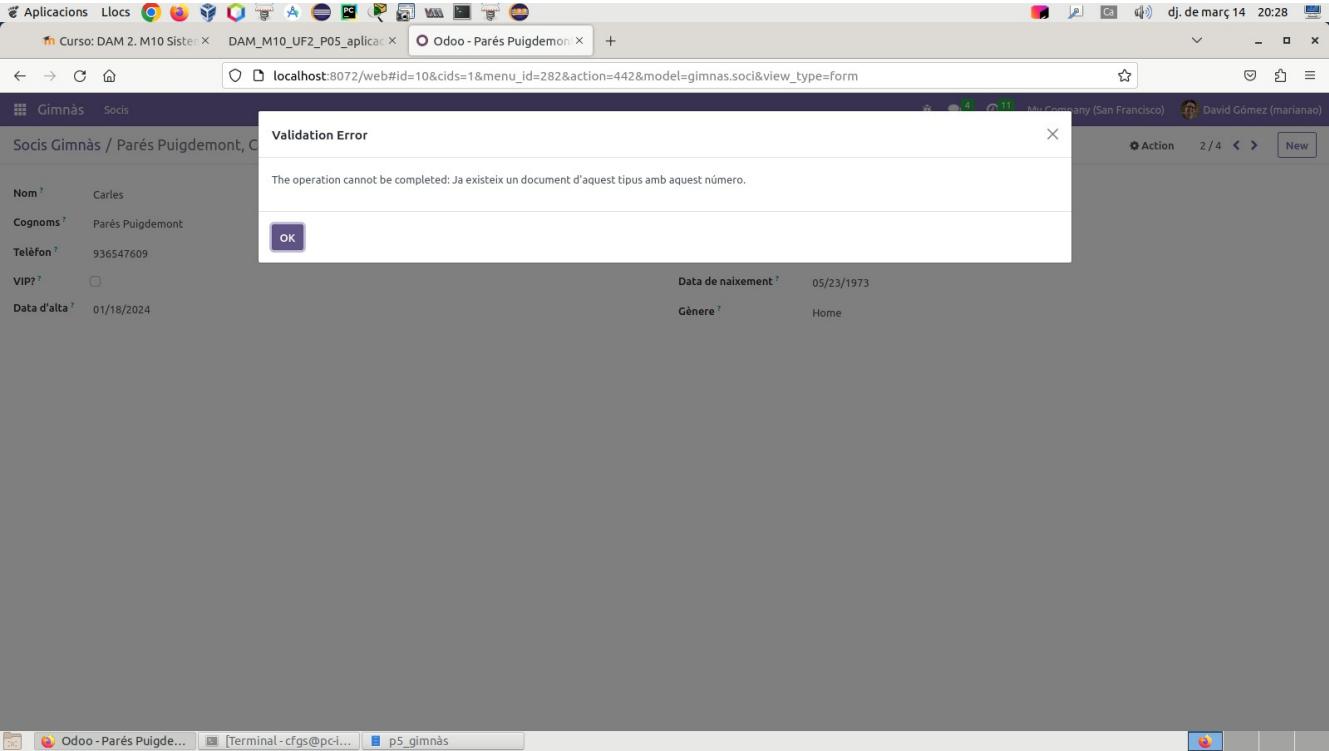
E1



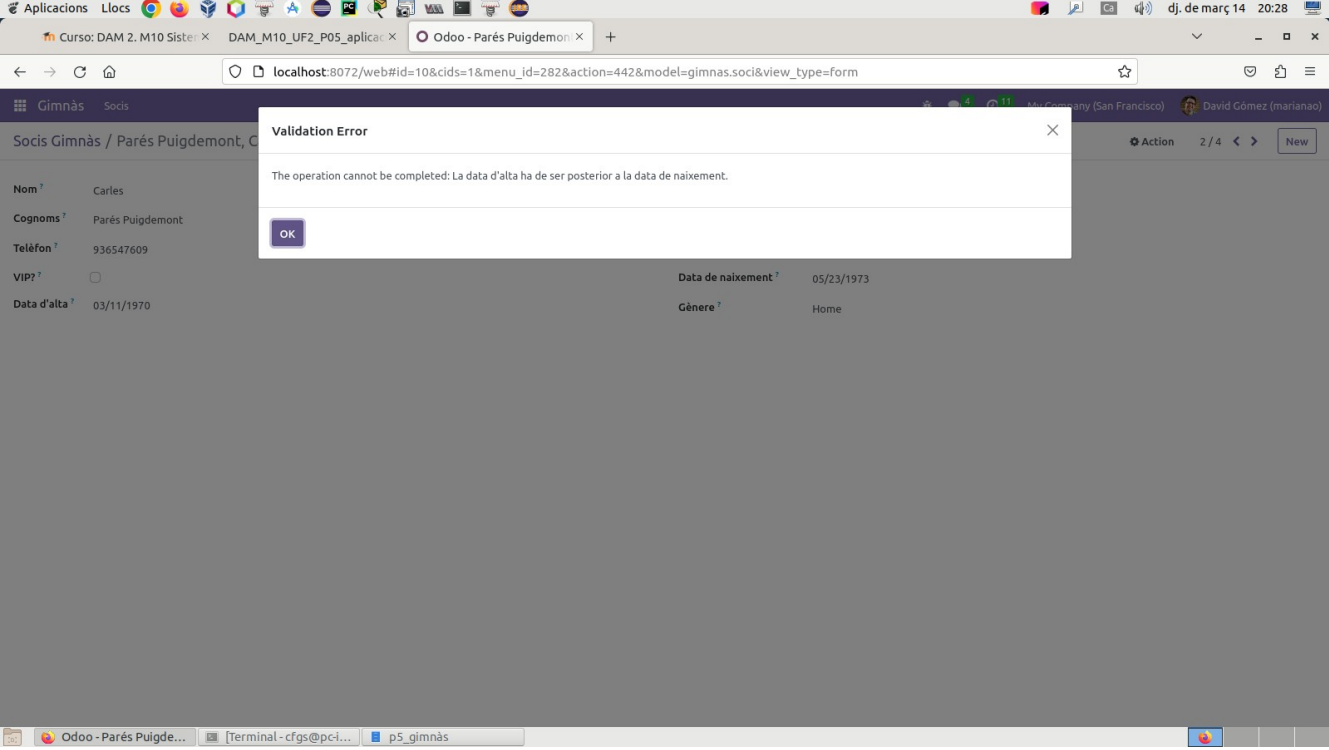
E2



E3



E4



E5: La funció de Python que valida el número de document no funciona. Sembla que recull el valor del camp del número de document com un booleà i no com un string. Tots els intents de fer que funcioni han estat en va. No obstant, el següent punt n'inclou el codi, tot i que està comentat per a evitar errors.

E6

```
from odoo import api, fields, models
from odoo.exceptions import ValidationError

class Soci(models.Model):

    _name = 'gimnas.soci'
    _description = 'Soci'
    _order = 'name_last, name_first'

    name = fields.Char('Nom complet', compute='_nom_complet')
    active = fields.Boolean('Actiu?', default=True)

    doc_type = fields.Selection([(('dni', 'DNI'), ('nie', 'NIE'))], string='Tipus de document', required=True)
    doc_num = fields.Char('Número de document', required=True)
    name_first = fields.Char('Nom', required=True)
    name_last = fields.Char('Cognoms', required=True)
    date_birthday = fields.Date('Data de naixement')
    gender = fields.Selection([(('male', 'Home'), ('female', 'Dona'), ('other', 'Altre'))], string='Gènere')
    phone = fields.Char('Telèfon', required=True)
    is_vip = fields.Boolean('VIP?')
    date_register = fields.Date('Data d\'alta', required=True)
    date_unsubscribe = fields.Date('Data de baixa')

    _sql_constraints = [
        ('doc_type_num_unique', 'UNIQUE(doc_type, doc_num)', 'Ja existeix un document d\'aquest tipus amb aquest número.'),
