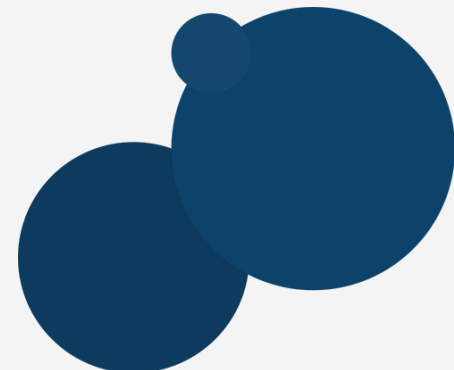


# Introduction to Django-CMS



- **Computer Science Master Student @ TU Berlin**
- **Python / Django since 2010**
- **First “coding” back in 1999**



# What is Django-CMS?



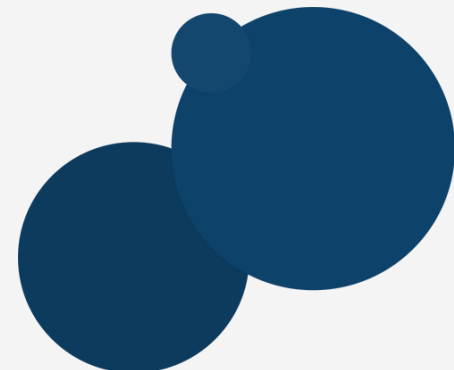
- **Uses the Django Web framework**
  - Python
  - Database independent (PostgreSQL, MySQL, ...)
- **Open Source (BSD License)**
  - Backed by a large Swiss company (Divio AG)
- **Widely used**
  - 1010 Forks, 2627 Watchers, 8283 Commits



# What is Django-CMS?



- **Integrates seamlessly in entire Internet presence**
- **Highly customizable**
  - Plugins
  - Apps
  - Menus
- **Templates / HTML**
- **CSS**



# Setting up a demo application

django The logo for django CMS, featuring the word 'django' in a lowercase sans-serif font followed by 'CMS' in a bold, uppercase sans-serif font inside a white rounded rectangle.

# Create a new virtualenv with a project directory

1. `$ mkproject -p /usr/bin/python2.7.djangocms-talk`

# Install Django, Django-CMS and a few other requirements

2. `$ pip install "Django==1.6.2" South psycopg2 Pillow`

3. `$ pip install -e \`  
`> "git+https://github.com/divio/django-cms.git@fdbccf1086#egg=django-cms"`

# Install a few Django-CMS plugins

4. `$ pip install.djangocms-text-ckeditor.djangocms-admin-style \`  
`>.djangocms-link`

# Create a new Django project

5. `$ django-admin.py startproject.djangocms_talk .`



# Database setup

```
# We will use PostgreSQL
```

```
# Create new database user
```

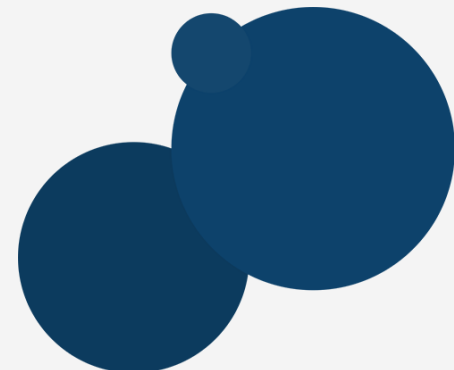
```
1. $ sudo -u postgres createuser -P -S -D -R.djangocms_talk
```

```
Enter password for new role:
```

```
Enter it again:
```

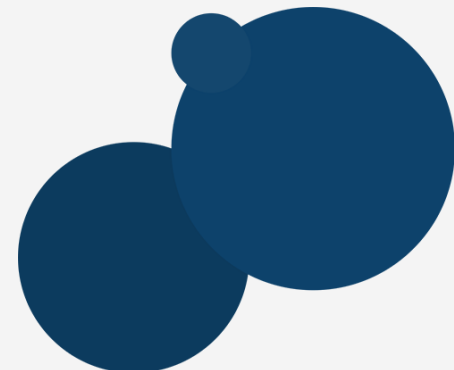
```
# Create new database
```

```
2. $ sudo -u postgres createdb .djangocms_talk
```



