HƯỚNG DẪN THỰC HÀNH TUẦN 01

TÌM HIỂU PHẦN MỀM OPENERP 7.0

# HƯỚNG DẪN CÀI ĐẶT

## Cài đặt chương trình

* Hướng dẫn cài đặt phần mềm OpenERP:

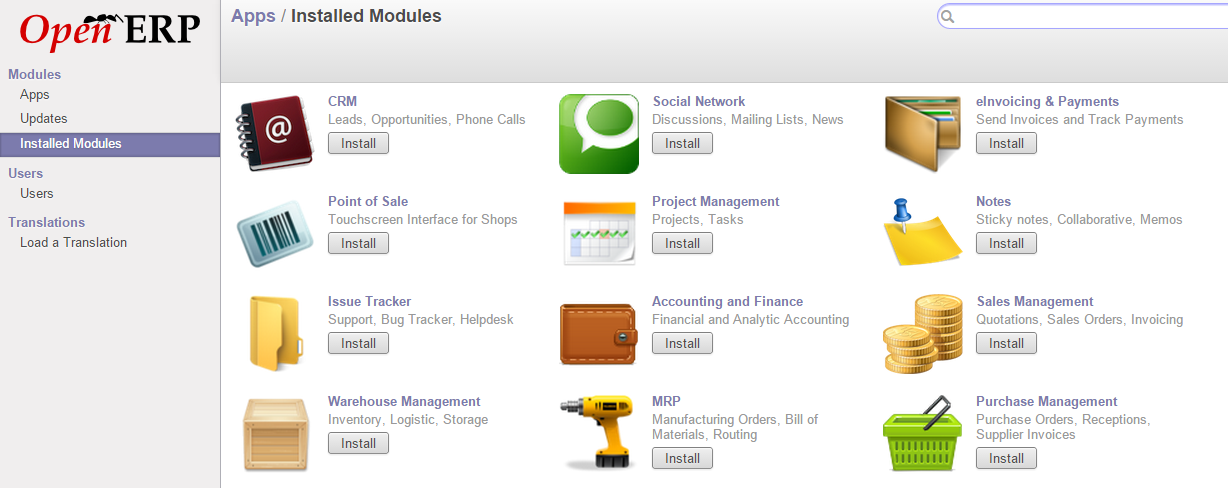
<https://doc.odoo.com/v7.0/install/windows/allinone/#downloading-openerp-all-in-one>

* Giao diện của OpenERP sau khi cài đặt thành công:



## Cài đặt module

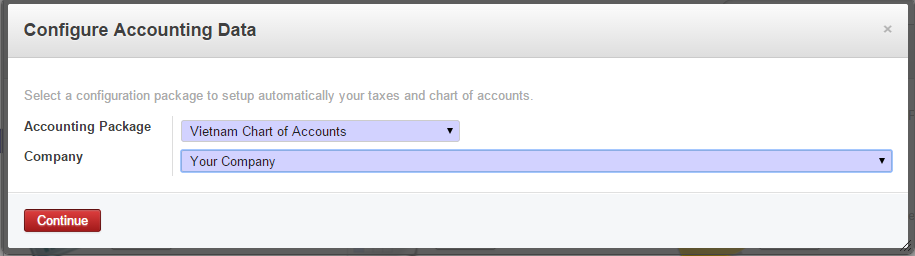
* Hiển thị danh sách tất cả các module hiện có (xem ở chế độ offline), chọn module cần cài đặt và chọn *Install*

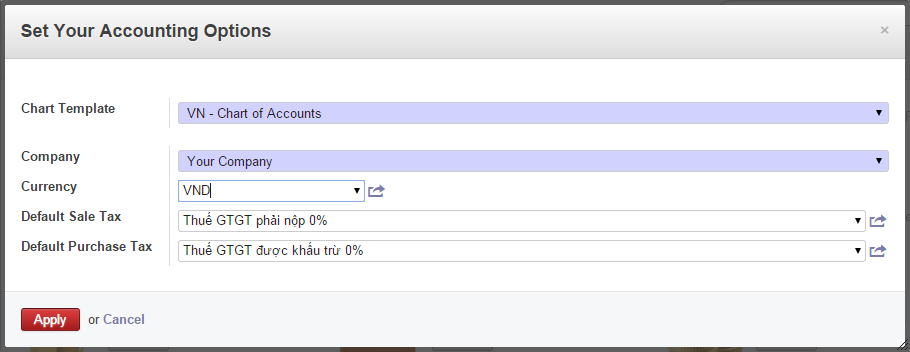


# BÀI TẬP ÁP DỤNG

## Cài đặt một số module sau:

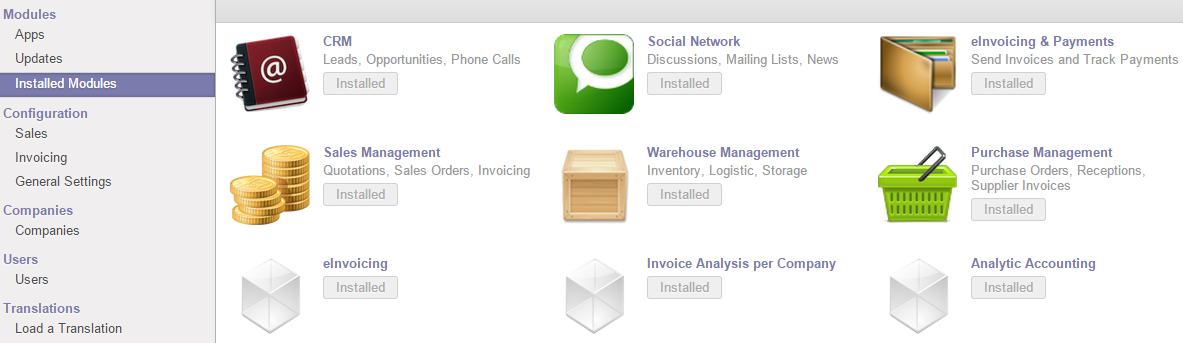
* CRM (*Customer Relationship Management*): quản lý quan hệ khách hàng
* Sale Management: quản lý bán hàng (khách hàng mua / đặt hàng tại công ty).
* Trong quá trình cài đặt sẽ bỏ qua việc cấu hình về kế toán vì thuộc về module Accounting & Finance
* Module được cài đặt kèm theo là eInvoicing & Payment: tạo hóa đơn điện tử và thanh toán





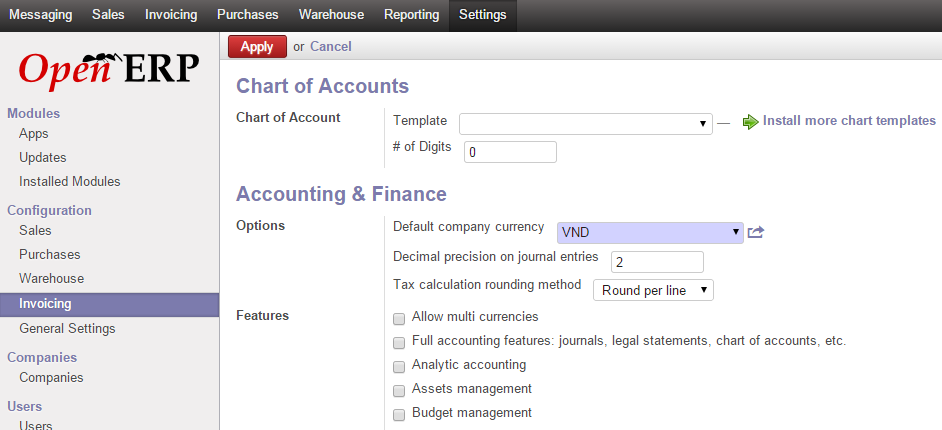
* Purchase Management: quản lý mua hàng (công ty mua hàng của nhà cung cấp)
* Module được cài đặt kèm theo là Warehouse Management: quản lý kho

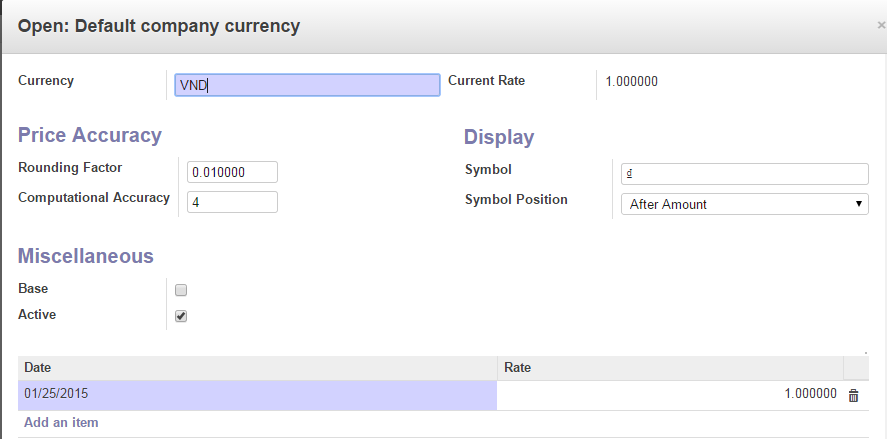
*Kết quả sau khi cài đặt các module như hình bên dưới:*



## Cấu hình tiền tệ (VND)

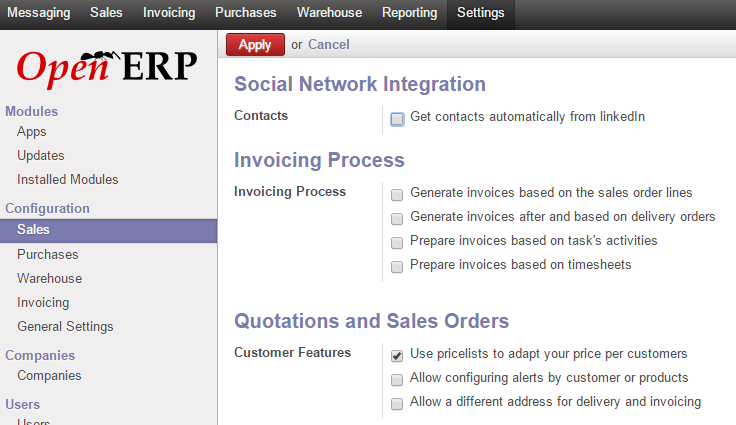
Chọn đơn vị tiền tệ mặc định cho công ty là VNĐ. Chọn *Settings* 🡪 *Invoicing* 🡪 *Default company currency: VND*



