

onnet

2021  
VIE

# Odoo theme

[www.on.net.vn](http://www.on.net.vn)  
[www.arrowhitech.com](http://www.arrowhitech.com)



# Table of contents

- **Structure of odoo page**
- **Odoo theme structure**
- **Thinking Modular**
- **Create theme module**
- **Install theme module**



# Structure of odoo page

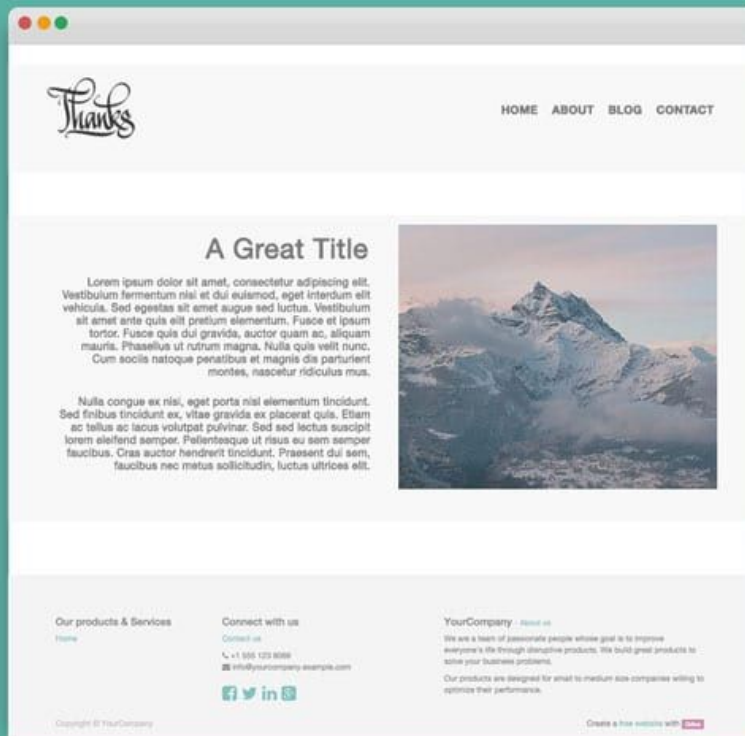
---

An Odoo page is the visual result of a combination of **2 kind of elements**:

- **cross-pages**
- and **unique**

By default, Odoo provides you with a **Header** and a **Footer** (cross-pages) and a unique **main element** that contains the content that makes your page unique.

# Structure of odoo page



Header  
(cross-pages)

Main  
(unique for this page)

Footer  
(cross-pages)

# Odoo theme structure

When using common **CMSs**, you have to code several files (like header.php, page.php, post.php, etc.) in order to create a basic structure for your website. With those systems, this base structure acts as a design foundation that you have to update over time to ensure compatibility within your CMS. So, even after you have spent hours coding the files, you have not even started on the design yet.



