



PROYECTO SGE

CFGS Desarrollo de Aplicaciones
Multiplataforma
Informática y Comunicaciones

Desarrollo del módulo “manage” con Odoo ERP

Año: 2024

Fecha de presentación: (fecha de presentación)

Nombre y Apellidos: Alejandro de Gregorio Miguel
Email: Alejandro.gremig@educa.jcyl.es

Índice

No se encontraron entradas de tabla de contenido.

1 Introducción

Un sistema ERP es una herramienta de software que ayuda a organizar y manejar las actividades clave de una empresa. Su principal meta es juntar información y tareas en un solo lugar para hacer todo más eficiente, ahorrar dinero y ayudar a tomar decisiones rápidas basadas en datos.

Características principales:

Integración: conectando diferentes áreas de la empresa.

Automatización: menos trabajo manual y menos errores.

Escalabilidad: ajustarse al crecimiento de tu negocio.

Análisis: Proporciona herramientas avanzadas para la toma de decisiones.

La metodología ágil es una forma de trabajar en proyectos que pone énfasis en ser flexible, colaborativo y entregar constantemente resultados valiosos. Se enfocan en ajustarse rápidamente a los cambios e incluir a los clientes en todas las partes del proceso. SCRUM es uno de los métodos ágiles más populares y organizados.

SCRUM es una estructura adaptable hecha para proyectos complicados y que cambian mucho. Separa el proceso de creación en partes más pequeñas que se llaman sprints, los cuales suelen durar entre 1 y 4 semanas. Cada sprint termina con la salida de un producto o de una mejora funcional.

Funciones principales:

Dueño del producto: identificar y poner en orden de importancia los requerimientos del producto en la lista de tareas.

Scrum Master: ayuda al equipo a seguir las reglas de desarrollo ágil, resuelve problemas y se asegura de que no haya obstáculos en el camino.

Equipo de desarrollo: hacer y entregar nuevos proyectos.

Eventos principales:

Planificación de Sprint: se establecen las metas y actividades del Sprint.

Daily Scrum: Reuniones diarias para coordinar esfuerzos y resolver problemas.

Sprint Review: muestra el trabajo que se ha hecho hasta ahora.

Revisión del Sprint: piensa en lo que salió bien y en lo que se puede mejorar.

SCRUM promueve que todo sea claro, se revise continuamente y se modifique según sea necesario. Es perfecto para proyectos en los que los objetivos pueden cambiar rápidamente.

2 Estado del arte

2.1 ERP

- a) Definición de los ERP
- b) Evolución de los ERPs
- c) Principales ERP
- d) ERP seleccionado (Odoo)
- e) Instalación y desarrollo (formas de instalación, explicando la que se va a usar para desarrollar el proyecto: Docker)
- f) Especificaciones técnicas
 - a. Arquitectura de Odoo
 - b. Composición de un módulo

2.2 SCRUM

- a) Definición de SCRUM
- b) Evolución
- c) Funcionamiento
- d) Principales conceptos (explicar los principales conceptos: proyecto, historias de usuario, sprint, tarea...)

3 Continuación del proyecto

4 Descripción general del proyecto

4.1 Objetivos

La meta principal de este proyecto es desarrollar un módulo de Odoo que facilite la gestión de los componentes de un proyecto mediante la metodología SCRUM, permitiéndonos agregar, modificar, eliminar y visualizar los distintos componentes de nuestro proyecto. Este módulo contará con campos computados para simplificar el trabajo a aquellos que lo empleen.