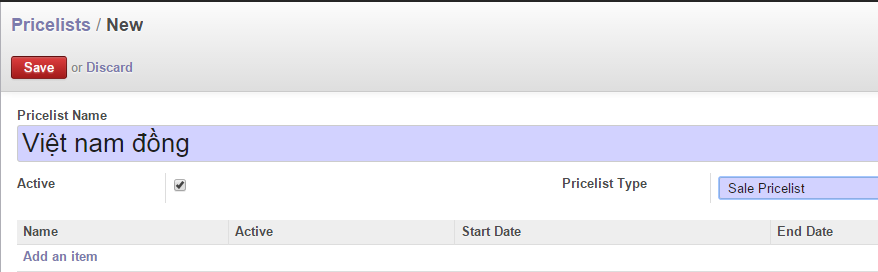


### Phần bán hàng (Sale)

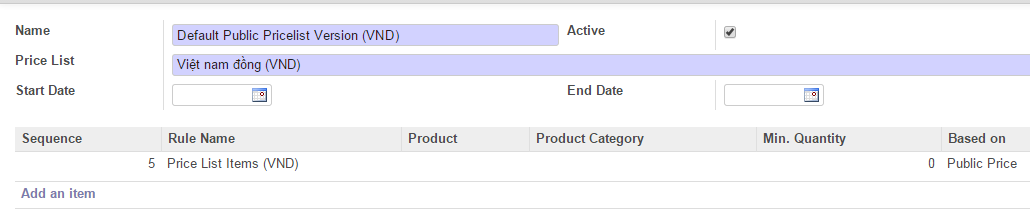
*Settings* 🡪 *Sales* 🡪 Check “*Use pricelists to adapt your price per customers*” 🡪 *Apply*

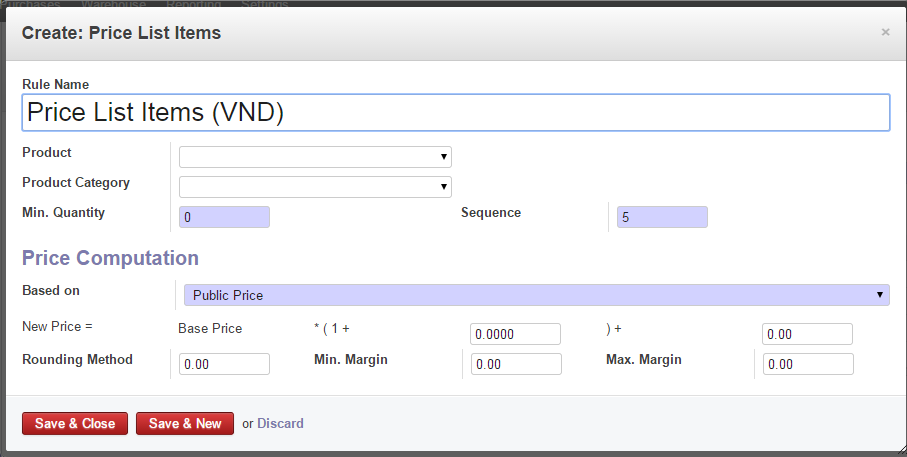


*Sales* 🡪 *Pricelists* 🡪 *Create*: để thêm 1 dòng đơn vị tiền tệ dùng cho giao dịch bán hàng là Việt nam đồng (VND)



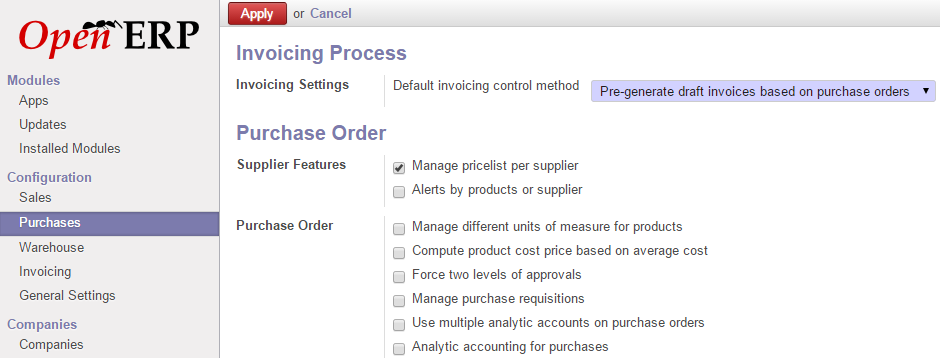
*Sales* 🡪 *Pricelist Version* 🡪 *Create*: thêm 1 dòng đơn vị tiền tệ dùng mặc định cho giao dịch bán hàng là Việt nam đồng (VND)





### Phần mua hàng (Purchase Management)

*Settings* 🡪 *Purchases* 🡪 Check “*Manage pricelist per supplier*” 🡪 *Apply*



Lưu ý: Các bước làm tương tự như trong phần Sales (xem như bài tập)

## Thêm khách hàng mới

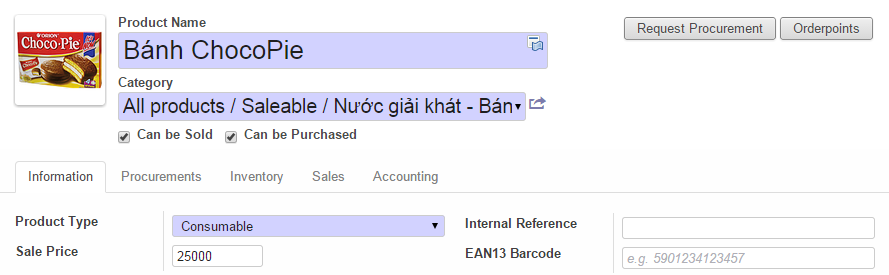
*Sales 🡪 Customers 🡪 Create*

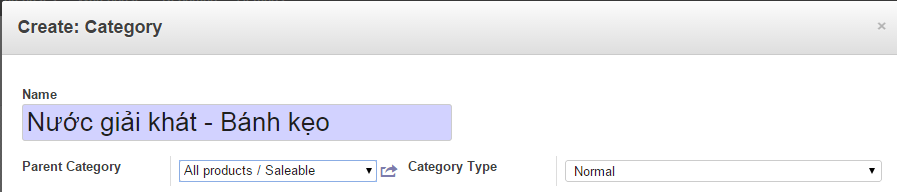


## Thêm các sản phẩm mới (dành cho bán hàng-Sales)

*Sales 🡪 Products 🡪 Create*

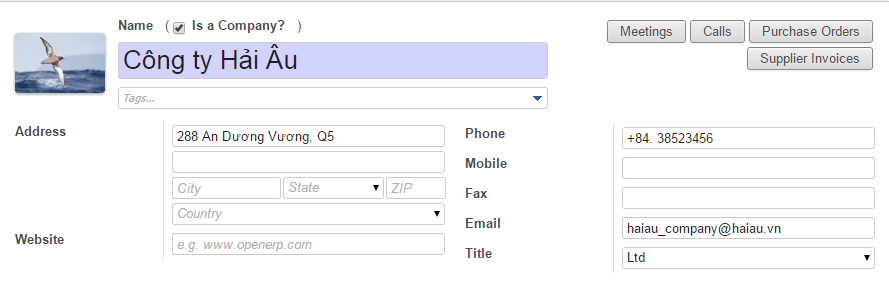
* Category: thuộc nhóm mặt hàng. Các sản phẩm mới nằm trong danh mục (category) Nước giải khát – Bánh kẹo
* Sale Price: giá bán
* Inventory 🡪 Quantity On Hand: số lượng mặt hàng đang có





## Thêm một nhà cung cấp mới

*Purchases 🡪 Suppliers🡪 Create*



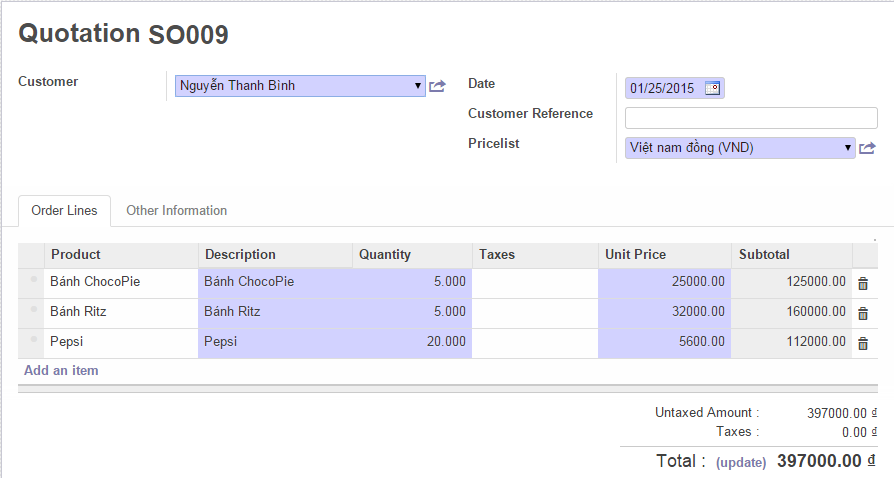
## Quy trình 1: Bán hàng (Sales)

Khách hàng đặt hàng (công ty ghi nhận thông tin, bảng báo giá) 🡪 công ty kiểm tra khả năng cung cấp🡪 Lập phiếu đặt hàng 🡪 Tình trạng giao hàng 🡪 Lập hóa đơn 🡪 Thanh toán.