# Database configuration

```
# Configure Django's database settings
$ vim.djangocms_talk/settings.py
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql_psycopg2',
        'NAME': 'djangocms_talk',
        'USER': 'djangocms_talk',
        'PASSWORD': 'djangocms_talk',
    }
}
INSTALLED_APPS = (
    ...
    'south',
)
```





# Create database tables

```
$ python manage.py syncdb
Syncing...
Creating tables ...
...
```

You just installed Django's auth system, which means you don't have any superusers defined.

Would you like to create one now? (yes/no): yes

Username (leave blank to use 'markus'):

Email address:

Password:

Password (again):

Superuser created successfully.

Installing custom SQL ...

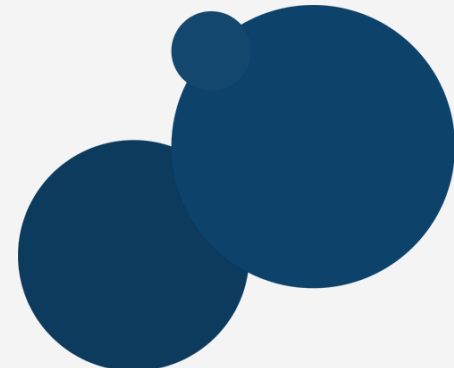
Installing indexes ...

Installed 0 object(s) from 0 fixture(s)

Synced:

> ...

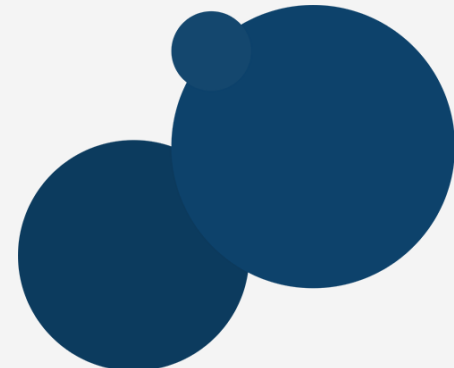
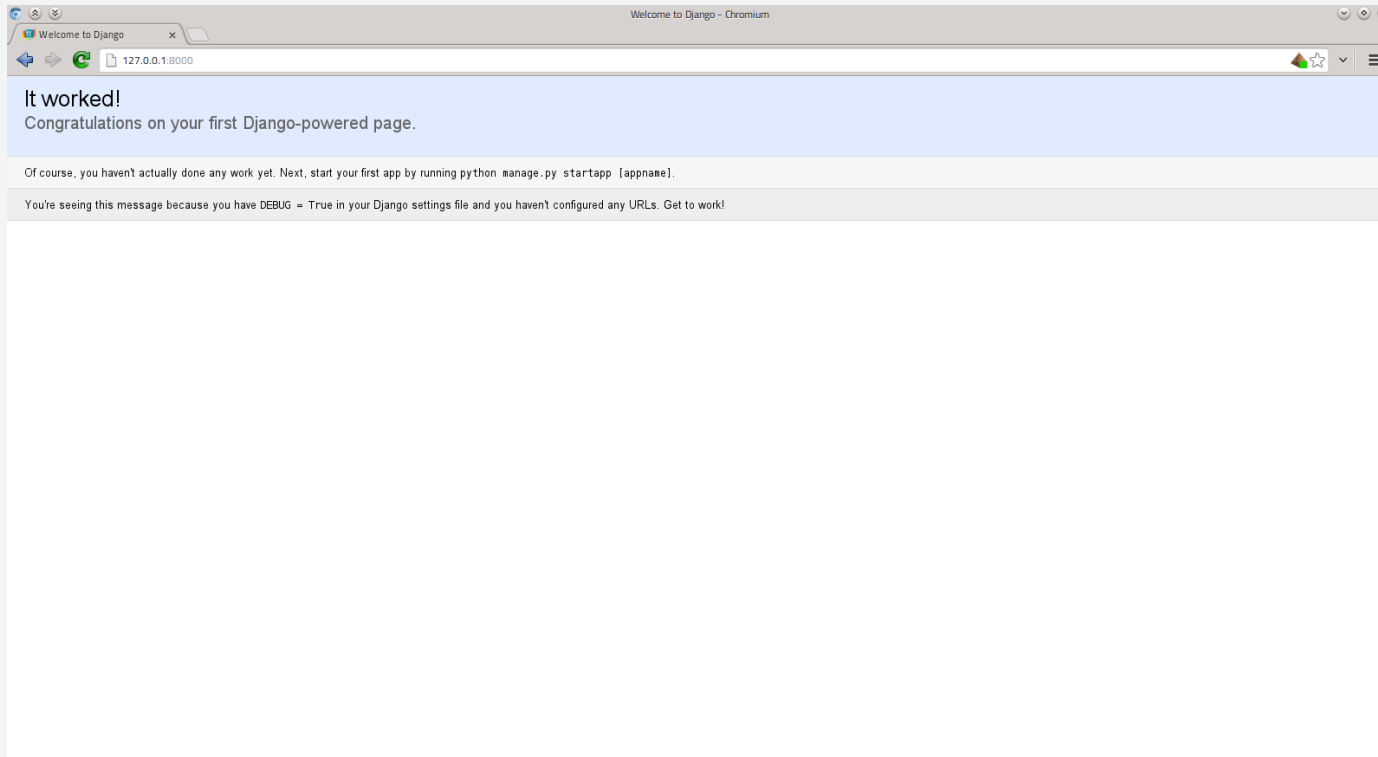
...