- Your website structure
- Your website styles and content

# Odoo theme structure

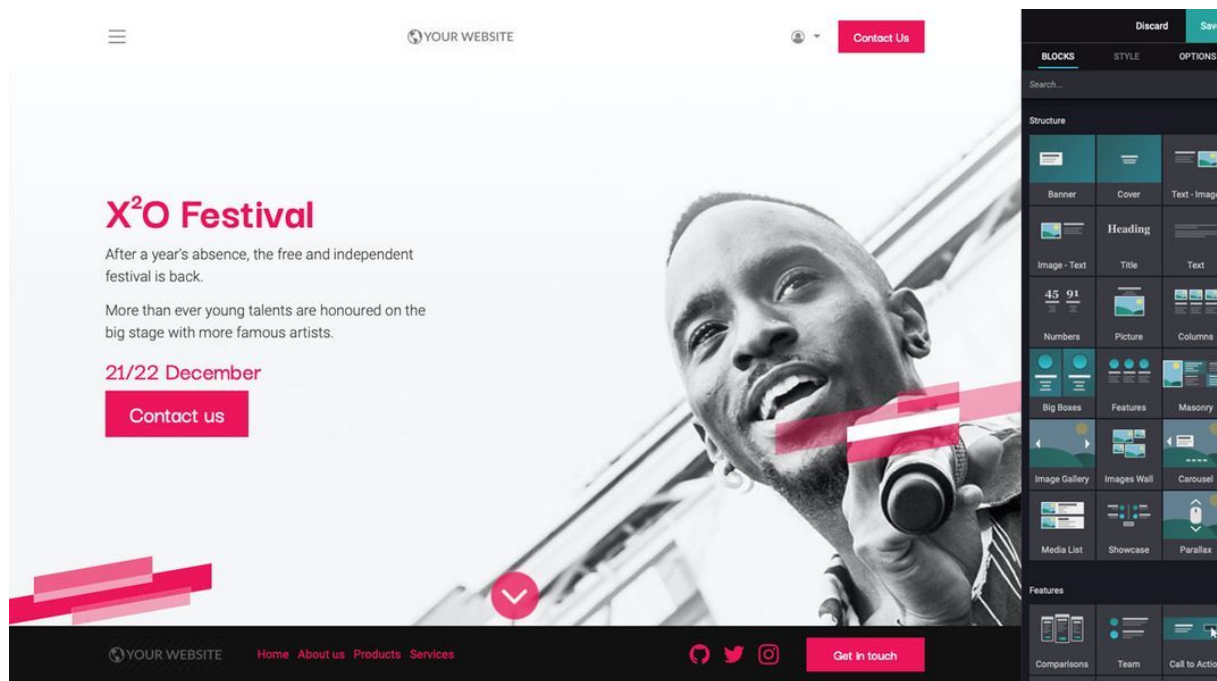
Odoo comes with a default website structure. It is a very basic “**theme**” that provides **minimal structure** and **layout**. When you create a new theme, you are actually **extending** this. Indeed it’s always enabled in your setup, except that you don’t have to create or maintain it. It will upgrade automatically within your Odoo installation and, since it is included in the **Website Builder module**, everything is smoothly integrated by default.

As a result, you are totally free to focus on design while this structure does the job of providing integrations and functionality.



- Odoo default theme structure
- Your website styles and content

# Odoo theme structure: Website builder



# Odoo theme structure

---

## Main features:

- Basic layouts for pages, blog, eCommerce and much more
- Website Builder integration
- Basic Snippets
- Automatic Sass compiling
- Automatic Js and CSS minification and combination

## Main technologies:

- Twitter Bootstrap
- jQuery
- jQuery UI
- underscore.js



# Thinking Modular

---

An **Odoo theme** is not a folder containing **HTML** or **PHP files**, it's a modular framework written in XML.

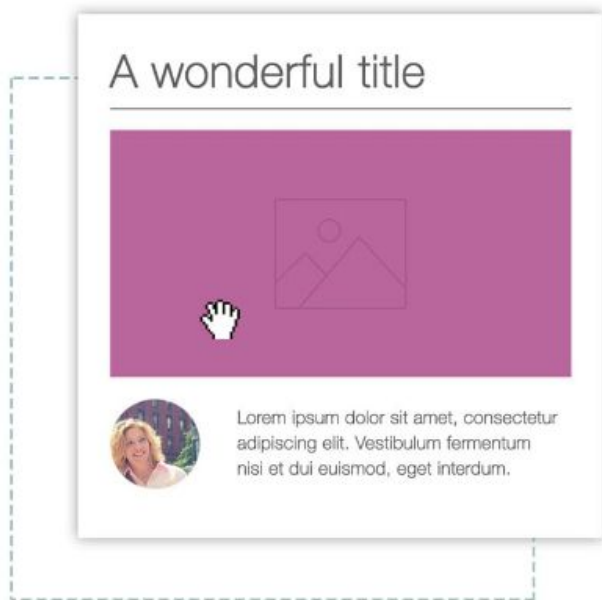
Using classical web design workflows, you usually code the layout of the entire page. The result of this is a “**static**” web page. You can update the content, of course, but your client will need you to work on making even basic changes.

Creating themes for Odoo is a total change of perspective. Instead of defining the complete layout for a page, you can create blocks (**snippets**) and let the user choose where to “**drag & drop**” them, creating the page layout on their own. We call this modular design.

Imagine an Odoo theme as a “**list**” of elements and options that you have to create or extend. As a designer, your goal is to style these elements in order to achieve a wonderful result, regardless of where the end user chooses to place them.