### Ghi nhận thông tin đặt hàng, lập bảng báo giá

Lập phiếu đặt hàng: *Sales 🡪 Quotations🡪 Create*



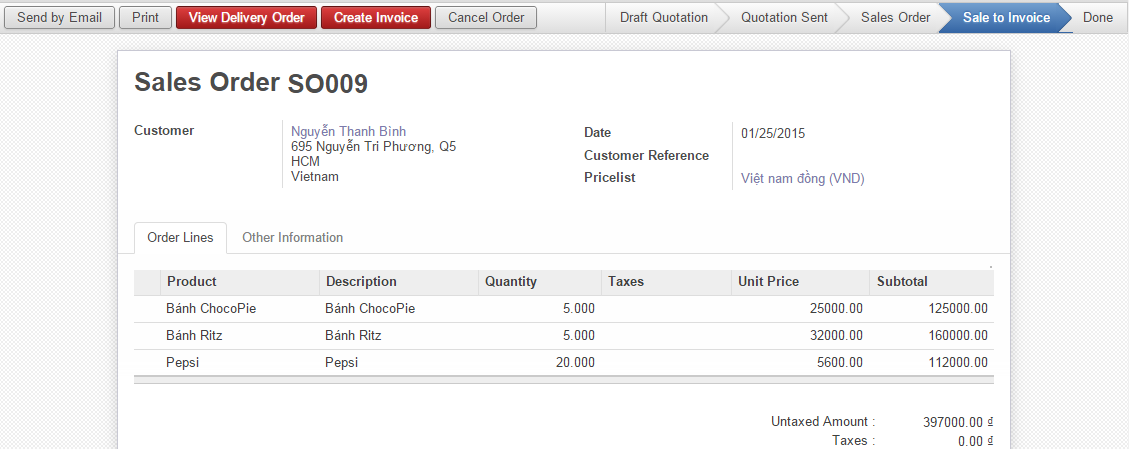
* *Pricelist*: chọn loại tiền tệ dùng để bán hàng

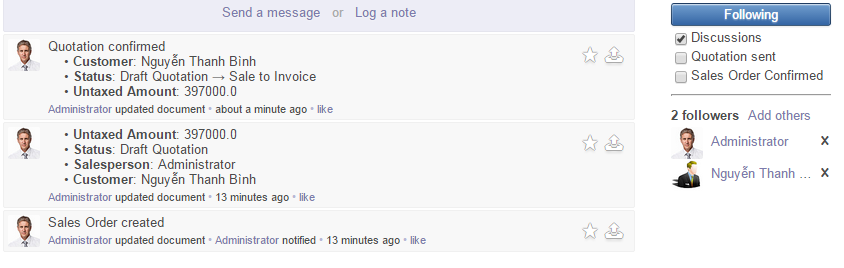
Sau khi thực hiện bước này, thông tin đặt hàng ở dạng nháp (draft)



### Xác định khả năng giao hàng

Nếu thông tin đặt hàng có thể đáp ứng được thì chọn bảng báo giá (Quotations) vừa ghi nhận 🡪 Chọn *Confirm Sales* để xác nhận bảng báo giá trở thành đơn đặt hàng chính thức. Bảng báo giá trở thành đơn đặt hàng chính thức như hình bên dưới.

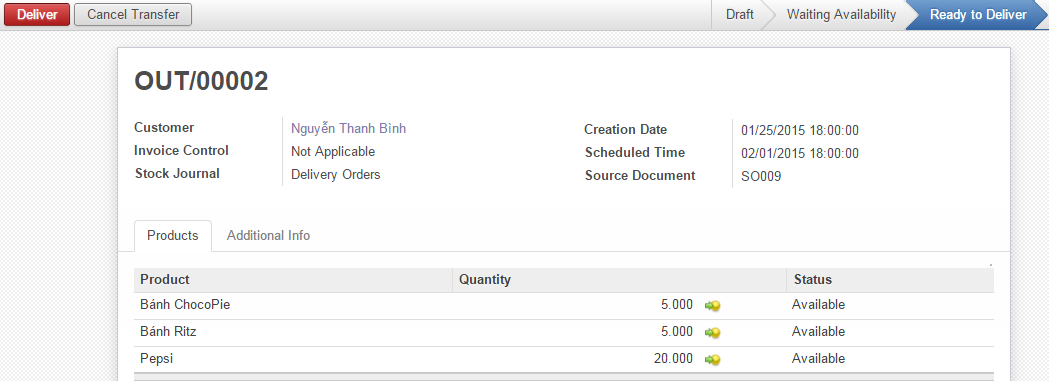




* *Following*: cho biết những người liên quan, chịu trách nhiệm với đơn đặt hàng này.

### Xem tình trạng giao hàng

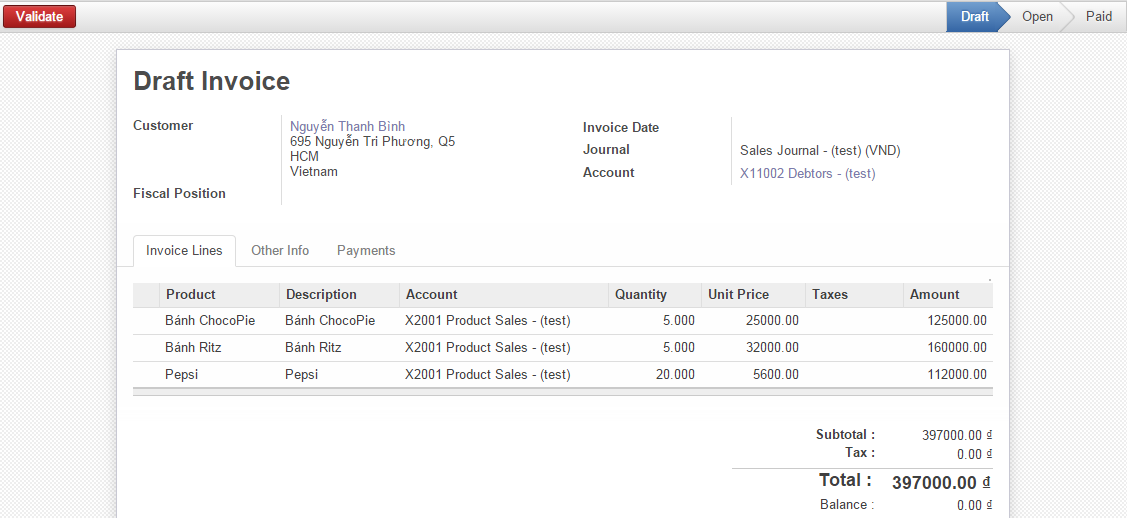
Trên mỗi đơn đặt hàng chọn *View Delivery Order,* để xem tình trạng giao hàng (số lượng có đáp ứng đủ hay không) và lập phiếu giao hàng.



Sau khi chọn chức năng *Deliver* thì lập tức số lượng của các mặt hàng được cập nhật lại.

### Lập hóa đơn

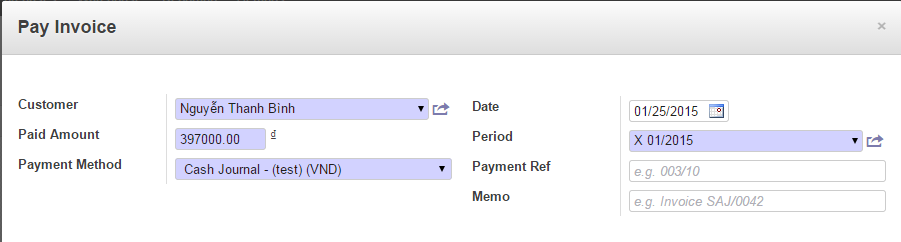
Chọn chức năng *Create Invoice* trên mỗi đơn hàng 🡪 Tạo ra hóa đơn nháp (draft) 🡪 Kiểm tra lại thông tin hóa đơn nếu xác nhận là đúng thì chọn *Validate*  🡪 Thực hiện việc thanh toán



Sau khi *Validate* thì hóa đơn chuyển sang trạng thái *Open* để khách hàng thanh toán