# Ready (1)

django **CMS**

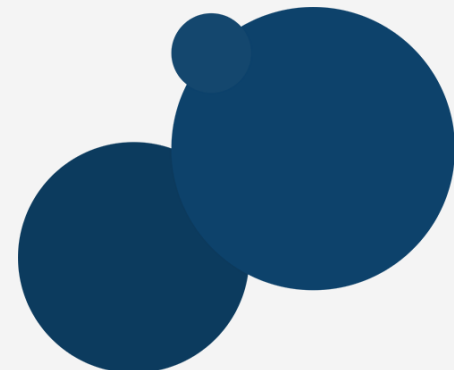
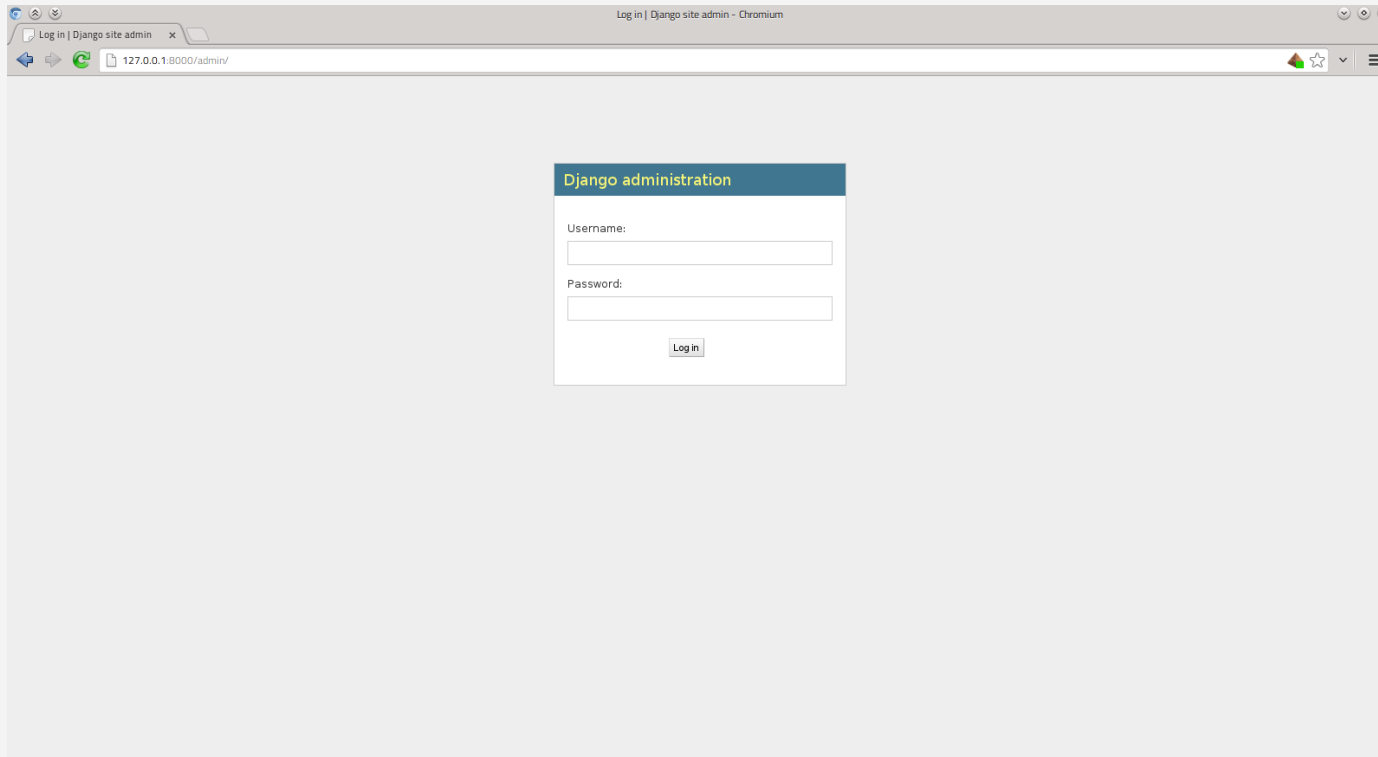
```
# Start Django's development server  
$ python manage.py runserver
```