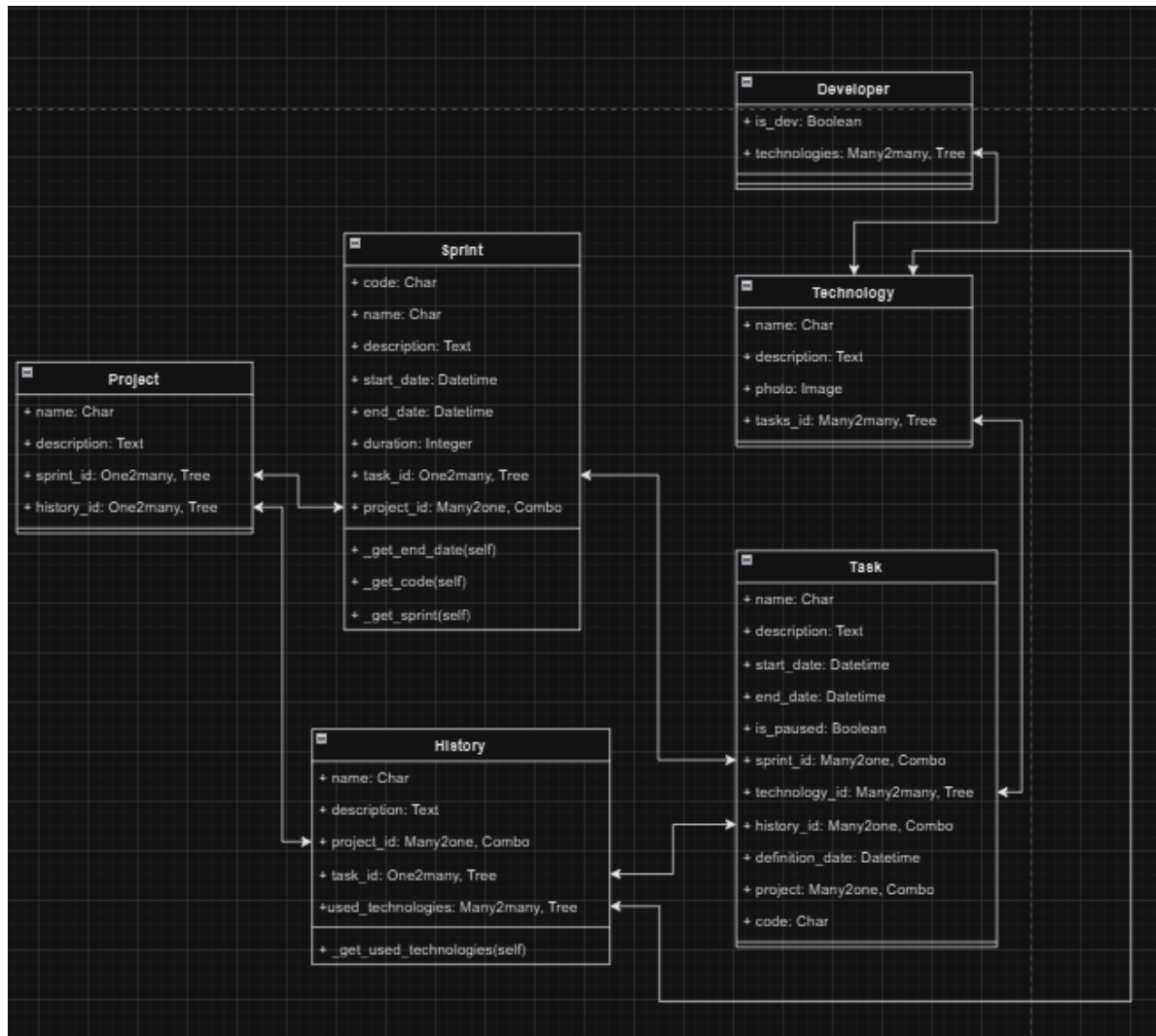
4.2 Entorno de trabajo

Para desarrollar los módulos de Odoo es necesario un IDE cuyo lenguaje de programación sea Python. El IDE que he escogido, en mi caso, es PyCharm, debido a que es un entorno optimizado para Python, con características avanzadas como autocompletado, depuración integrada y herramientas específicas para desarrollo en este lenguaje.

Para desarrollar un módulo de Odoo, debemos tener instalado el propio sistema de Odoo. Para poder facilitar el trabajo tanto en la modificación como para la programación de nuestro módulo hemos utilizado Docker, un sistema operativo que crea unos contenedores que pueden ser ejecutados en local.