### Đăng ký thanh toán

Trên hóa đơn vừa lập chọn *Register Payment*

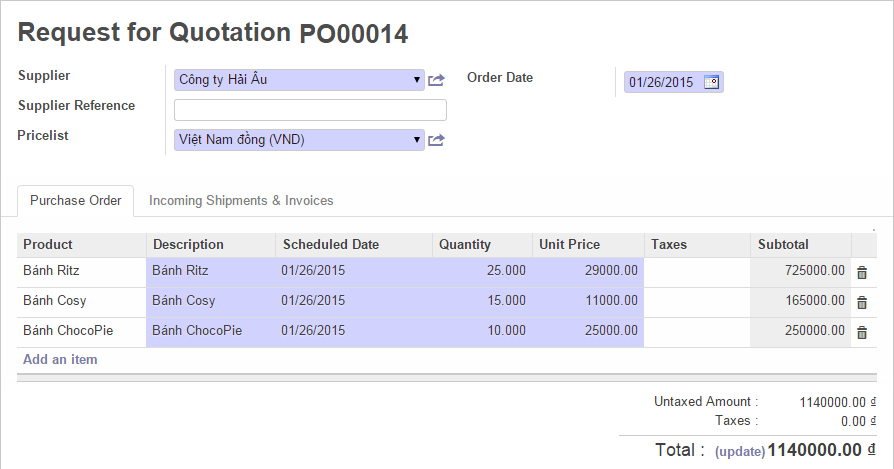


## Quy trình 2: đặt hàng từ nhà cung cấp

Đặt hàng từ nhà cung cấp 🡪 kiểm tra kho (warehouse) 🡪 vận chuyển giao hàng

### Lập dự toán các mặt hàng cần mua

*Purchases 🡪 Quotations 🡪 Create*



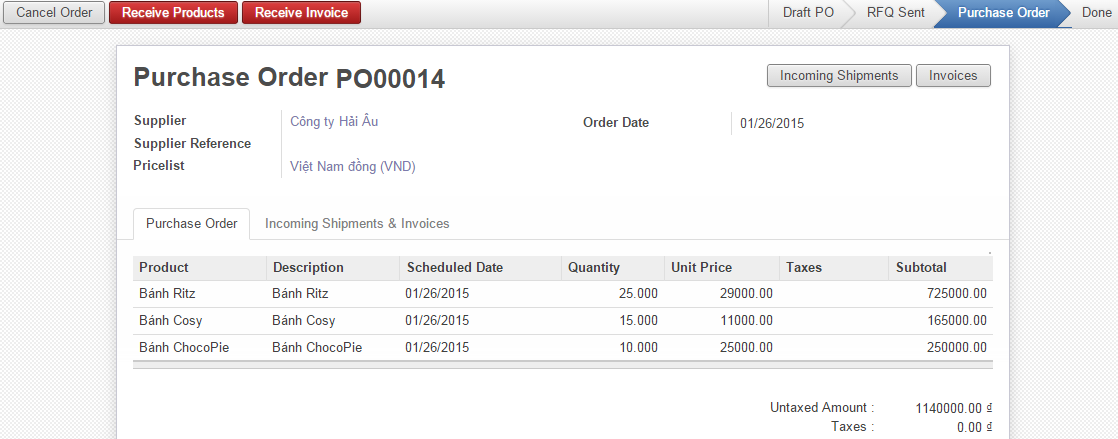
### Xác nhận đặt hàng

Chọn *Quotations* tương ứng 🡪 chọn *Confirm* *Order*

### Xác nhận đã nhận được mặt hàng mua

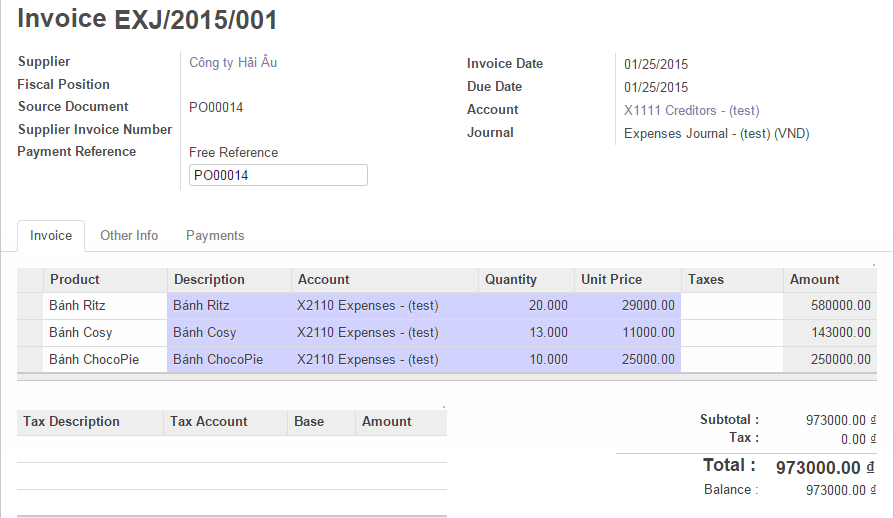
Chọn *Receive Products*

Trong trường hợp xem chi tiết thông tin giao hàng thì vào Incoming Shipments. Việc giao hàng có thể giao nhiều lần, khi đó phải thực hiện việc xác nhận số lượng mặt hàng đã đặt nhiều lần.

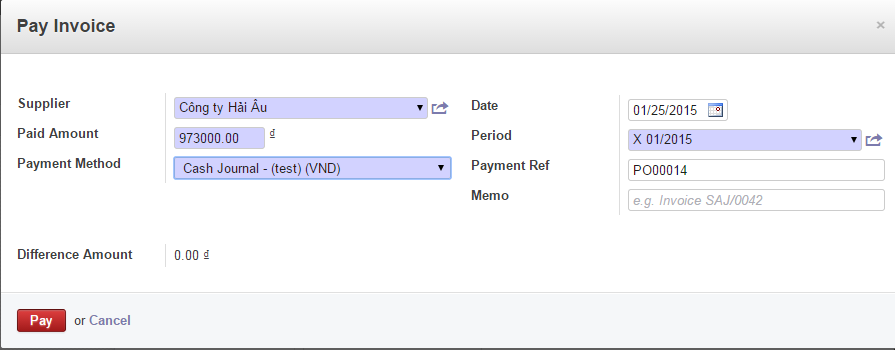


### Xác nhận đã nhận hóa đơn

Chọn *Receive Invoice*



### Thanh toán



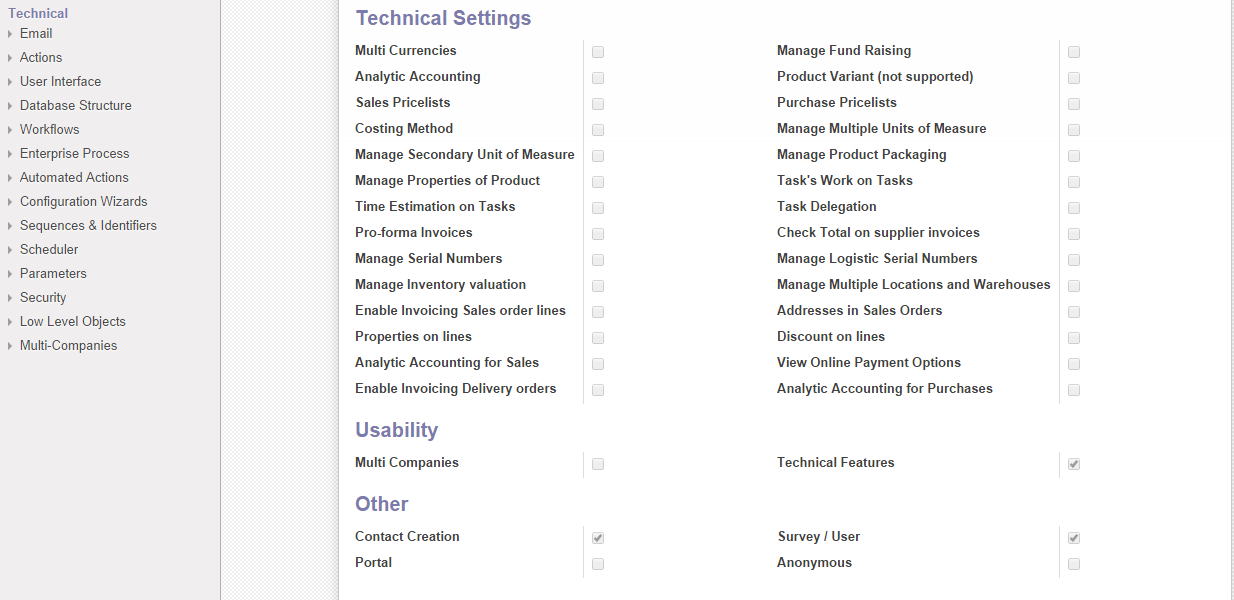
HƯỚNG DẪN THỰC HÀNH TUẦN 02

CHỈNH SỬA (CUSTOMIZE) MODULES TRONG OPENERP 7.0

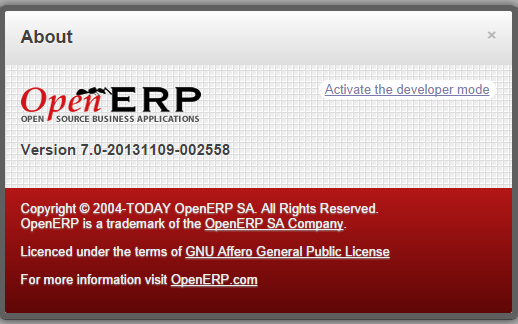
# CẤU HÌNH

Để bật chế độ có thể tùy chỉnh module (developer mode) thực hiện các bước như sau:

* Bước 1: *Setting 🡪 Users 🡪 Vai trò tương ứng 🡪 Access Rights 🡪 Technical Features*



* Bước 2: *Administrator🡪 About OpenERP* để kích hoạt chế độ chỉnh sửa



# CÁCH THỰC HIỆN

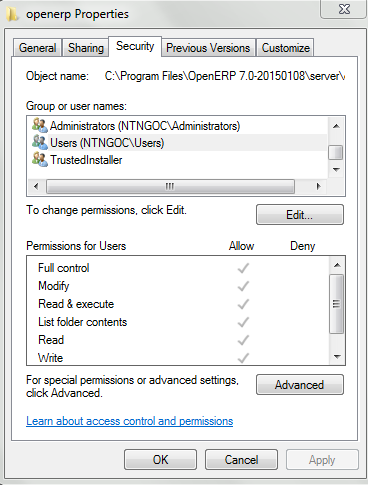
Việc thêm fields mới vào module có sẵn, cần thực hiện các bước sau:

* Chỉnh sửa file định nghĩa module (*file \*.py*)
* Thêm field mới vào bảng tương ứng trong CSDL
* Chỉnh sửa giao diện (*file \*.xml*) để hiện thị field mới

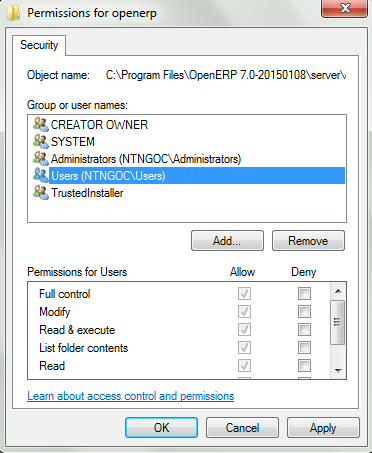
Thông tin đường dẫn thư mục chứa các file cần chỉnh sửa:

* File định nghĩa module: C:\Program Files\OpenERP 7.0-20150108\server\openerp\addons\
* Thường file sẽ mang tên *<tên module>.py*

Lưu ý: nếu sau khi chỉnh sửa xong trong file python mà không cho lưu vì không đủ quyền hạn thì cần click phải vào folder *openerp 🡪Properties🡪 Security*



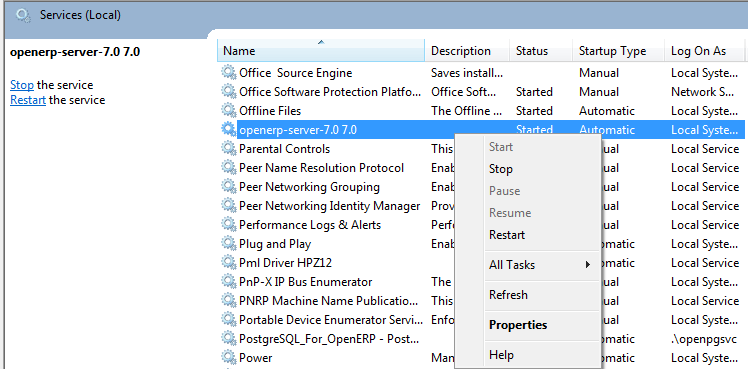
*Edit* 🡪 thêm quyền *Full Control* cho user đang sử dụng hiện hành



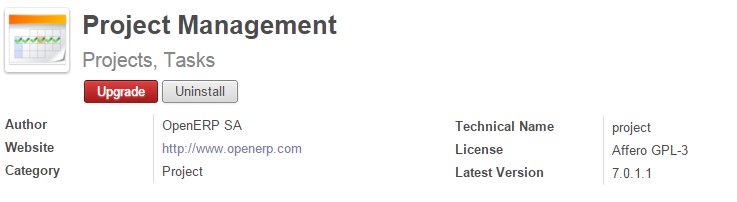
* Bảng lưu trữ thông tin trong CSDL(PostgreSQL) là project\_project
* File định nghĩa giao diện: C:\Program Files\OpenERP 7.0-20150108\server\openerp\addons\
* Thường file sẽ mang tên *<tên module>\_view.****xml***

Sau khi chỉnh sửa xong, cần khởi động lại server (services.msc 🡪 openerp server 7.0 🡪 Restart) và Upgrade module.

