Contents

[I. Odoo là gì? 1](#_Toc184134601)

[II. Kiến trúc module: 2](#_Toc184134602)

[1. Các thành phần chính trong module: 2](#_Toc184134603)

[a. \_\_manifest\_\_.py 2](#_Toc184134604)

[b. Models 2](#_Toc184134605)

[c. Views 2](#_Toc184134606)

[d. Sercurity 2](#_Toc184134607)

[e. Data 2](#_Toc184134608)

[f. Controllers 3](#_Toc184134609)

[g. Static 3](#_Toc184134610)

[2. Cách tạo một module: 4](#_Toc184134611)

[a. Import các thư viện cần thiết: 4](#_Toc184134612)

[b. Định nghĩa mô hình 4](#_Toc184134613)

[c. Định nghĩa các trường: 4](#_Toc184134614)

[d. Định nghĩa các phương thức 4](#_Toc184134615)

[e. Gắn các decorator cho các phương thức 4](#_Toc184134616)

[f. Định nghĩa các view để hiển thị dữ liệu từ model 4](#_Toc184134617)

[g. Định nghĩa action 5](#_Toc184134618)

[h. Thiết Lập Security Cho Model Trong Odoo 6](#_Toc184134619)

[i. Định Nghĩa Nhóm Người Dùng Tùy Chỉnh (Nếu Cần) 6](#_Toc184134620)

[j. Thêm tất cả vào manifest 6](#_Toc184134621)

[3. Tạo module nâng cao 6](#_Toc184134622)

[a. Các thuộc tính nâng cao trong views 6](#_Toc184134623)

[b. Button 7](#_Toc184134624)

[c. Wizard 7](#_Toc184134625)

[d. Report 8](#_Toc184134626)

[e. Controller 8](#_Toc184134627)

[f. Static files 9](#_Toc184134628)

[g. Templates 9](#_Toc184134629)

[4. Kế thừa trong Odoo 10](#_Toc184134630)

[Kế thừa lớp (Model Inheritance) 10](#_Toc184134631)

[Kế thừa view và template (View & Template Inheritance) 10](#_Toc184134632)

[Kế Thừa Template Qweb 10](#_Toc184134633)

[Kế thừa JavaScript 12](#_Toc184134634)

[5. Snippet Trong Odoo 12](#_Toc184134635)

[Snippet Là Gì? 12](#_Toc184134636)

[Các Snippet Có Sẵn Trong Odoo 12](#_Toc184134637)

[Tạo Snippet Tùy Chỉnh 12](#_Toc184134638)

# Odoo là gì?

Odoo là một nền tảng phần mềm quản lý doanh nghiệp mã nguồn mở được thiết kế để đáp ứng nhu cầu quản lý đa dạng của các doanh nghiệp, từ nhỏ đến lớn. Nó cung cấp một bộ ứng dụng tích hợp, cho phép các doanh nghiệp quản lý các hoạt động như tài chính, bán hàng, sản xuất, nhân sự, kho vận và tiếp thị.

# Kiến trúc module:

Module là một thành phần độc lập trong Odoo, cung cấp một tính năng hoặc nhóm tính năng cụ thể, như quản lý bán hàng, kế toán, kho, CRM, hoặc nhân sự. Trong Odoo, tất cả mọi thứ đều bắt đầu từ module

## Các thành phần chính trong module:

### \_\_manifest\_\_.py

Đây là file mô tả của module, bao gồm Tên, phiên bản, tác giả, các phụ thuộc và các file cần nạp.

Các thành phần nên xuất hiện trong \_\_manifest\_\_.py, ví dụ:

{

'name': 'Custom Product Module',

'version': '1.0',

'author': 'Your Name',

'category': 'Sales',

'summary': 'Module for managing custom products',

'description': """

This module allows users to manage custom products,

including tracking product details and prices.

""",

'depends': ['base', 'sale'],

'data': [

'views/product\_views.xml',

'security/ir.model.access.csv',

],

'demo': ['demo/product\_demo.xml'],

'installable': True,

'auto\_install': False,

'application': True,

'license': 'AGPL-3',

'website': 'http://www.example.com',

'images': ['static/description/icon.png'],

}

### Models

Đây là nơi chứa các file python định nghĩa các logic nghiệp vụ và tương tác với cơ sở dữ liệu thông qua ORM.

