

Odoo 15

1. Installation

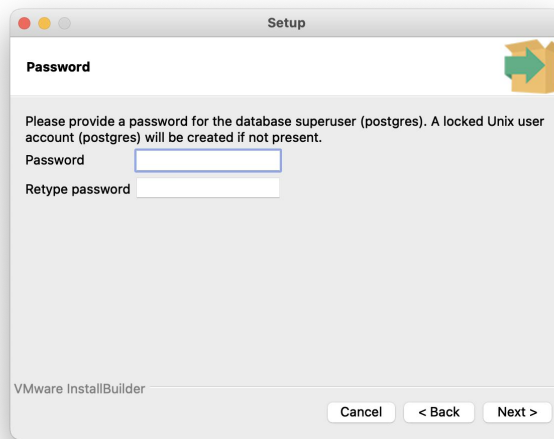
Database Installing

Download postgres gui tool and postgresql with EDB

<https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>

And install

Next > Next > Next > Next>



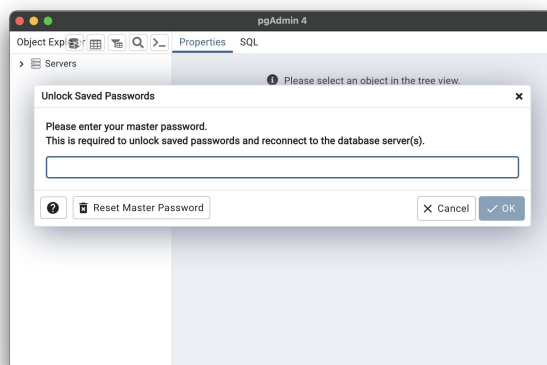
Password : admin

Retype password : admin

Next > Port 5432 Next > Next > Next >

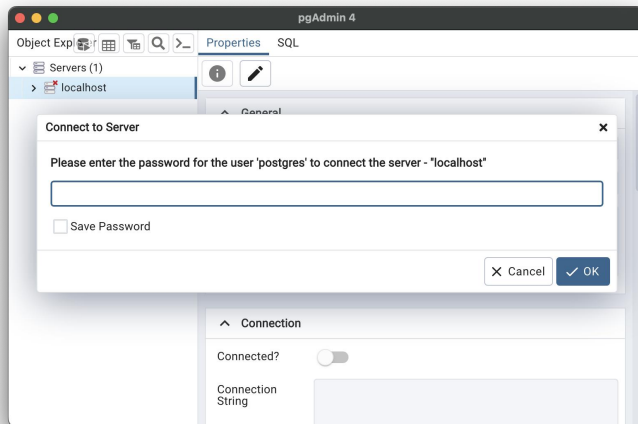
Create DB user account

1.Open pgadmin 4



Password : admin

2. Click Server and localhost



Password : admin

3. Right Click Login/Group Roles

4. And Create , Login/Group Roles

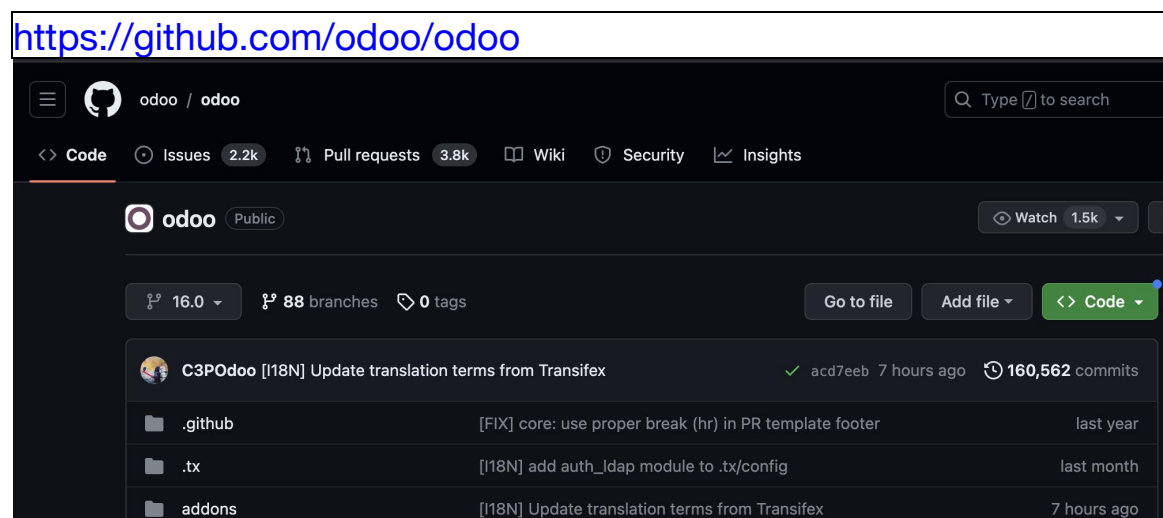
General > Name : odoo_user
Definition > Password : superuser
Privileges > All allow permission