5 Diseño de la aplicación

5.1 Modelo relacional de la base de datos



5.2 Partes del proyecto

Este proyecto se compone de varios modelos y vistas:

5.2.1 Clase Project y vista

```
class project(models.Model):
    _name='manage.project'
    _description='manage.project'

    name = fields.Char(string="Nombre", readonly=False, required=True, help="Introduzca el nombre")
    description = fields.Text(string="Descripción")
    history_id = fields.One2many(string="Histories", comodel_name="manage.history", inverse_name="project_id")
    sprint_id = fields.One2many(string="Sprints", comodel_name="manage.sprint", inverse_name="project_id")
```

```

<odoo>
<data>

<record model="ir.ui.view" id="vista_manage_project_tree">
  <field name="name">vista_manage_project_tree</field>
  <field name="model">manage.project</field>
  <field name="arch" type="xml">
    <tree>
      <field name="name"/>
      <field name="description"/>
    </tree>
  </field>
</record>

<record model="ir.ui.view" id="vista_manage_project_form">
  <field name="name">vista_manage_project_form</field>
  <field name="model">manage.project</field>
  <field name="arch" type="xml">
    <form string="manage_project">
      <sheet>
        <group name="group_top">
          <field name="name"/>
          <field name="description"/>
          <field name="history_id"/>
          <field name="sprint_id"/>
        </group>
      </sheet>
    </form>
  </field>
</record>

<record model="ir.actions.act_window" id="accion_manage_project_form">
  <field name="name">manage projects</field>
  <field name="type">ir.actions.act_window</field>
  <field name="res_model">manage.project</field>
  <field name="view_mode">tree,form</field>
  <field name="help" type="html">
    <p class="oe_view_nocontent_create">
      Projects
    </p>
    <p> Click <strong> 'Crear' </strong> para añadir nuevos elementos
    </p>
  </field>
</record>

<menuitem name="Projects" id="menu_manage_projects" parent="menu_manage_management" action="accion_manage_project_form"/>
</data>
</odoo>

```

5.2.2 Clase Sprint y vista

```
class sprint(models.Model):
    _name='manage.sprint'
    _description='manage.sprint'
    code = fields.Char(string="Codigo", compute="_get_code")
    name = fields.Char(string="Nombre", readonly=False, required=True, help="Introduzca el nombre")
    description = fields.Text(string="Descripción")
    start_date = fields.Datetime(string="Fecha de inicio")
    end_date = fields.Datetime(string="Fecha de fin", compute="_get_end_date", store=True)
    duration = fields.Integer(string="Duracion", default="15")
    task_id = fields.One2many(string="Tasks", comodel_name="manage.task", inverse_name="sprint_id")
    project_id = fields.Many2one("manage.project", string="Project", required=True, ondelete="cascade")

    def _get_code(self):
        for sprint in self:
            if len(sprint.task_id) == 0:
                sprint.code = "FILM_"+str(sprint.id)
            else:
                sprint.code = str(sprint.task_id.name).upper()+"_"+str(sprint.id)

    @api.depends('start_date','duration')
    def _get_end_date(self):
        for sprint in self:
            if isinstance(sprint.start_date,datetime.datetime) and sprint.duration>0:
                sprint.end_date=sprint.start_date+datetime.timedelta(days=sprint.duration)
            else:
                sprint.end_date=sprint.start_date

    @api.depends('code')
    def _get_sprint(self):
        #global sprints, task, found
        for task in self:
            sprints = self.env['manage.sprint'].search([('project.id', '=', task.history.project.id)])
            found = False
            for sprint in sprints:
                if isinstance(sprint.end_date, datetime.datetime) and sprint.end_date > datetime.datetime.now():
                    task.sprint = sprint.id
                    found = True
            if not found:
                task.sprint = False
```



```
<odoo>
<data>

<record model="ir.ui.view" id="vista_manage_sprint_tree">
  <field name="name">vista_manage_sprint_tree</field>
  <field name="model">manage.sprint</field>
  <field name="arch" type="xml">
    <tree>
      <field name="name"/>
      <field name="description"/>
      <field name="start_date"/>
      <field name="end_date"/>
    </tree>
  </field>
</record>

<record model="ir.ui.view" id="vista_manage_sprint_form">
  <field name="name">vista_manage_sprint_form</field>
  <field name="model">manage.sprint</field>
  <field name="arch" type="xml">
    <form string="manage_sprint">
      <sheet>
        <group name="group_top">
          <field name="name"/>
          <field name="description"/>
          <field name="duration"/>
          <field name="start_date"/>
          <field name="end_date"/>
          <field name="task_id"/>
          <field name="project_id"/>
        </group>
      </sheet>
    </form>
  </field>
</record>

<record model="ir.actions.act_window" id="accion_manage_sprint_form">
  <field name="name">manage sprints</field>
  <field name="type">ir.actions.act_window</field>
  <field name="res_model">manage.sprint</field>
  <field name="view_mode">tree,form</field>
  <field name="help" type="html">
    <p class="oe_view_nocontent_create">
      Sprints
    </p>
    <p> Click <strong> 'Crear' </strong> para añadir nuevos elementos
    </p>
  </field>
</record>
<menuitem name="Sprints" id="menu_manage_sprints" parent="menu_manage_management" action="accion_manage_sprint_form"/>
</data>
</odoo>
```