* Mỗi model đại diện cho một bảng trong Database, mỗi đối tượng trong model đại diện cho một bảng ghi trong bảng đó
* Model bao gồm các fields, các phương thức và các quan hệ giữa các model

#### Các Class Model Trong Odoo:

* models.Model: Đây là loại model cơ bản và phổ biến nhất trong Odoo. Nó đại diện cho các bảng dữ liệu thường dùng trong hệ thống.
* models.TransientModel: Loại model này được sử dụng cho các wizard (trình hướng dẫn) hoặc các hành động tạm thời. Dữ liệu trong TransientModel thường không được lưu trữ lâu dài và có thể bị xóa tự động sau một thời gian.
* models.AbstractModel: Đây là loại model không tạo ra một bảng trong cơ sở dữ liệu riêng biệt. Thay vào đó, nó được dùng để định nghĩa các phương thức và thuộc tính chung mà các model khác có thể kế thừa.

### Views

Đây là nơi định nghĩa giao diện người dùng trong Odoo

* Bao gồm các view cơ bản như Form, Tree, Kaban, Calendar, ...
* Menu view để điều hướng đến các view
* Các menu view phải gắn với action view để thực hiện hành động từ giao diện người dùng

### Sercurity

Bao gồm các setup quyền truy cập và các qui tắc bảo mật trong module

#### Các thành phần bảo mật trong Odoo

* Access Control Lists (ACLs)
* Record Rules
* Groups
* Access Control for Models
* Sercurity for Action and Views

### Data

Nạp dữ liệu mẫu/demo hoặc cấu hình dữ liệu ban đầu cho module

### Controllers

Định nghĩa các API và route để xử lý các yêu cầu HTTP

Cấu trúc của Controller trong Odoo

* **Route:** Được sử dụng để ánh xạ URL với một hàm xử lý. Odoo sử dụng decorator @http.route() để xác định các route.
* **Action**: Hàm xử lý các yêu cầu HTTP. Mỗi action được ánh xạ với một route cụ thể
* **Response**: Các hàm này trả về phản hồi HTTP như dữ liệu JSON, HTML, hoặc tệp tải xuống.

### Static

Đây là thư mục chứa các tệp tĩnh như css, javascript, images,...

#### Thư mục static thường bao gồm các thư mục con như sau:

#### **img/**: Chứa các hình ảnh được sử dụng trong module.

#### **css/**: Chứa các tệp CSS để tùy chỉnh giao diện của ứng dụng.

#### **js/**: Chứa các tệp JavaScript để xử lý các hành động tương tác và hiệu ứng động trong giao diện.

#### **description/**: Thư mục này thường chứa ảnh đại diện hoặc mô tả ngắn gọn của module khi hiển thị trên giao diện quản trị hoặc cửa hàng Odoo.