5. And Save

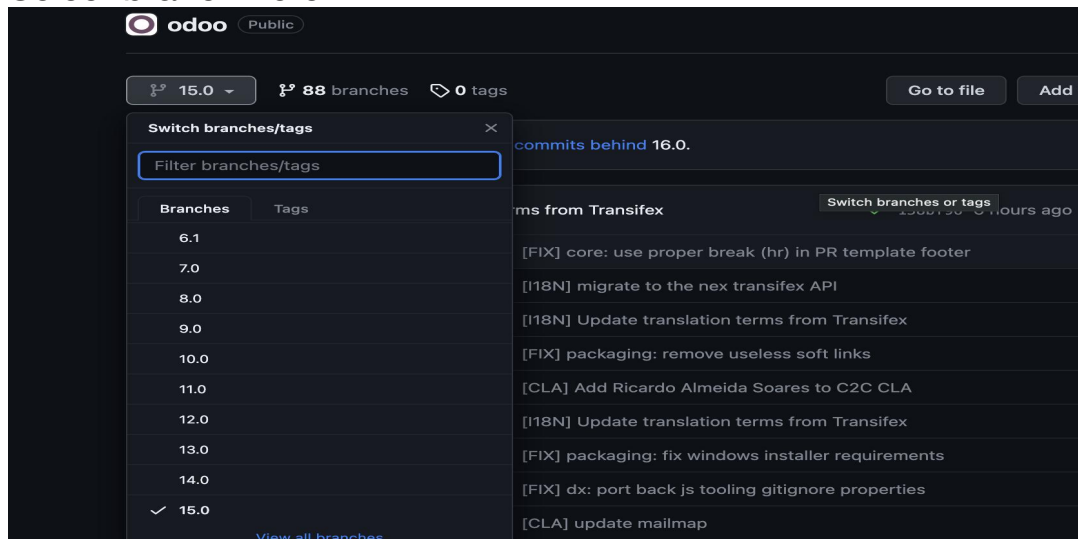
Odoo project installing

Download from github

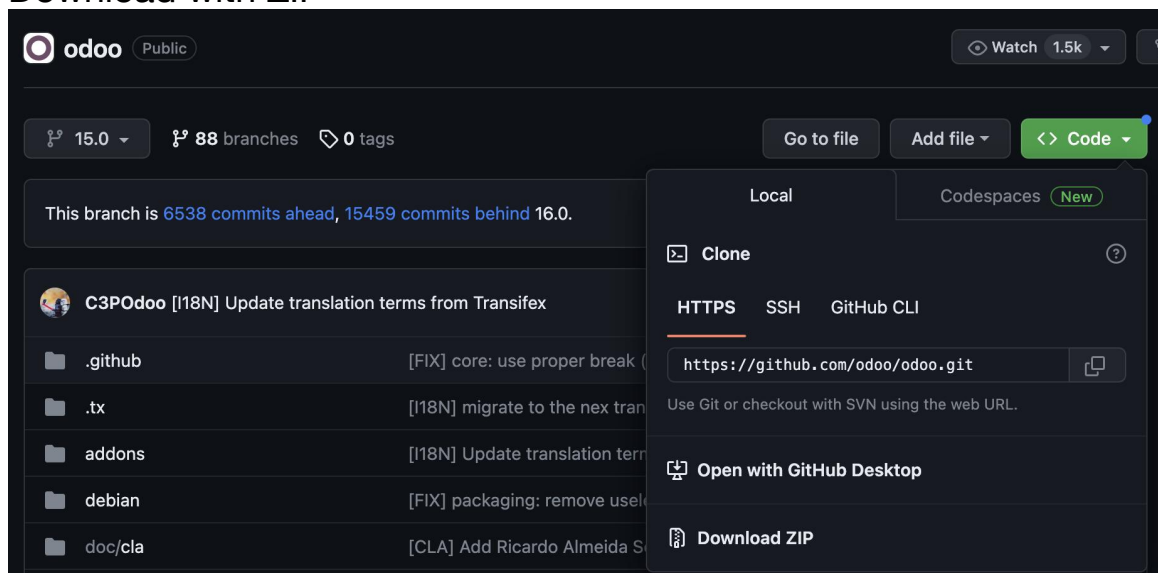
<https://github.com/odoo/odoo>



Select branch 15.0



Download with ZIP



Unzip Download ZIP File

Create a virtual environment

```
python -m venv odoo_venv
```

Activate virtual environment

```
./odoo_venv/Script/activate
```

Installation odoo requirements

```
cd odoo-15.0
```

```
python -m pip install -r requirements.txt
```