5.2.3 Clase Task y vista

```
class task(models.Model):
    _name='manage.task'
    _description='manage.task'

    name = fields.Char(string="Nombre", readonly=False, required=True, help="Introduzca el nombre")
    description = fields.Text(string="Descripción")
    start_date = fields.Datetime(string="Fecha de inicio")
    end_date = fields.Datetime(string="Fecha de fin")
    is_paused = fields.Boolean(string="¿Pausado?")
    sprint_id = fields.Many2one("manage.sprint", string="Sprint", ondelete="cascade", compute="_get_sprint", store=True)
    history_id = fields.Many2one("manage.history", string="History", required=True, ondelete="cascade")
    technology_id = fields.Many2many(comodel_name="manage.technology",
                                    relation="technogies_tasks",
                                    column1="technologies_ids",
                                    column2="tasks_ids")

    definition_date = fields.Datetime(default=lambda p:datetime.datetime.now())
    project = fields.Many2one("manage.project", related="history_id.project_id", readonly=True)
    code = fields.Char(compute="_get_code")
```

```
<odoo>
<data>

<record model="ir.ui.view" id="vista_manage_task_tree">
  <field name="name">vista_manage_task_tree</field>
  <field name="model">manage.task</field>
  <field name="arch" type="xml">
    <tree>
      <field name="name"/>
      <field name="description"/>
      <field name="start_date"/>
      <field name="end_date"/>
      <field name="is_paused"/>
    </tree>
  </field>
</record>

<record model="ir.ui.view" id="vista_manage_task_form">
  <field name="name">vista_manage_task_form</field>
  <field name="model">manage.task</field>
  <field name="arch" type="xml">
    <form string="manage_task">
      <sheet>
        <group name="group_top">
          <field name="name"/>
          <field name="description"/>
          <field name="start_date"/>
          <field name="end_date"/>
          <field name="is_paused"/>
          <field name="sprint_id"/>
          <field name="project"/>
          <field name="history_id"/>
          <field name="technology_id"/>
          <field name="definition_date"/>
        </group>
      </sheet>
    </form>
  </field>
</record>

<record model="ir.actions.act_window" id="accion_manage_task_form">
  <field name="name">manage tasks</field>
  <field name="type">ir.actions.act_window</field>
  <field name="res_model">manage.task</field>
  <field name="view_mode">tree,form</field>
  <field name="help" type="html">
    <p class="oe_view_nocontent_create">
      Tasks
    </p>
    <p> Click <strong> 'Crear' </strong> para añadir nuevos elementos
    </p>
  </field>
</record>

<menuitem name="Tasks" id="menu_manage_tasks" parent="menu_manage_management" action="accion_manage_task_form"/>
</data>
</odoo>
```