**Tips: Kiểm tra file python vừa chỉnh sửa (\*.py) có được biên dịch lại thành file (\*.pyc) tương ứng hay không, bằng cách xem thời gian tạo của 2 file đó. Nếu file \*.py không được biên dịch có nghĩa là phần sửa bị lỗi**



* Installed Modules 🡪 Upgrade module



**Lưu ý: Nếu chỉ chỉnh sửa lại file giao diện xml thì chỉ cần upgrade lại module.**

# HƯỚNG DẪN & BÀI TẬP

Hãy bổ sung thêm các fields vào giao diện ban đầu (Hình 5) để lần lượt thành giao diện như bên dưới:

## Customize 1: thêm một field mới là Project Location có kiểu dữ liệu là char

Bước 1: chỉnh sửa ở CSDL PostgreSQL, bằng cách tạo một cột có tên tương ứng là “project\_location” trong bảng project\_project

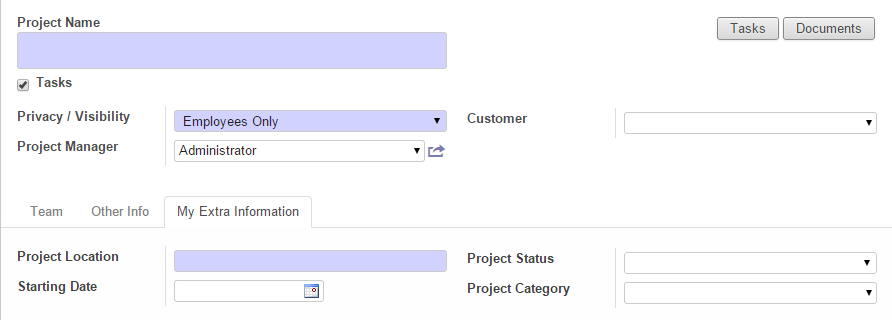
Lệnh SQL:

*ALTER TABLE project\_project ADD COLUMNS* ***project\_location*** *VARCHAR(50)*

Bước 2: chỉnh sửa ở file python (xác định đúng class **project.project** và ở phần \_**column**)

**'project\_location':fields.char('Project Location',size=50,required=True,translate=False),**

Bước 3: cập nhật lại giao diện <field name="**project\_location**"/>. Lưu ý xem vị trí đặt cho phù hợp.



## Customize 2: thêm một field mới là Project Status có kiểu dữ liệu là selection

Bước 1: chỉnh sửa ở CSDL PostgreSQL, bằng cách tạo một cột có tên tương ứng là “my\_status” trong bảng project\_project

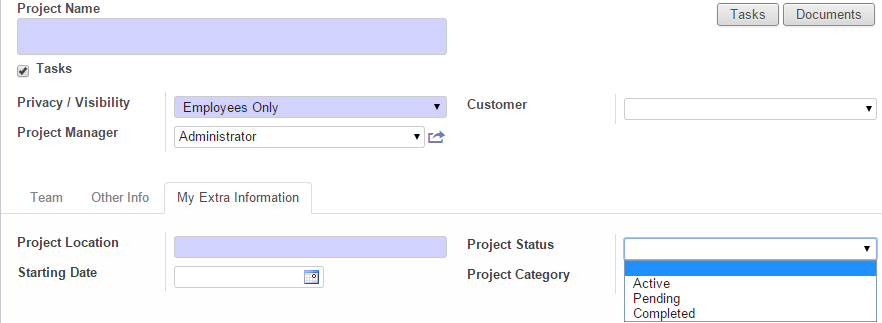
Lệnh SQL:

*ALTER TABLE project\_project ADD COLUMNS* ***my\_status*** *VARCHAR(50)*

Bước 2: chỉnh sửa ở file python (xác định đúng class **project.project** và ở phần \_**column**)

**'my\_status':fields.selection([('active','Active'),('pending','Pending'),('completed','Completed')],'Project Status'),**

Bước 3: cập nhật lại giao diện <field name="**my\_status**"/>. Lưu ý xem vị trí đặt cho phù hợp.



## Customize 3: thêm một field mới là Starting Date có kiểu dữ liệu là datetime

Bước 1: chỉnh sửa ở CSDL PostgreSQL, bằng cách tạo một cột có tên tương ứng là “project\_datetime” trong bảng project\_project

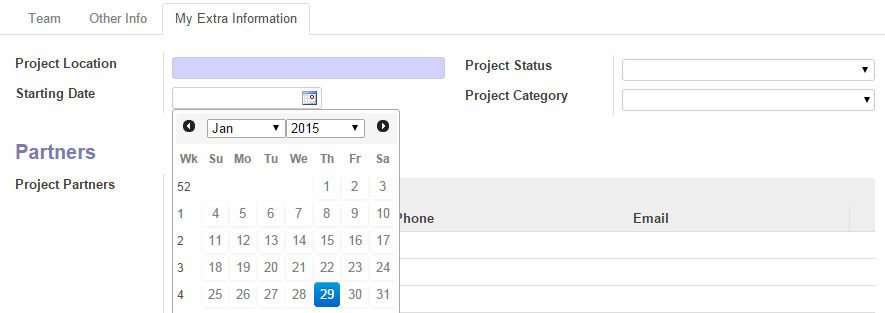
Lệnh SQL:

*ALTER TABLE project\_project ADD COLUMNS* ***project\_datetime*** *VARCHAR(50)*

Bước 2: chỉnh sửa ở file python (xác định đúng class **project.project** và ở phần \_**column**)

**'project\_datetime':fields.datetime('Starting Date'),**

Bước 3: cập nhật lại giao diện <field name="**project\_datetime**"/>. Lưu ý xem vị trí đặt cho phù hợp.



## Customize 4: thêm một field mới là My Note có kiểu dữ liệu là text

Bước 1: chỉnh sửa ở CSDL PostgreSQL, bằng cách tạo một cột có tên tương ứng là “my\_note” trong bảng project\_project

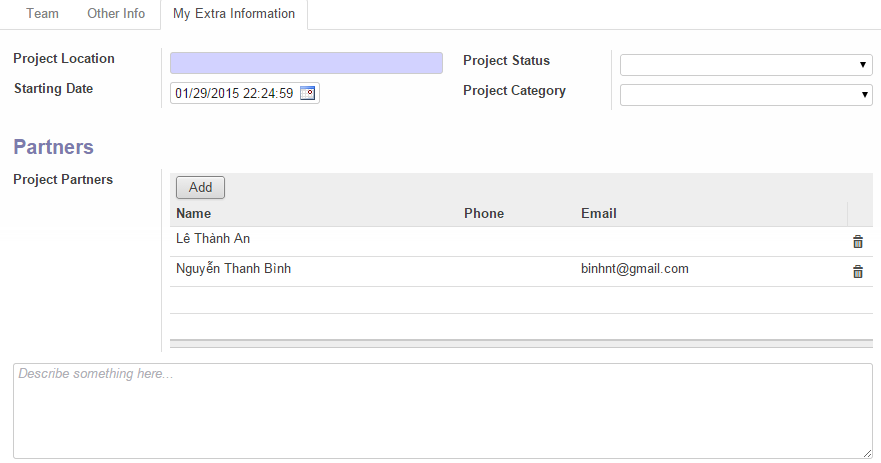
Lệnh SQL:

*ALTER TABLE project\_project ADD COLUMNS* ***my\_note*** *VARCHAR(200)*

Bước 2: chỉnh sửa ở file python (xác định đúng class **project.project** và ở phần \_**column**)

**'my\_note':fields.text('My Note'),**

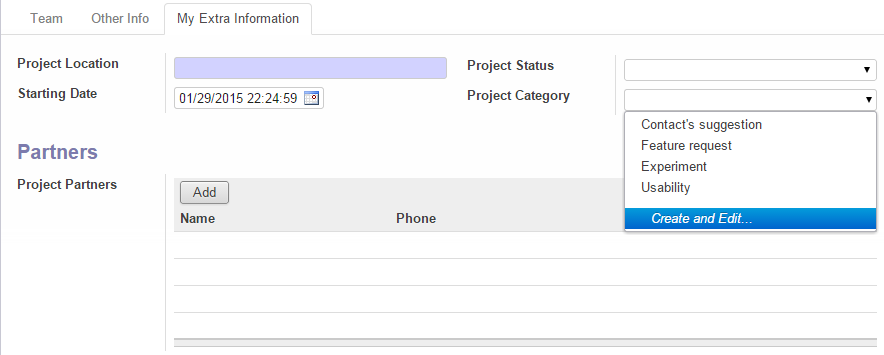
Bước 3: cập nhật lại giao diện <field name="**my\_note**"/>. Lưu ý xem vị trí đặt cho phù hợp.



## Customize 5: thêm một field mới là Project Category tham chiếu đến bảng project\_category đã có sẵn, chọn kiểu dữ liệu là many2one

**<Làm tương tự như trên>**

**'project\_category':fields.many2one('project.category','Project Category', help="Chose the project category"),**



**Mở rộng:**

Trong trường hợp muốn kết nối với một bảng chưa có, thì cần tạo 1 class mới trong file python và sử dụng kết nối many2one thì class đó sẽ tương ứng với 1 bảng “thật” sẽ được tự động tạo trong CSDL PostgreSQL.

Ví dụ tạo class project.location

# projecct\_location

class **project\_location**(osv.osv):

\_name = "project.location"

\_description = "Location of project..."

\_columns = {

'name': fields.char('Location',size=50,required=True)

}

Sau đó thêm 1 field mới trong file **project.project** như sau:

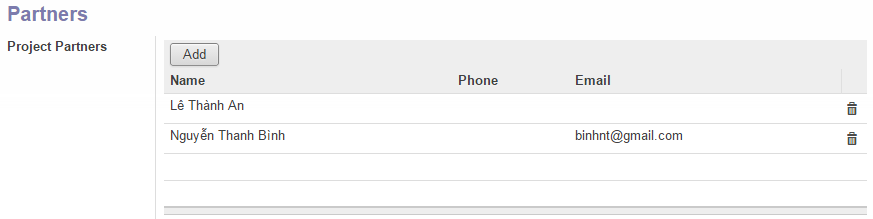
**'my\_location\_list':fields.many2one('project.location','Location List', help="Choose location from list")**

## Customize 6: thêm một field mới là Project Partners có kiểu dữ liệu là many2many

Ví dụ, muốn thể hiện 1 project có nhiều người tham gia (partner), và 1 partner cũng có thể tham gia nhiều project, chúng ta sử dụng liên kết **many2many**.