# Ready (1)

django **CMS**

```
# Start Django's development server  
$ python manage.py runserver
```



# Django-CMS configuration (1)



```
$ vim.djangocms_talk/settings.py
```

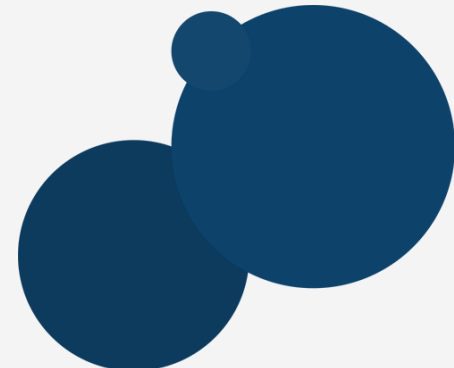
```
# ADD: Absolute path to our project directory
```

```
PROJECT_DIR = os.path.join(BASE_DIR, 'djangocms_talk')
```

```
# Django-CMS has multi-site support.
```

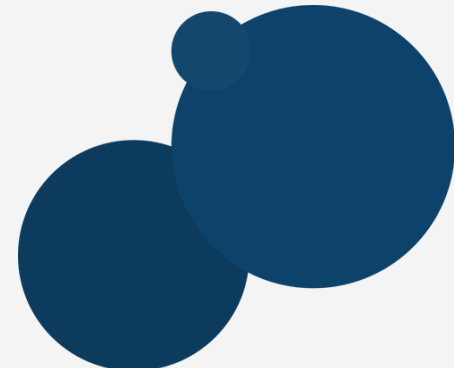
```
# ADD: Site-ID for this site
```

```
SITE_ID = 1
```