5.2.4 Clase History y vista

```
class history(models.Model):
    _name='manage.history'
    _description='manage.history'

    name = fields.Char(string="Nombre", readonly=False, required=True, help="Introduzca el nombre")
    description = fields.Text(string="Descripción")
    task_id = fields.One2many(string="Tasks", comodel_name="manage.task", inverse_name="history_id")
    project_id = fields.Many2one("manage.project", string="Project", required=True, ondelete="cascade")
    used_technologies = fields.Many2many("manage.technology", compute="_get_used_technologies")

    def _get_used_technologies(self):
        for history in self:
            technologies = None # Array para concatenar todas las tecnologías. Inicialmente no tiene valor
            for task in history.task_id: # Para cada una de las tareas de la historia
                if not technologies:
                    technologies = task.technology_id
                else:
                    technologies = technologies + task.technology_id
            history.used_technologies = technologies # Asignar las tecnologías a la historia
```

```
<odoo>
<data>

<record model="ir.ui.view" id="vista_manage_history_tree">
  <field name="name">vista_manage_history_tree</field>
  <field name="model">manage.history</field>
  <field name="arch" type="xml">
    <tree>
      <field name="name"/>
      <field name="description"/>
    </tree>
  </field>
</record>

<record model="ir.ui.view" id="vista_manage_history_form">
  <field name="name">vista_manage_history_form</field>
  <field name="model">manage.history</field>
  <field name="arch" type="xml">
    <form string="manage_history">
      <sheet>
        <group name="group_top">
          <field name="name"/>
          <field name="description"/>
          <field name="task_id"/>
          <field name="project_id"/>
          <field name="used_technologies"/>
        </group>
      </sheet>
    </form>
  </field>
</record>

<record model="ir.actions.act_window" id="accion_manage_history_form">
  <field name="name">manage histories</field>
  <field name="type">ir.actions.act_window</field>
  <field name="res_model">manage.history</field>
  <field name="view_mode">tree,form</field>
  <field name="help" type="html">
    <p class="oe_view_nocontent_create">
      Histories
    </p>
    <p> Click <strong> 'Crear' </strong> para añadir nuevos elementos
    </p>
  </field>
</record>

<menuitem name="Histories" id="menu_manage_histories" parent="menu_manage_management" action="accion_manage_history_form"/>
</data>
</odoo>
```

5.2.5 Clase Technology y vista

```
class technology(models.Model):
    _name='manage.technology'
    _description='manage.technology'

    name = fields.Char(string="Nombre", readonly=False, required=True, help="Introduzca el nombre")
    description = fields.Text(string="Descripción")
    photo = fields.Image(string="Imagen")
    tasks_id = fields.Many2many(comodel_name="manage.task",
                                relation="technologies_tasks",
                                column1="tasks_ids",
                                column2="technologies_ids")
```

```
<odoo>
<data>

<record model="ir.ui.view" id="vista_manage_technology_tree">
  <field name="name">vista_manage_technology_tree</field>
  <field name="model">manage.technology</field>
  <field name="arch" type="xml">
    <tree>
      <field name="name"/>
      <field name="description"/>
      <field name="photo"/>
    </tree>
  </field>
</record>

<record model="ir.ui.view" id="vista_manage_technology_form">
  <field name="name">vista_manage_technology_form</field>
  <field name="model">manage.technology</field>
  <field name="arch" type="xml">
    <form string="manage_technology">
      <sheet>
        <group name="group_top">
          <field name="name"/>
          <field name="description"/>
          <field name="photo"/>
          <field name="tasks_id"/>
        </group>
      </sheet>
    </form>
  </field>
</record>

<record model="ir.actions.act_window" id="accion_manage_technology_form">
  <field name="name">manage technologies</field>
  <field name="type">ir.actions.act_window</field>
  <field name="res_model">manage.technology</field>
  <field name="view_mode">tree,form</field>
  <field name="help" type="html">
    <p class="oe_view_nocontent_create">
      Technologies
    </p>
    <p> Click <strong> 'Crear' </strong> para añadir nuevos elementos
    </p>
  </field>
</record>

<menuitem name="Technologies" id="menu_manage_technologies" parent="menu_manage_management" action="accion_manage_technology_form"/>
</data>
</odoo>
```