Đối với trường hợp many2many thì không cần tạo 1 cột mới trong PostgreSQL (bỏ qua bước 1) vì hệ thống sẽ tự động phát sinh ra 1 bảng mới project\_partner\_rel như đã mô tả với 2 thuộc tính là project\_id và partner\_id. Như vậy, chỉ cần định nghĩa trong file python (bước 2), và khai báo trong file xml để hiện thị trên giao diện (bước 3).

**'project\_partner':fields.many2many('res.partner','project\_partner\_rel','project\_id','partner\_id', 'Project Partners', help="Project's partners are partners who get involved in this project.")**



HƯỚNG DẪN THỰC HÀNH TUẦN 03

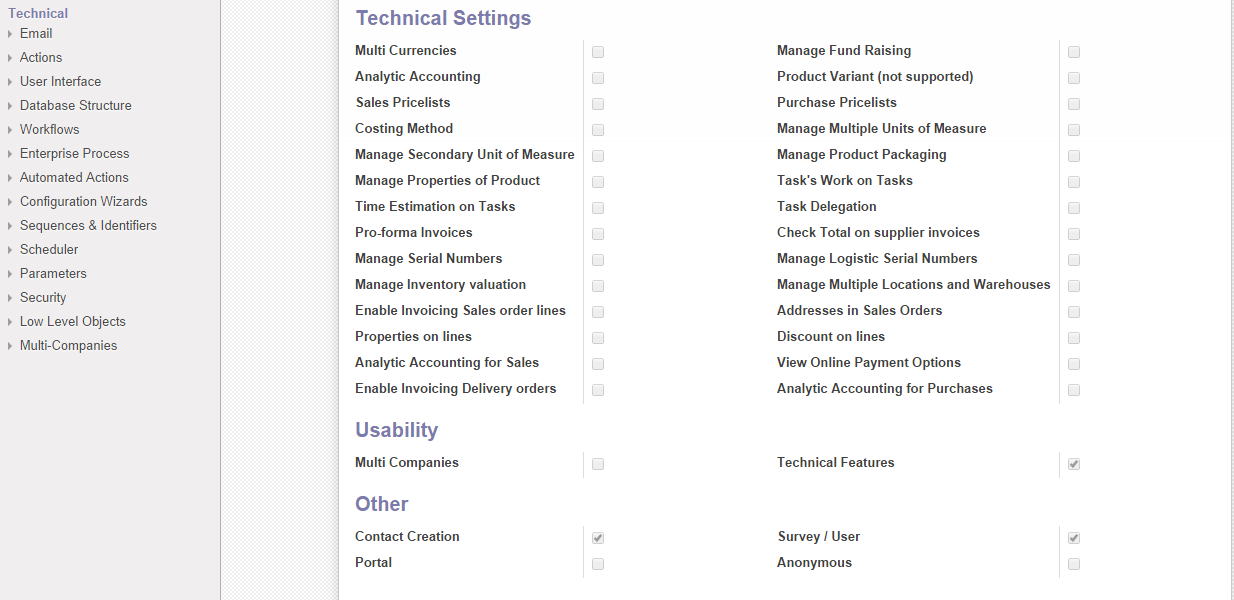
TẠO MODULES MỚI TRONG OPENERP 7.0

# CÁCH THÊM MODULE MỚI

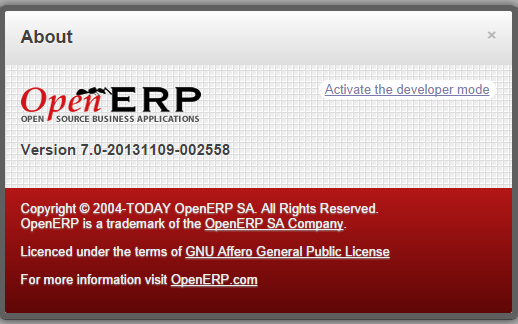
## Cấu hình

Để bật chế độ có thể tùy chỉnh module (developer mode) thực hiện các bước như sau:

* Bước 1: *Setting 🡪 Users 🡪 Vai trò tương ứng 🡪 Access Rights 🡪 Technical Features*

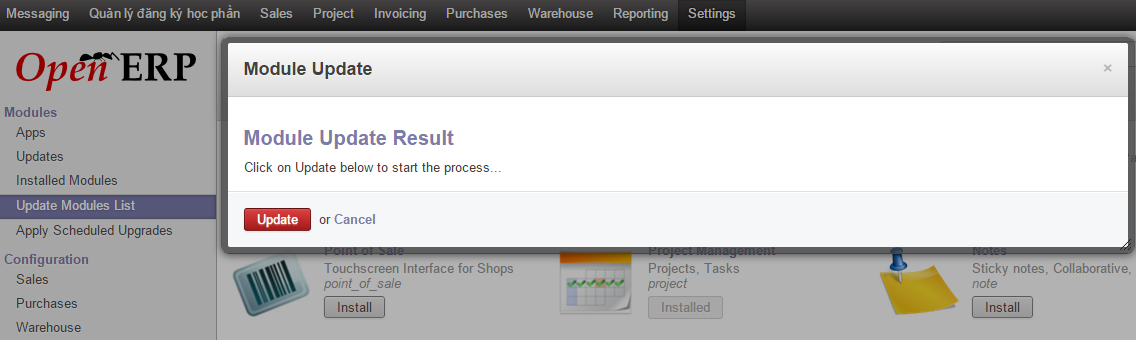


* Bước 2: *Administrator🡪 About OpenERP* để kích hoạt chế độ chỉnh sửa

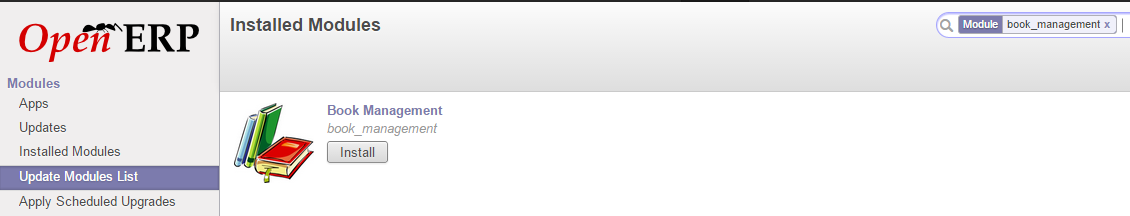


## Hướng dẫn test thử module mới

* Copy folder module mới tạo (Vd: *book\_management*) vào thư mục *addons* (C:\Program Files\OpenERP 7.0-20150108\server\openerp\addons\)
* Logout sau đó login vào lại OpenERP
* Vào *Settings* 🡪 *Update Modules List* 🡪 *Update*



* Sau khi update modules list xong thì gõ search tên module sẽ thấy xuất hiện module mới để cài đặt



# CÁCH TẠO MODULE MỚI

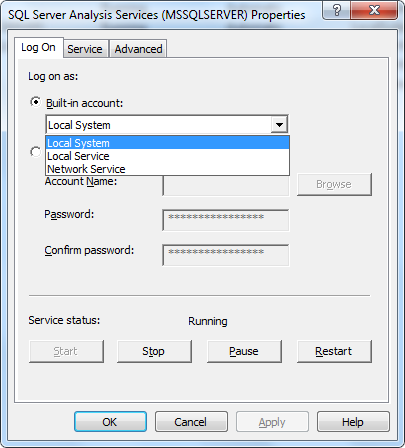
* Tạo 1 folder mới tên: book\_management
* Trong folder vừa tạo ở trên luôn luôn có các file như sau:
* \_\_init\_\_.py: file này làm nhiệm vụ import các file python sẽ được sử dụng (trong ví dụ là book.py, thì sẽ khai báo là **import book**). Lưu ý: tên file không được thay đổi phải đúng như quy định kể cả các dấu gạch chân \_\_
* \_\_openerp\_\_.py: file này dùng để mô tả về module, với một số thuộc tính như:
* Name: tên module
* Author: tác giả
* Description: mô tả tóm tắt về module
* Category: module sẽ nằm trong mục nào
* Depends: khai báo những module liên quan, những module kế thừa (mặc định là module base)
* Demo\_xml: nếu muốn có dữ liệu demo trước cho module thì khai báo những file xml demo
* Update\_xml: chỉ ra các file xml nào sẽ được cập nhật khi cập nhật module
* Installable: là cho phép cài đặt, mặc định là true
* File python (trong ví dụ là file book.py): đây là file chứa các khai báo lớp, khai báo methods.
* File xml: dùng để tạo menu, tạo giao diện form hay tree view cho module

Cách khai báo chi tiết trong từng file sẽ xem chi tiết trong module book\_management đính kèm.

HƯỚNG DẪN THỰC HÀNH TUẦN 04

TẠO CUBE BẰNG SSAS – TRUY VẤN MDX

# Lưu ý: Cần mở SQL server Analysis services

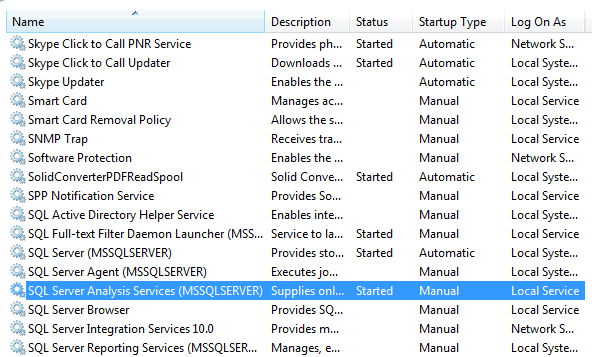


Chọn Local system

# TẠO CUBE

## Kiểm tra cấu hình

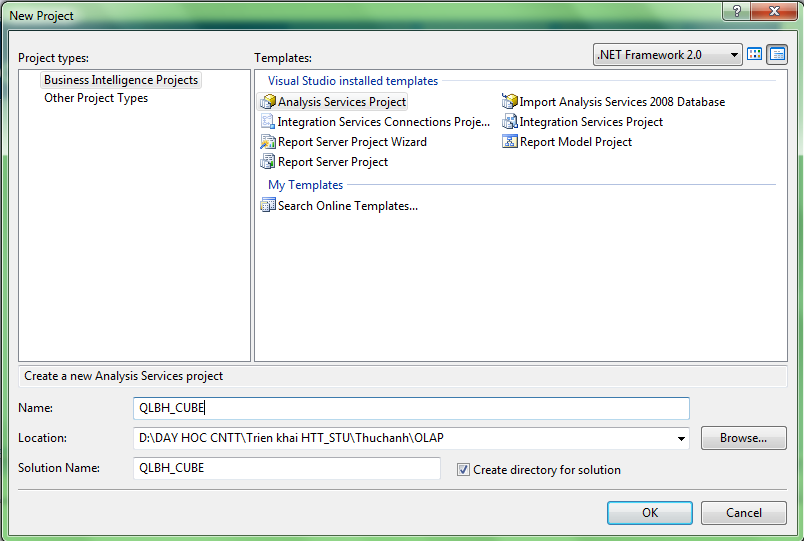
* Gõ *services.msc* trong ô Search, kiểm tra service SQL Server Analysis Services đã được Start



## Thực hiện

Việc xây dựng Cube với SSAS được thực hiện bằng công cụ SQL Server Business Intelligence Development Studio (BIDS).