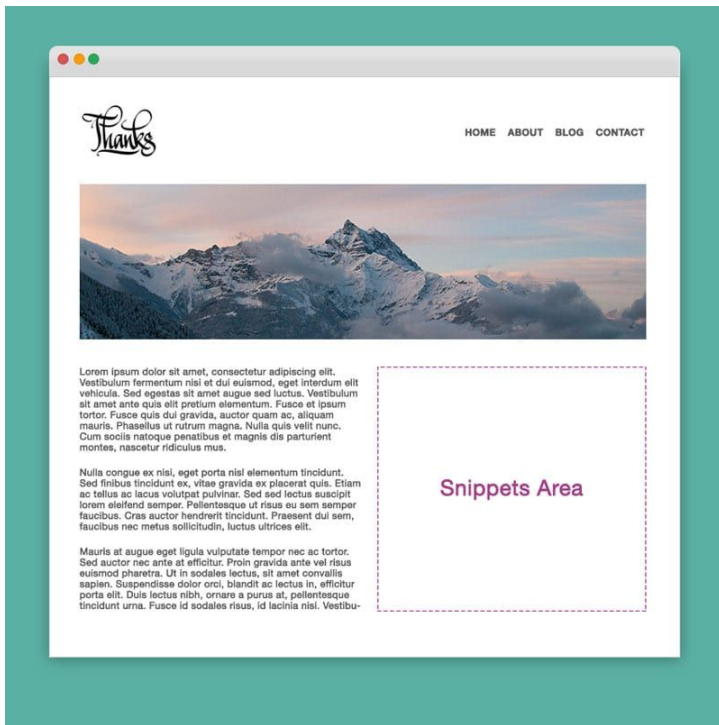
# Thinking Modular: Snippets

Snippet is **a piece of HTML code**. The user will **drag & drop**, modify and combine them using our built-in **Website Builder** interface. You can define sets of options and styles for each snippet. The user will choose from them according to their needs.



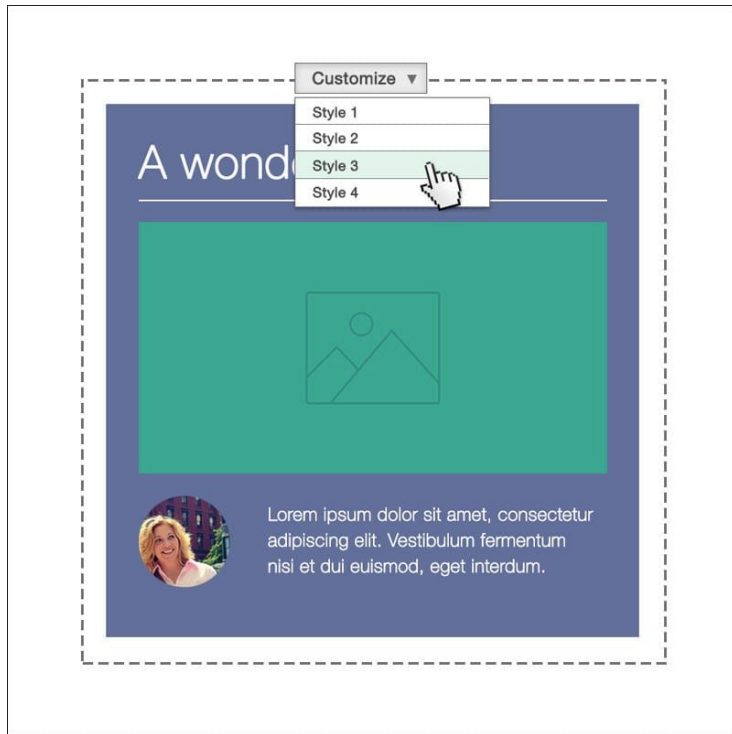
# Thinking Modular: Pages

These are normal web pages, except that they will be **editable** by the final user and that you can define an empty area that the user can **“fill”** by **dragging snippets** into it.



# Thinking Modular: Styles

**Styles** are defined using **SCSS files** (or you can use standard CSS files). You can define a style as default or optional. The default styles are always active in your theme, the optional styles can be enabled or disabled by the user.



# Thinking Modular: Functionalities

Thanks to **Odoo's modularity**, everything can be personalized even more. This means there are endless possibilities for your creativity. Adding functionalities is easy and it's simple to provide the end user with customizable options.



# Thinking Modular: XML files

Any Odoo XML file starts with encoding specifications. After that, you have to write your code inside a `<odoo>` tag.

## Example:

```
<?xml version="1.0" encoding="utf-8" ?>
```

```
<odoo>
```

```
    ## YOUR CODE HERE
```

```
</odoo>
```

# Thinking Modular: XML files

Almost every **element** and **option** that you create has to be placed inside a **<template>** tag, like in this example:

```
<template id="my_title" name="My title">

    <h1>This is an HTML block</h1>

    <h2 class="lead">And this is a subtitle</h2>

</template>
```

The above code defines a title, but it will not be displayed anywhere because that *template* is not associated with any part of the **Odoo default structure**. In order to do that you can use **xpath**, **qWeb** or a combination of both.