## Cách tạo một module:

### Import các thư viện cần thiết:

#### from odoo import models, fields, api, \_

#### from odoo.exceptions import ValidationError, UserError

#### from datetime import datetime, timedelta

### Định nghĩa mô hình

#### Cần khai bao các thành phần sau: \_name: tên của mô hình, được ánh xạ sang tên bảng trong Database

#### \_decription: Mô tả của model

#### \_sql\_constraints: các ràng buộc SQL để đảm bảo hợp lệ dữ liệu

#### \_inherit: các model khác mà model này kế thừa

### Định nghĩa các trường:

#### Odoo cung cấp các field khác nhau, phụ thuộc vào thuộc tính của dữ liệu mà lựa chọn kiểu field hợp lý

#### Lưu ý về một số từ khóa quan trong khi tạo filed:

#### String: Tên hiển thị của Field khi hiển thị trên giao diện người dùng

#### Required: xác định xem trường này có bắc buộc phải nhập hay không

#### Default: Đặt giá trị mặc định cho record khi tạo mới

#### Tracking: Bật theo dõi giá trị thay đổi của trường

#### Compute: Đặt tên phương thức để tính toán giá trị cho trường

#### Store: xác định xem giá trị tính toán có được lưu vào db hay không

#### Readonly: xác định xem trường này có được phép chỉnh sửa hay không

#### copy: Xác định xem giá trị của trường có được sao chép khi nhân bản bản ghi hay không.

#### currency\_field: Trường liên kết đến loại tiền tệ cho các trường Monetary.

#### domain: Điều kiện lọc dữ liệu liên quan đến các quan hệ trường.

#### onchange: Đặt phương thức để thực thi khi giá trị của trường thay đổi.

#### constraints: Đặt các ràng buộc để đảm bảo dữ liệu hợp lệ.

### Định nghĩa các phương thức

#### Tùy vào nhu cầu thực tế, thêm các phương thức phù hợp.

#### Một số lưu ý:

#### Tên phương thức nên rõ ràng và theo quy ước đặt tên của Odoo.

#### Sử dụng hàm \_() để đánh dấu các chuỗi cần dịch.

#### **for record in self**: đảm bảo phương thức hoạt động trên từng bản ghi

#### **Exception Handling:** sử dụng các exception trong odoo để xứ lý các trường hợp ngoại lệ

### Gắn các decorator cho các phương thức

#### @api.model: phương thức ở cấp model, thực hiện các thao tác chung trên bảng (thêm/xóa)

#### @api.depends: Xác định các trường mà một trường tính toán (computed field) phụ thuộc vào.

#### @api.onchange: Thực thi một phương thức khi giá trị của một hoặc nhiều trường thay đổi trong giao diện người dùng.

#### @ api.constrains : Đánh dấu phương thức để kiểm tra ràng buộc dữ liệu trước khi lưu vào cơ sở dữ liệu.

### Định nghĩa các view để hiển thị dữ liệu từ model

#### Form view

##### A computer code with text Description automatically generated

##### record: Định nghĩa một view mới.

##### id: Mã định danh duy nhất cho view.

##### model: Model mà view này liên kết.

##### name: Tên của view.

##### arch: Nội dung của view được định nghĩa bằng XML.

##### <tree>: Xác định rằng đây là Tree (List) View.

##### <field>: Các trường sẽ hiển thị trong bảng.

#### Tree view

##### A computer screen shot of a code Description automatically generated

##### <form>: Xác định rằng đây là Form View.

##### <sheet>: Khu vực chứa nội dung của form.

##### <group>: Nhóm các trường để tổ chức bố cục.

#### Search view

##### A computer code with text Description automatically generated

##### <search>: Xác định rằng đây là Search View.

##### <filter>: Thêm các bộ lọc để lọc dữ liệu theo điều kiện.

#### Menu view

##### A computer code with text Description automatically generated

##### id: Định danh duy nhất cho menu.

##### model: Luôn là ir.ui.menu cho các menu.

##### name: Tên hiển thị của menu.

##### parent\_id: ID của menu cha (nếu có).

##### action: Tham chiếu đến action mà menu sẽ kích hoạt khi được chọn.

##### sequence: Thứ tự hiển thị của menu.

### Định nghĩa action

##### A screenshot of a computer code Description automatically generated

##### name: Tên của hành động.

##### res\_model: Model liên kết với view.

##### view\_mode: Các chế độ view (tree, form).

##### view\_id: Tham chiếu đến Tree View đã định nghĩa.

##### search\_view\_id: Tham chiếu đến Search View đã định nghĩa.

### Thiết Lập Security Cho Model Trong Odoo

##### Nội Dung File ir.model.access.csv

##### id,name,model\_id:id,group\_id:id,perm\_read,perm\_write,perm\_create,perm\_unlink

##### access\_estate\_property\_offer,access\_estate\_property\_offer,model\_estate\_property\_offer,base.group\_user,1,1,1,1

##### **id**: Định danh duy nhất cho ACL.

##### **name**: Tên của ACL.

##### **model\_id:id**: Tham chiếu đến model (model\_estate\_property\_offer).

##### **group\_id:id**: Nhóm người dùng có quyền này (base.group\_user cho tất cả người dùng).

##### **perm\_read**, **perm\_write**, **perm\_create**, **perm\_unlink**: Quyền đọc, viết, tạo, xoá (1 = cho phép, 0 = không).