Setup Odoo Configuration

Create **odoo.conf** file in odoo-15.0

odoo.conf

[options]

admin_passwd = 12345

db_user = odoo_user

db_password = superuser

dbfilter = ^odoo_15db.*\$

addons_path = /Users/Desktop/odoo_dev/odoo-15.0/addons

http_port = 8069

* Change **addons_path**

right click **addons** folder and copy path in vs code
and paste

Run Odoo Project in CMD

```
python odoo-bin --conf odoo.conf
```

(or)

```
python odoo-bin -c odoo.conf
```

Search **localhost:8069** in your browser

odoo

Master Password

Database Name

Email

Password

Phone number

Language

Country

Demo data ☐

[Create database](#) or [restore a database](#)

Create Database

Master Password - 12345

Database Name - odoo_15

Email - admin@gmail.com

Password - superuser

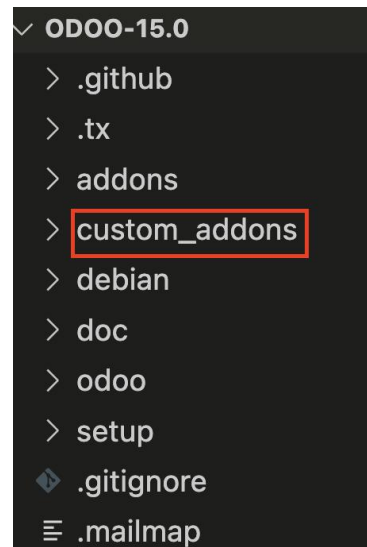
Click Create database

Testing

- Logout and login
- Create new database
 - logout -> Manage Databases -> Create Database ->
- Create new user and password
 - Create User
 - Settings -> User & Companies -> User -> Create
 - User Permission
 - > Access Rights -> Administration -> Settings
 - User Password
 - Click Save -> Action -> Change Password -> New Password
- Customize company
 - Settings -> Users & Companies -> Companies
- Discuss
- Point of Sale
- eCommerce
- Website
- Attendances
-

Create Custom addons

- . Create **custom_addons** folder in odoo-15.0



- . Add custom_addons path to odoo.conf addons_path

- . Right click **custom_addons** folder and copy path in vs code and paste

odoo.conf

```
....  
addons_path = /Users/Desktop/odoo_dev/odoo-15.0/addons,  
/Users/Desktop/odoo_dev/odoo-15.0/custom_addons  
....  
....
```

Create Module with scaffold

```
python odoo-bin scaffold module_one custom_addons
```

Odoo Developer mode

Change Url

`http://localhost:8069/web#cids=1&menu_id=5&actio.....`

to

`http://localhost:8069/web?debug=1#cids=1&menu_id=5&actio.....`

Update Module

Click **Update Apps List** button in header

Search **module_one** in search box

Add Icon and Description

Create folder and add photo and design html

- module_one
 - static
 - description
 - icon.png
 - index.html

Download Odoo theme Module and Install

<https://apps.odoo.com/apps/themes?price=Free>

Create Module With Normal

Create **todo_app** folder in **custom_addons**

Create **__manifest__.py** file in **todo_app**

```
__manifest__.py
{
    "name": "My To Do App",
    "description": """
        this is my todo app
        this is my todo app
        this is my todo app
    """,
    "author": "Your Name",
    "website": "www.pagemyanmar.com",
    "category": "todo",
    "summary": "my todo"
}
```

Click **Update Apps List** button in header

Search **todo_app** in search box

Model

Create **models** folder in **todo_app**

Create **__init__.py** file in **todo_app**

Create **__init__.py** file in **models**

Create **todo_model.py** file in **models**

todo_model.py

```
from odoo import models, fields

class TodoTask(models.Model):

    _name = 'todo.task'
    _description = 'this is todo table'

    name = fields.Char('Description', required=True)
    is_done = fields.Boolean('Done?')
    active = fields.Boolean('Active?', default=True)
```