5.2.6 Clase Developer y vista

```
class developer(models.Model):
    _name = "res.partner"
    _inherit = "res.partner"

    is_dev = fields.Boolean()
    technologies = fields.Many2many("manage.technology",
        relation="developer_technologies",
        column1="developer_id",
        column2="technologies_id")
```

```
<?xml version="1.0" encoding="utf-8"?>
<odoo>
  <data>
    <record model="ir.ui.view" id="manage.devs_partner_form">
      <field name="name">manage devs form</field>
      <field name="model">res.partner</field>
      <field name="inherit_id" ref="base.view_partner_form"/>
      <field name="mode">primary</field>
      <field name="arch" type="xml">
        <xpath expr="//sheet/notebook/page[@name='internal_notes']" position="after">
          <page name="devs" string="Devs">
            <group>
              <group>
                <field name="technologies"/>
              </group>
            </group>
          </page>
        </xpath>
      </field>
    </record>

    <record model="ir.actions.act_window" id="manage.action_developer_window">
      <field name="name">manage developer window</field>
      <field name="res_model">res.partner</field>
      <field name="view_mode">tree,form</field>
    </record>

    <record model="ir.actions.act_window.view" id="manage.action_view_developer_tree">
      <field name="sequence" eval="1"/>
      <field name="view_mode">tree</field>
      <field name="view_id" ref="base.view_partner_tree"/>
      <field name="act_window_id" ref="manage.action_developer_window"/>
    </record>

    <record model="ir.actions.act_window.view" id="manage.action_view_developer_form">
      <field name="sequence" eval="2"/>
      <field name="view_mode">form</field>
      <field name="view_id" ref="manage.devs_partner_form"/>
      <field name="act_window_id" ref="manage.action_developer_window"/>
    </record>

    <menuitem name="Devs" id="menu_manage_devs" parent="manage.menu_manage_management"
      action="manage.action_developer_window"/>
  </data>
</odoo>
```

5.2.7 Menú padre

```
<odoo>
  <data>
    <menuitem name="Manage" id="menu_manage_raiz"/>
    <menuitem name="Management" id="menu_manage_management" parent="menu_manage_raiz"/>
  </data>
</odoo>
```

5.2.8 Archivo de configuración y de manifiesto

```
id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
access_manage_manage,manage.manage,model_manage_manage,base.group_user,1,1,1,1
access_manage_task,manage.task,model_manage_task,base.group_user,1,1,1,1
access_manage_sprint,manage.sprint,model_manage_sprint,base.group_user,1,1,1,1
access_manage_project,manage.project,model_manage_project,base.group_user,1,1,1,1
access_manage_history,manage.history,model_manage_history,base.group_user,1,1,1,1
access_manage_technology,manage.technology,model_manage_technology,base.group_user,1,1,1,1
```

```
# -*- coding: utf-8 -*-
{
    'name': "manage",

    'summary': """
        Short (1 phrase/line) summary of the module's purpose, used as
        subtitle on modules listing or apps.openerp.com""",

    'description': """
        Long description of module's purpose
        """,

    'author': "My Company",
    'website': "https://www.yourcompany.com",

    # Categories can be used to filter modules in modules listing
    # Check https://github.com/odoo/odoo/blob/16.0/odoo/addons/base/data/ir_module_category_data.xml
    # for the full list
    'category': 'Uncategorized',
    'version': '0.1',

    # any module necessary for this one to work correctly
    'depends': ['base'],

    # always loaded
    'data': [
        'security/ir.model.access.csv',
        'views/views.xml',
        'views/models.xml',
        'views/task.xml',
        'views/sprint.xml',
        'views/project.xml',
        'views/history.xml',
        'views/technology.xml',
        'views/developer.xml',
        'views/templates.xml',
    ],
    # only loaded in demonstration mode
    'demo': [
        'demo/demo.xml',
    ],
}
```


5.3 Ampliación del proyecto

6 Pruebas de funcionamiento

7 Conclusiones y posibles ampliaciones

8 Bibliografía

Índice

Modificar la **portada** con vuestros datos personales.