### Định Nghĩa Nhóm Người Dùng Tùy Chỉnh (Nếu Cần)

##### Tạo File security/security.xml

##### A computer code with text Description automatically generated

### Thêm tất cả vào manifest

##### A screen shot of a computer code Description automatically generated

## Tạo module nâng cao

### Các thuộc tính nâng cao trong views

#### Widget

##### Xác định loại widget được sử dụng để hiển thị trường. Có thể là các widget mặc định hoặc tùy chỉnh

#### Options

##### Cung cấp các tùy chọn bổ sung cho widget.

#### Attrs

##### Xác định các điều kiện để thay đổi thuộc tính của trường (như invisible, readonly) dựa trên giá trị của các trường khác.

#### Required

##### Chỉ định trường này là bắt buộc phải nhập.

#### Readonly

##### Làm cho trường chỉ đọc.

#### Invisible

##### Làm cho trường ẩn đi trong giao diện người dùng

#### Help

##### Cung cấp thông tin trợ giúp

#### Domain

##### Áp dụng bộ lọc để giới hạn các lựa chọn trong các trường quan hệ

#### Context

##### Cung cấp ngữ cảnh bổ sung cho trường

#### Class

##### Thêm các lớp CSS để tùy chỉnh giao diện

#### Col

##### Xác định spps cột trong một nhóm

### Button

#### Sử dụng để thực hiện hành động cụ thể trên view

#### Gọi phương thức Python (Object)

##### Tạo phương thức ở lớp Python

###### A screen shot of a computer code Description automatically generated

##### Gọi phương thức trên view

###### 

###### name: Tên phương thức sẽ được gọi khi button được click.

###### type="object": Chỉ định rằng button sẽ gọi một phương thức Python.

#### Gọi Hành Động (Action) (type="action")

##### Button này sẽ gọi một hành động (action) được định nghĩa trong Odoo, chẳng hạn như mở một wizard hoặc một view khác.

##### Định nghĩa một action button

###### 

###### name: Tham chiếu đến action mở tree view (action\_estate\_property\_offer\_tree).

###### type="action": Chỉ định rằng button sẽ gọi một hành động

### Wizard

#### Tạo model cho wizard

##### Wizard được định nghĩa dưới dạng TransientModel, có nghĩa là dữ liệu của chúng không được lưu trữ lâu dài và có thể bị xóa tự động sau một khoảng thời gian. A screen shot of a computer code Description automatically generated

#### Định nghĩa view cho wizard

##### A screenshot of a computer code Description automatically generated

#### Thêm button để gọi wizard

##### A computer code with many colorful text Description automatically generated

### Report

##### Tạo view trong cơ sở dữ liệu

###### A screenshot of a computer program Description automatically generated

##### Tạo wizard để thu thập thông tin

##### Tạo controller để xứ lý yêu cầu tạo báo cáo

##### Tạo view cho wizard

### Controller

#### Tạo thư mục controller

#### Định nghĩa controller

##### Ví dụ:

##### A screen shot of a computer code Description automatically generated

#### Cập nhật \_\_init\_\_.py

#### Cập nhật \_\_manifest\_\_.py

#### Khởi động lại Odoo và kiểm tra

### Static files

#### **Static files** trong module Odoo bao gồm các tài nguyên như CSS, JavaScript, hình ảnh, và các file tĩnh khác được sử dụng để cải thiện giao diện người dùng và chức năng của module. Chúng giúp tùy chỉnh và nâng cao trải nghiệm người dùng trong Odoo.