__init__.py/todo_app

```
from . import models
```

__init__.py/models

```
from . import todo_model
```

Create **__init__.py** file in **todo_app**

Stop and Restart the server
And Upgrade todo_app

See **todo.task** table from **Settings/Technical/Models** And
pgadmin

View

Create **views** folder in **todo_app**

Create **todo_view.xml** file in **views**

todo_view.xml

```
<odoo>
  <data>
    <!-- kanban view -->

    <!-- tree view -->

    <!-- Form View -->

    <!-- search view -->

    <!-- Action to open To-do Task list -->

    <!-- Menu item to open To-do Task list -->

  </data>
</odoo>
```

Add Action and Menu

todo_view.xml

```
....
    <!-- Action to open To-do Task list -->
    <record id="action_todo_task"
model="ir.actions.act_window">
      <field name="name">To-do Task</field>
      <field name="res_model">todo.task</field>
      <field name="view_mode">kanban,tree,form</field>
      <field name="help" type="html">
        <p class="o_view_nocontent_smiling_face">
          Nothing to Show !
        </p>
      </field>
    </record>

    <!-- Menu item to open To-do Task list -->
    <menuitem id="menu_todo_task"
      name="To Do"
```

```
parent="mail.menu_root_discuss"
sequence="20"
action="action_todo_task"
/>
...
```

Security

Create **security** folder in todo_app

Create **ir.model.access.csv** in security

ir.model.access.csv

```
id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
access_todo_task_group_user,todo.task.user,model_todo_task,base.group_user,1,1,1,1
```

Add to __manifest__.py

```
__manifest__.py
....
"summary": "my todo"
'data': [
    'views/todo_view.xml',
    'security/ir.model.access.csv'
]
}
```

Stop and Restart the server
And Upgrade todo_app

Add Function

todo_model.py

```
.....
name = fields.Char('Description', required=True)
is_done = fields.Boolean('Done?')
active = fields.Boolean('Active?', default=True)

def do_toggle_done(self):
    self.is_done = not self.is_done
    return True

def do_clear_done(self):
    done_recs = self.search([('is_done', '=', True)])
    done_recs.write({'active': False})
    return True
```

Add Form View

todo_view.xml

```
....
<!-- Form View -->
<record id="view_form_todo_task" model="ir.ui.view">
    <field name="name">To-do Task Form</field>
    <field name="model">todo.task</field>
    <field name="arch" type="xml">
        <form>
            <header>
                <button name="do_toggle_done" type="object"
                    string="Toggle Done" class="oe_highlight" />
                <button name="do_clear_done" type="object"
                    string="Clear All Done" />
            </header>
            <sheet>
                <group name="group_top">
                    <group name="group_left">
                        <field name="name" />
                    </group>
                    <group name="group_right">
                        <field name="is_done" />
                        <field name="active" readonly="1" />
                    </group>
                </group>
            </sheet>
        </form>
    </field>
</record>
```

```

        </sheet>
      </form>
    </field>
  </record>
...

```

Add Tree View

```

<!-- tree view -->
<record id="view_tree_todo_task" model="ir.ui.view">
  <field name="name">To-Do Task Tree</field>
  <field name="model">todo.task</field>
  <field name="arch" type="html">
    <tree>
      <field name="name"/>
      <field name="is_done"/>
      <field name="active"/>
    </tree>
  </field>
</record>

```

Add Kanban View

```

<!-- kanban view -->
<record id="view_kanban_todo_task" model="ir.ui.view">
  <field name="name">To-do task Kanban</field>
  <field name="model">todo.task</field>
  <field name="arch" type="xml">
    <kanban class="o_kanban_mobile">
      <field name="id" />
      <field name="name" />
      <field name="active" />
      <field name="is_done" />
      <templates>
        <t t-name="kanban-box">
          <div t-attf-class="oe_kanban_card
oe_kanban_global_click">
            <div class="o_kanban_content">
              <div class="oe_kanban_details">

```

```

<strong>
<field name="name" />
</strong>
<br/>
<field name="active" />
</div>
<div
class="o_kanban_record_bottom">
<div
class="oe_kanban_bottom_left">
<field name="is_done" />
</div>
</div>
</div>
</t>
</templates>
</kanban>
</field>
</record>

```

I18n

Report

Demo

Data

Controllers