TÍTULO DEL PROYECTO: “Desarrollo del módulo “manage” con Odoo ERP; para gestionar proyectos usando metodologías ágiles: scrum”.

El primer apartado que vamos a incluir es la **INTRODUCCIÓN**: hablar brevemente sobre los sistemas ERP y las metodologías ágiles (en concreto sobre SCRUM).

Segundo apartado: ORGANIZACIÓN DE LA MEMORIA (lo vamos a dejar para el final; consistirá en enumerar los apartados de la memoria, con una pequeña descripción de cada uno de ellos).

Tercer apartado: ESTADO DEL ARTE. Con los siguientes subapartados:

1. ERP

- a. Definición de los ERP
- b. Evolución de los ERPs
- c. Principales ERP
- d. ERP seleccionado (Odoo)
- e. Instalación y desarrollo (*formas de instalación, explicando la que se va a usar para desarrollar el proyecto: Docker*)
- f. Especificaciones técnicas
 - i. Arquitectura de Odoo
 - ii. Composición de un módulo

2. SCRUM

- a. Definición de SCRUM
- b. Evolución
- c. Funcionamiento
- d. Principales conceptos (*explicar los principales conceptos: proyecto, historias de usuario, sprint, tarea...*)

Enlace sugerido: <https://www.atlassian.com/es/agile/scrum>

CONTINUACIÓN PROYECTO MANAGE

Investigar una posible ampliación de este proyecto e implementarla en vuestra aplicación. La ampliación tiene que incluir alguno(s) de los siguientes apartados:

- Algún aspecto no visto hasta ahora en el módulo de Sistemas de Gestión Empresarial
- Algún aspecto relacionado con el uso de CRUD, usando el ORM de Odoo

Añadir los siguientes puntos, a continuación de los que ya teníamos anteriormente en la memoria:

Descripción general del proyecto:

- **Objetivos** (breve descripción de lo que se ha pretendido alcanzar con el proyecto);
- **Entorno de trabajo** (explicar todas las herramientas utilizadas para desarrollar el proyecto: Docker, navegador, visual studio code...)

Diseño de la aplicación:

- **Modelo relacional de la BBDD**
- **Partes del proyecto** (models, views, security...), explicando brevemente lo que consideréis importante
- **Ampliación del proyecto**, explicando detalladamente el objetivo de la ampliación, el desarrollo...

Pruebas de funcionamiento

Conclusiones y posibles ampliaciones

Bibliografía

Normas de entrega:

Extensión mínima: 30 páginas

Formato entrega:

- Repositorio de Github, con los archivos generados en el desarrollo.
- Archivo README con este contenido: título del proyecto, descripción del proyecto, enlace a la memoria del proyecto en formato pdf
- La memoria en archivo independiente,
 - formato pdf;
 - nombre: Apellido1_Apellido2_Nombre_proyectomanage.pdf

Presentación oral en clase de algunos apartados del proyecto:

- Descripción general del proyecto
- Diseño de la aplicación (incluir vuestra ampliación)
- Conclusiones y posibles ampliaciones

Criterios calificación memoria proyecto manage:

Presentación formal (20%)

- Se ajusta a los requerimientos formales establecidos: portada, bibliografía, formato entrega, nombre archivo
- Se incluye un índice estructurado y coherente con el contenido del proyecto
- El texto está bien redactado, no presenta incoherencias gramaticales ni faltas de ortografía
- Presenta el proyecto en forma y plazo establecidos (1 punto menos por cada día de retraso)

Contenidos (40%)

- Originalidad del tema elegido (contenidos originales, no repetidos)
- Grado de dificultad en orden a la investigación de contenidos
- Grado de profundización en la investigación de contenidos
- Explicación clara y concisa, con capturas, explicaciones breves...
- Incluye todos los apartados requeridos en el proyecto

Defensa oral (40%)

- Se ajusta al tiempo marcado
- Objetivo del proyecto
- Puntos esenciales de la resolución
- Conclusiones finales
- Grado de conocimiento y dominio de los contenidos expuestos
- Lenguaje técnico utilizado