#### Cấu Trúc Thư Mục Static

###### A screen shot of a computer Description automatically generated

###### src/css/: Chứa các file CSS để tùy chỉnh giao diện.

###### src/js/: Chứa các file JavaScript để thêm chức năng tương tác.

###### src/img/: Chứa hình ảnh như logo, icon, v.v.

###### description/: Được sử dụng cho các file mô tả module, như biểu tượng hiển thị trong Apps.

#### Sử Dụng Static Files Trong XML Views

##### Ví dụ: Thêm CSS và JavaScript Vào Form View

###### A computer code on a white background Description automatically generated

###### inherit\_id="web.assets\_backend": Kế thừa từ bộ tài sản backend của Odoo.

###### <link> và <script>: Thêm các file CSS và JavaScript vào tài sản của backend.

#### Quản Lý Tài Nguyên Tĩnh Trong JavaScript

##### Ví dụ: Sử Dụng JavaScript Trong Widget

###### A screen shot of a computer program Description automatically generated

###### odoo.define: Định nghĩa module JavaScript.

###### template: Tham chiếu đến template sử dụng trong widget.

### Templates

#### Định Nghĩa Template Qweb

##### A computer code with text Description automatically generated

##### id: Định danh duy nhất cho template.

##### inherit\_id: Kế thừa từ template gốc của Odoo (website.homepage).

##### xpath: Xác định vị trí chèn nội dung mới.

#### Thêm Template Vào Assets

##### A computer code with many colorful text Description automatically generated with medium confidence

## Kế thừa trong Odoo

### Kế thừa lớp (Model Inheritance)

### Kế thừa view và template (View & Template Inheritance)

#### Thêm Trường Vào Form View

##### A computer code with text Description automatically generated

##### inherit\_id: Tham chiếu tới ID của view gốc mà bạn muốn kế thừa.

##### xpath: Xác định vị trí để chèn thêm trường mới.

##### position="after": Chèn trường mới sau trường sale\_price.

### Kế Thừa Template Qweb

#### Thêm Phần Nội Dung Vào Homepage

##### A computer code with text Description automatically generated

QWeb là hệ thống templating của Odoo, sử dụng XML để định nghĩa giao diện người dùng. Dưới đây là các thẻ chính trong QWeb và cách sử dụng chúng:

**<template>**

Định nghĩa một template QWeb.

A white background with black text

Description automatically generated

**<t>**

Thẻ chính để thực hiện các logic trong QWeb.

**t-if**: Điều kiện hiển thị.

A white background with red and blue text

Description automatically generated

**t-foreach**: Lặp qua danh sách.

A white background with red and blue text

Description automatically generated

**t-call**: Gọi một template khác.



**t-set**: Đặt biến.



**<t t-jquery> và <t t-operation>**

Thao tác DOM bằng jQuery.

A close up of a white background

Description automatically generated

### Kế thừa JavaScript

#### Kế thừa JavaScript cho phép bạn mở rộng hoặc thay đổi chức năng của các widget hoặc module JavaScript hiện có.

#### Ví Dụ: Thay Đổi Hành Vi Của Button

##### A screen shot of a computer code Description automatically generated

##### odoo.define: Định nghĩa module JavaScript của Odoo.

##### FormController.include: Kế thừa và mở rộng FormController.

##### \_onButtonClicked: Thay đổi hành vi khi button được nhấn.

## Snippet Trong Odoo

### Snippet Là Gì?

#### Định Nghĩa: Snippet là các phần tử giao diện được thiết kế sẵn, cho phép người dùng tùy chỉnh và sắp xếp để xây dựng trang web theo ý muốn.

#### Ưu Điểm:

##### Dễ Dàng Sử Dụng: Không yêu cầu kiến thức lập trình.

##### Tùy Biến Linh Hoạt: Có thể điều chỉnh kích thước, màu sắc, nội dung.

##### Tiết Kiệm Thời Gian: Tăng tốc quá trình xây dựng trang web.

### Các Snippet Có Sẵn Trong Odoo

#### Odoo cung cấp nhiều snippet mặc định mà bạn có thể sử dụng ngay trong Website Builder:

##### Header: Thêm tiêu đề trang web.

##### Text Block: Chèn đoạn văn bản.

##### Image: Thêm hình ảnh.

##### Carousel: Tạo khung trượt ảnh.

##### Columns: Chia bố cục thành nhiều cột.

##### Video: Chèn video từ YouTube hoặc Vimeo.

##### Contact Form: Tạo biểu mẫu liên hệ.

### Tạo Snippet Tùy Chỉnh

#### Tạo Template QWeb: Định nghĩa giao diện của snippet trong file XML.

##### A computer code with text Description automatically generated with medium confidence

#### Đăng Ký Snippet Trong Assets: Thêm snippet vào tài sản frontend để Odoo nhận diện.

##### A close-up of text Description automatically generated

#### Tạo JavaScript và CSS (Nếu Cần):

##### custom\_snippet.js

###### A computer code with colorful text Description automatically generated

##### custom\_snippet.css

###### A screenshot of a computer code Description automatically generated