* Bước 1: Khởi động BIDV 🡪 New Project 🡪 Analysis Services Project 🡪 Đặt tên project



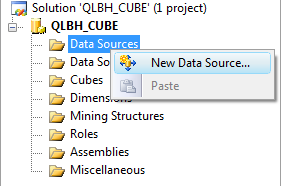
Để tạo Cube gồm 3 bước chính như sau:

+ định nghĩa Data Sources

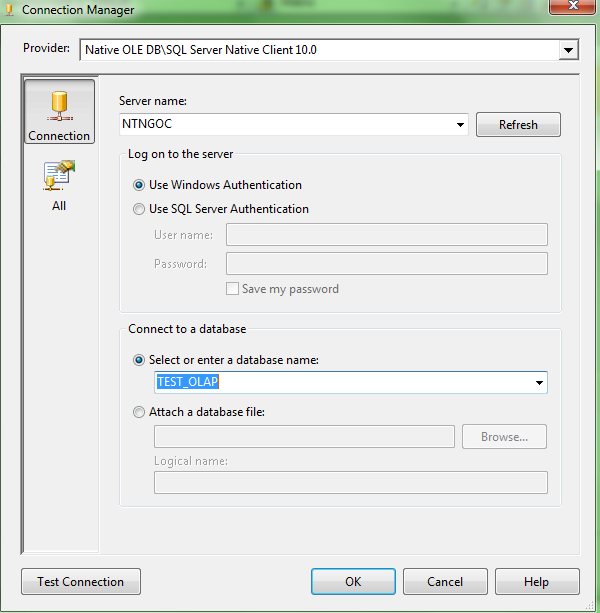
+ định nghĩa Data Source Views

+ tạo Cube

* Bước 2: Định nghĩa Data Sources: chỉ ra sẽ sử dụng CSDL nào



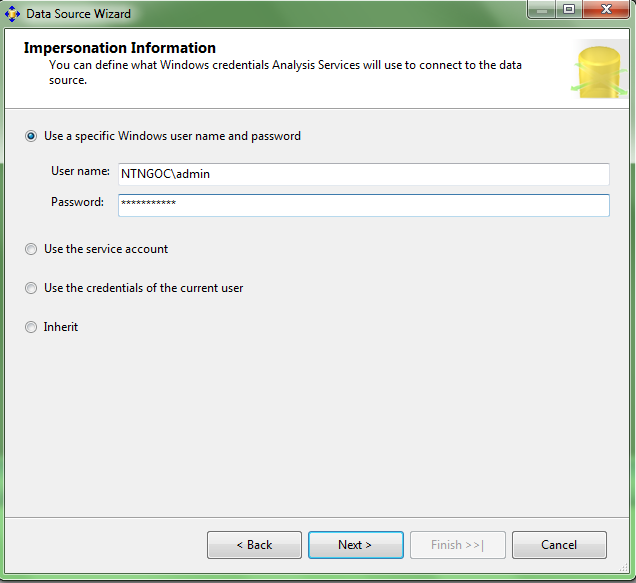
2.1. Định nghĩa connection



2.2. Định nghĩa bảo mật

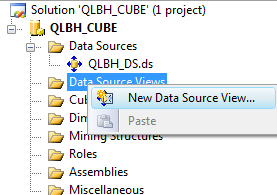
+ Username: tên máy\role

+Password: mật khẩu đăng nhập vào máy tính



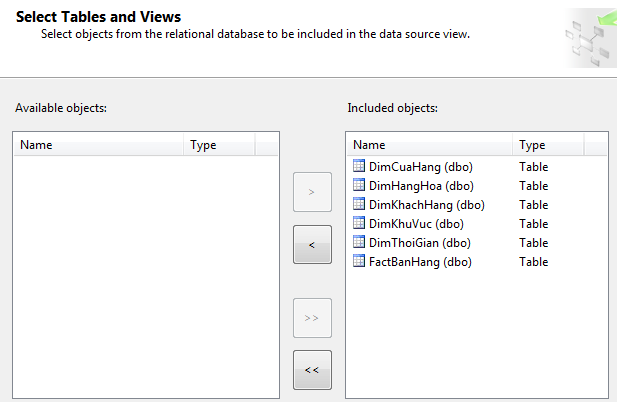
**Lưu ý: Trong trường hợp không rõ username, password thì có thể chọn Inherit**

* Bước 3: Định nghĩa Data Source Views: chỉ ra sẽ sử dụng những bảng nào



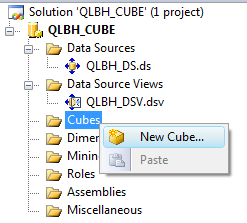
Chọn tương ứng với Data Source vừa tạo ở bước 2 (mặc định) 🡪 Next

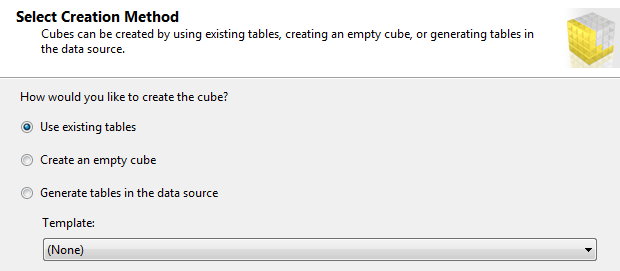
Chọn các bảng sẽ sử dụng



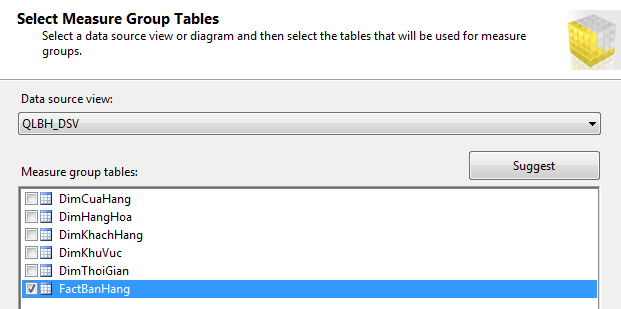
Đặt tên cho Data Source Views 🡪 Next

* Bước 4: Tạo Cube

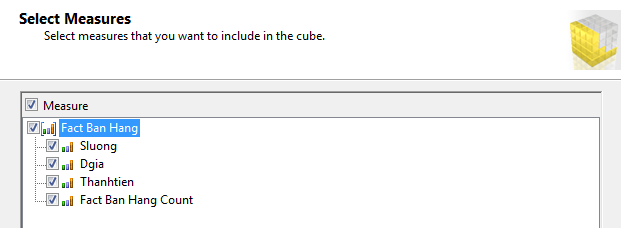




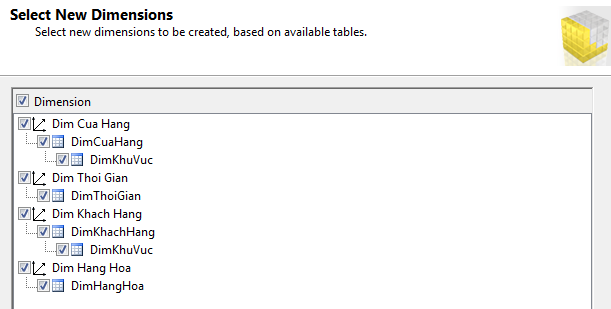
Chọn bảng chứa Measures



Chọn các yếu tố cần phân tích 🡪 Next

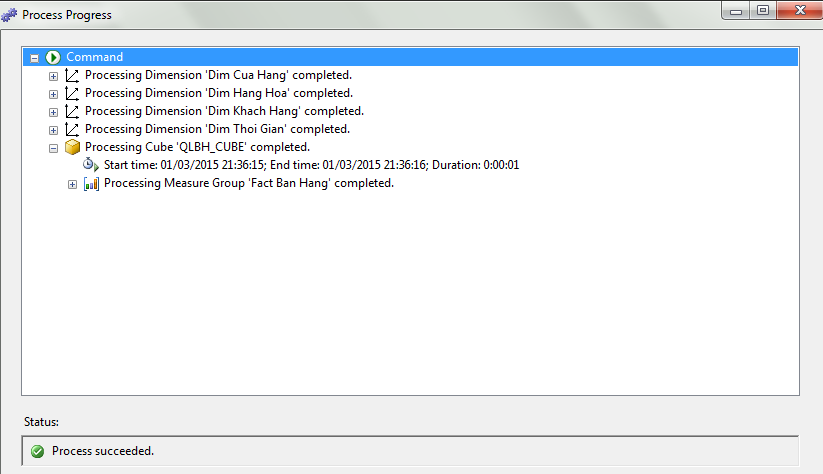


Chọn các tiêu chí (Dimensions)



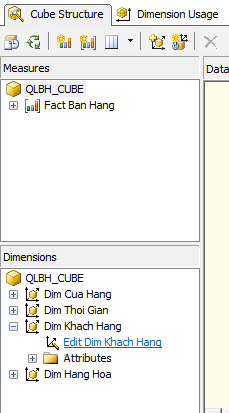
Đặt tên Cube 🡪 Next

Chọn cube vừa tạo 🡪 phải chuột 🡪 Process 🡪 Chọn Run. Phải đảm bảo Cube được build thành công như hình bên dưới