# Create a theme module

**Odoo's themes** are packaged like **modules**. Even if you are designing a very simple website for your company or client, you need to package the theme like an Odoo module.

## main folder

Create a folder and name it like this: **theme\_** followed by your theme's name.

## `__manifest__.py`

Create an empty document and save it to your folder as `__manifest__.py`. This will contain the configuration info for your theme.

## `__init__.py`

Create another empty file and name it `__init__.py`. It's a mandatory system file. Create and leave it blank.

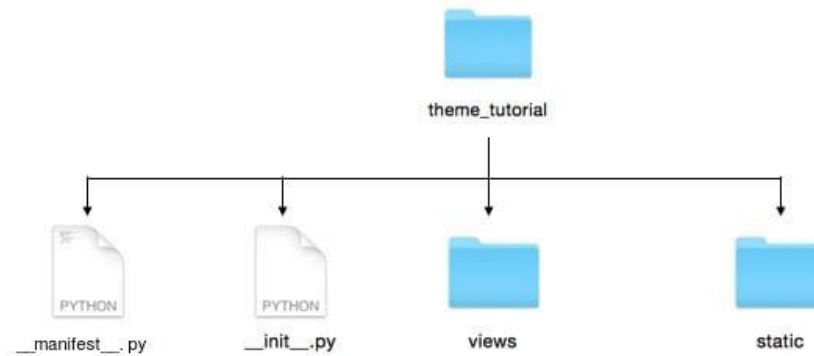
## views and static folders

Create them in the main folder. In **views** you'll place your **xml files** that define your **snippets**, your **pages** and your **options**. **static** folder is the right place for your style , images and custom js code.



# Create a theme module

---



# Create a theme module: `__manifest__.py`

Open the `__manifest__.py` you created and copy/paste the following:

```
{
    'name': 'Tutorial theme',
    'description': 'A description for your theme.' ,
    'version': '1.0',
    'author': 'Your name',
    'category': 'Theme/Creative',
    'depends': ['website', 'website_theme_install'],
    'data': [
    ],
}
```

# Create a theme module: `__manifest__.py`



- **category:** defines your module category (always “Theme”) and, after a slash, the subcategory. You can use one subcategory from the Odoo Apps categories list. (<https://www.odoo.com/apps/themes>)
- **depends:** specifies the modules needed by our theme to work properly.
- **data:** property will contain the xml files list. Right now it’s empty, but we will add any new files created.

# Install theme

---

To install your theme, you just place your theme folder inside addons in your Odoo installation.

After that, navigate to the Odoo **Website** module, go to **Configuration** › **Settings**.

Under **Website** section click the **Pick a Theme** button, then hover over your theme and click **Use this theme**.

# Install theme

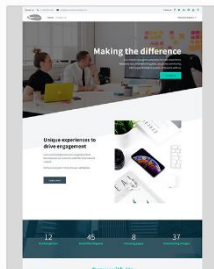
Settings / Pick a Theme



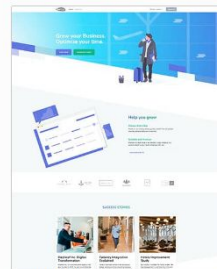
Gromed



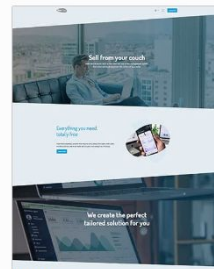
DEVELOPMENT Cobalt



SERVICE Graphene



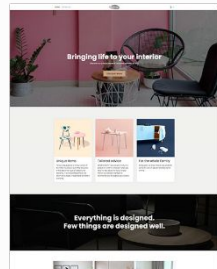
CONSULTANCY Paptic



LEGAL Clean



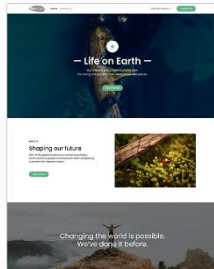
TRAVEL Aviato



FURNITURE Loftspace



CORPORATE Buzzy



ENVIRONMENT Treehouse



DESIGN Avantgarde



DIGITAL Zap



BEAUTY Beauty



# Q&A