        ('date_birthday_valid', 'CHECK(date_birthday >= \'1900-01-01\')', 'La data de naixement no pot ser anterior a 01/01/1900.'),
        ('date_register_&gt;_birthday', 'CHECK(date_register > date_birthday)', 'La data d\'alta ha de ser posterior a la data de naixement.'),
        ('date_unsubscribe_&gt;_register', 'CHECK(date_unsubscribe >= date_register)', 'La data de baixa no pot ser anterior a la data d\'alta.')
    ]

    @api.depends('name_first', 'name_last')
    def _nom_complet(self):
        for record in self:
            record.name = record.name_last + ', ' + record.name_first

# @api.constrains('doc_num','doc_type')
# def _validacio_num_document(self):
#     letters = 'TRWAGMYFPDXBNJZSQVHLCKE'
#     letters_nie = 'XYZ'
#     for record in self:
#         doc_num = str(record.doc_num)
#         if record.doc_type == 'NIE':
#             check_first = str(doc_num[0])
#             if check_first in letters_nie:
#                 doc_num = doc_num.replace(check_first, str(letters_nie.find(check_first)))
#             else:
#                 raise ValidationError('NIE amb primera lletra incorrecta.')
#         num = int(doc_num[:-1])
#         if not num.isdigit():
#             raise ValidationError('El número de document només pot contenir dígits.')
#         char = doc_num[-1:]
#         mod = int(num) % 23
#         if letters[mod] != char:
#             raise ValidationError('La lletra per a aquest número de document és incorrecta.')
```