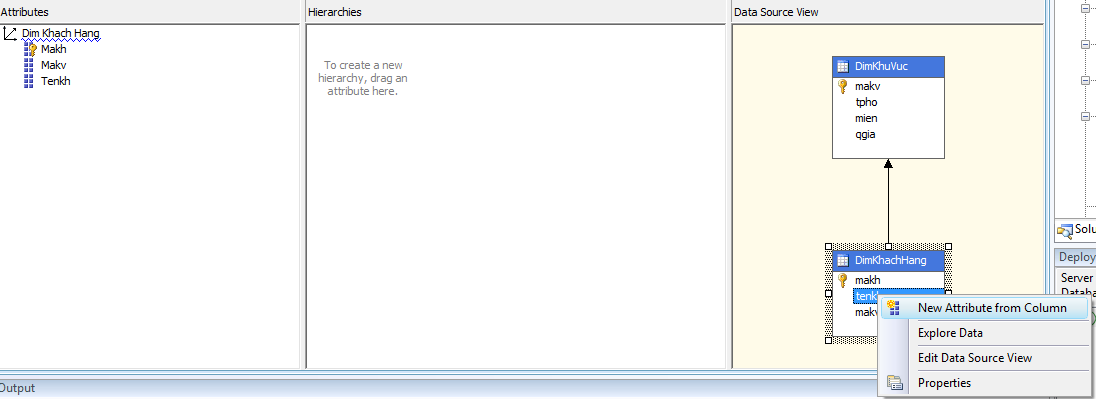


4.1. Định nghĩa các attribute để hiện thị cho việc gom nhóm

Chọn Cube Structure 🡪 Dimensions 🡪 Chọn Dimension cần chỉnh sửa 🡪 Edit Dimension …



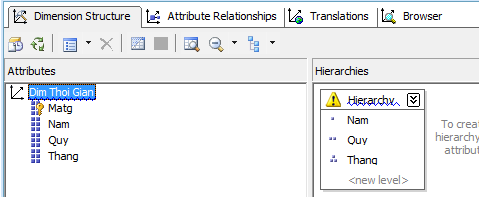
Click phải chuột trên thuộc tính cần hiển thị 🡪 New Attribute From Column 🡪 Save



4.2 Định nghĩa Hierarchy

Ví dụ muốn tạo phân cấp Năm 🡪 Qúy 🡪 Tháng chọn DimThoiGian và thêm New Attribute From Column cho 3 thuộc tính Năm, Qúy, Tháng (như bước vừa mô tả ở trên).

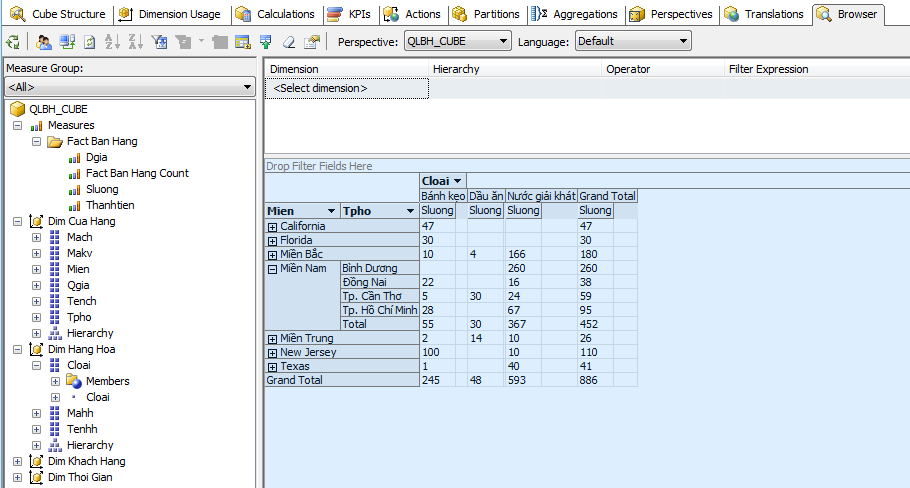
Kéo thả thuộc tính năm vào khung Hierarchies kế bên, tương tự cho Qúy, và Tháng. Lưu ý khi kéo thả phải đúng các thứ tự phân cấp 🡪 Save



* Bước 5: Xem và truy vấn trên Cube

Click phải chuột vào Cube 🡪 Process để build lại cube

Vào Browse 🡪 Kéo thả các thuộc tính vào dòng và cột tương ứng



0) Giới thiệu CSDL

* Nhắc lại FactTable gồm danh sách các Dims và các Measures

1) Minh họa SQL 99

* WITH ROLLUP
* WITH CUBE
* PIVOT

🡺Hạn chế của SQL 99

2) Xây dựng Cube bằng SSAS

* Khôi dữ liệu tổng hợp được thể hiện 1 cách trực quan, và sau đó sẽ có thể thực hiện các câu truy vấn trên cube đó.
* 1 Cube là cài đặt của 1 FactTable
* 1 ứng dụng có thể có nhiều cube, mỗi cube có thể liên quan đến 1 lãnh vực (tài chính, kinh doanh, nhân sự, …).

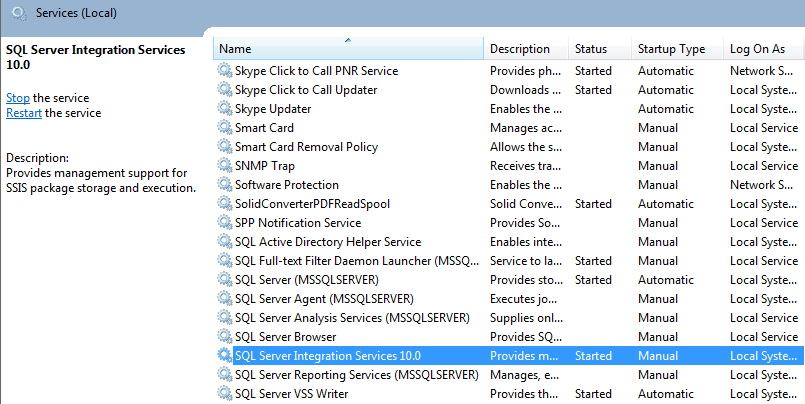
3) Truy vấn MDX trên Cube

HƯỚNG DẪN THỰC HÀNH TUẦN 05

THỰC HÀNH ETL VỚI SQL SERVER INTEGRATION SERVICE

# CẤU HÌNH

* Gõ *services.msc* trong ô Search, kiểm tra service SQL Server Analysis Services đã được Start

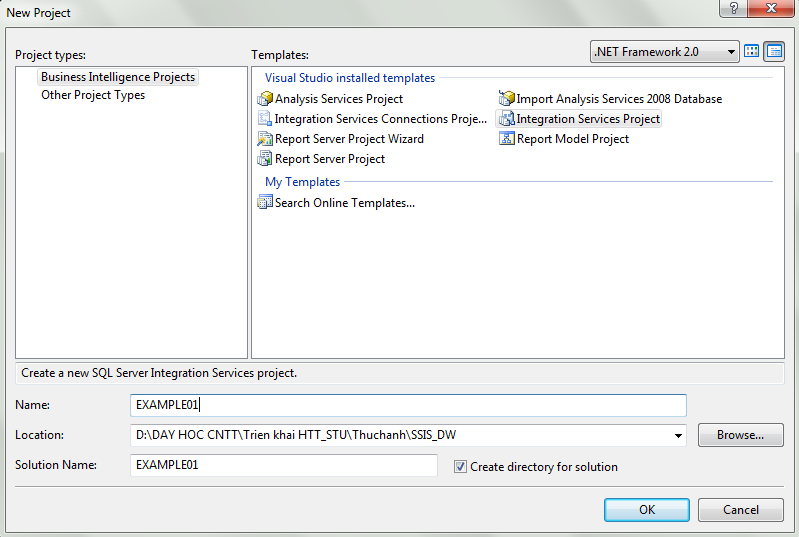


# THỰC HIỆN

## Tạo project

Việc xây dựng Cube với SSIS được thực hiện bằng công cụ SQL Server Business Intelligence Development Studio (BIDS).

* Bước 1: Khởi động BIDV 🡪 New Project 🡪 Integration Services Project 🡪 Đặt tên project



## Thực hiện

-Tạo 1 CSDL mới tên là TEST\_SSIS (CREATE DATABASE TEST\_SSIS)

Thông thường, 1 package của SSIS gồm Control Flow, Data Flow

* Control Flow có các components thực hiện 1 số công việc, trong đó quan trọng nhất là Data Flow Task
* Data Flow

Thực hiện với 1 số component trong nhóm Basic Transformation:

* **Conditional Split**
* **Sort**
* **Data Conversion**
* Aggregate
* Multi-cast
* Sampling
* **Union All**
* Merge Join

### Bài tập 1: Chuyển từ file excel 🡪 OLE DB

Nguồn: Excel (SINH\_VIEN.xlxs)

Đích: OLEDB (tblSV)

Chạy lại lần 2 thì nó append dữ liệu 🡪 muốn không append dữ liệu giới thiệu Execute SQL Task.

### Bài tập 2: Chuyển ngược từ OLEDB 🡪 Flat File

Nguồn: OLEDB (tblSV)

Đích: Flat File (Tạo trước file: SV\_01.txt)

### Bài tập 3: Chuyển Excel + Flat File 🡪 OLE DB

Nguồn: 2 nguồn Excel (SINH\_VIEN.xlxs) và (SINH\_VIEN.txt)

Đích: OLEDB (tblSV\_TONGHOP)

Trước đó phải sort lại dữ liệu (Sort, Union All)

### Bài tập 4: Chuyển dữ liệu có điều kiện

Nguồn: Flat File (SINH\_VIEN.txt)

Đích: OLEDB

Trước khi đưa dữ liệu vào OLEDB thì tách dữ liệu Nam thành 1 bảng (tblSV\_Nam), Nữ thành 1 bảng (tblSV\_Nu), có sắp xếp thứ tự. Nếu dữ liệu OLEDB đã có sẵn thì xóa trước khi thực hiện (Conditional Split)

### Bài tập 5: Có nhiều file text (lặp tự động)

Nguồn: 3 file excels (MONHOC\_01.xls; MONHOC\_02.xls; MONHOC\_03.xls)

Đích: OLEDB (tblMH)

Tạo bảng MH với script sau đây:

IF OBJECT\_ID('tblMH') IS NOT NULL

DROP TABLE tblMH

GO

CREATE TABLE tblMH

(

mamh VARCHAR(2) PRIMARY KEY,

tenmh VARCHAR(100) NOT NULL,

songay INT

)

Chuyển 3 file vào 1 OLEDB

### Bài tập 6: Lặp file text tự động

**Xem chi tiết trong project EXAMPLE01. Trong project này có nhiều packages tương ứng với các bài tập trên.**

**Lưu ý phải thay đổi các connection phù hợp với máy đang thực thi project.**