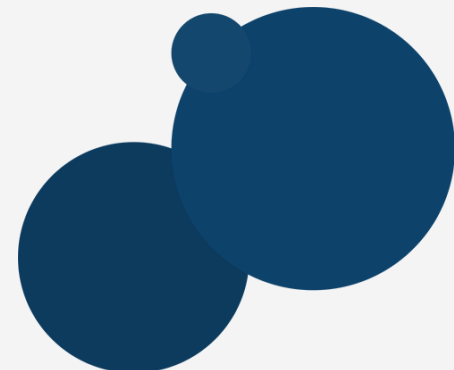
# Django-CMS configuration (2)

```
INSTALLED_APPS = (  
    'djangocms_admin_style', # must be before 'django.contrib.admin'!  
    'django.contrib.admin', # already there  
    'django.contrib.auth', # already there  
    'django.contrib.contenttypes', # already there  
    'django.contrib.sessions', # already there  
    'django.contrib.sites',  
    'django.contrib.messages', # already there  
    'django.contrib.staticfiles', # already there  
    'south', # already there  
    'djangocms_text_ckeditor', # must be before 'cms'!  
    'djangocms_link',  
    'cms',  
    'mptt',  
    'menus',  
    'sekizai',  
)
```



# Django-CMS configuration (3)

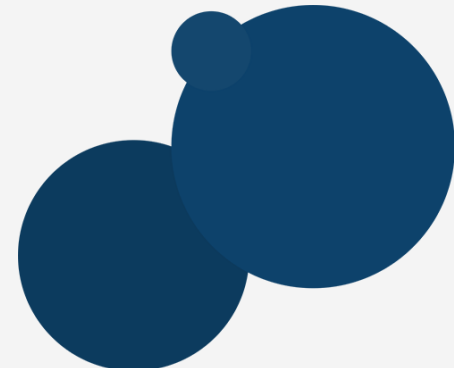
```
MIDDLEWARE_CLASSES = (  
    'django.contrib.sessions.middleware.SessionMiddleware', # already there  
    'django.middleware.common.CommonMiddleware', # already there  
    'django.middleware.csrf.CsrfViewMiddleware', # already there  
    'django.contrib.auth.middleware.AuthenticationMiddleware', # already there  
    'django.middleware.locale.LocaleMiddleware',  
    'django.contrib.messages.middleware.MessageMiddleware', # already there  
    'django.middleware.clickjacking.XFrameOptionsMiddleware', # already there  
    'cms.middleware.page.CurrentPageMiddleware',  
    'cms.middleware.user.CurrentUserMiddleware',  
    'cms.middleware.toolbar.ToolbarMiddleware',  
    'cms.middleware.language.LanguageCookieMiddleware',  
)
```



# Django-CMS configuration (4)

# ADD: not part of the project template in Django 1.6, just add them

```
TEMPLATE_CONTEXT_PROCESSORS = (  
    'django.contrib.auth.context_processors.auth',  
    'django.core.context_processors.debug',  
    'django.core.context_processors.i18n',  
    'django.core.context_processors.request',  
    'django.core.context_processors.media',  
    'django.core.context_processors.static',  
    'django.core.context_processors.tz',  
    'django.contrib.messages.context_processors.messages',  
    'cms.context_processors.cms_settings',  
    'sekizai.context_processors.sekizai',  
)
```



# Django-CMS configuration (5)

# ADD: where to look for static file assets (CSS, JS, Images)

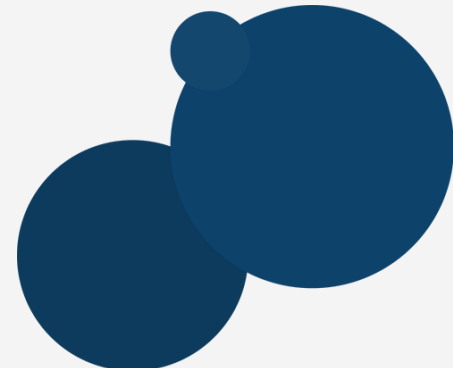
```
STATICFILES_DIRS = (  
    os.path.join(PROJECT_DIR, 'static'),  
)
```

# ADD: where to put user uploaded content

```
MEDIA_ROOT = os.path.join(PROJECT_DIR, 'media')
```

# ADD: which URL to use for user uploaded content

```
MEDIA_ = '/media/'
```





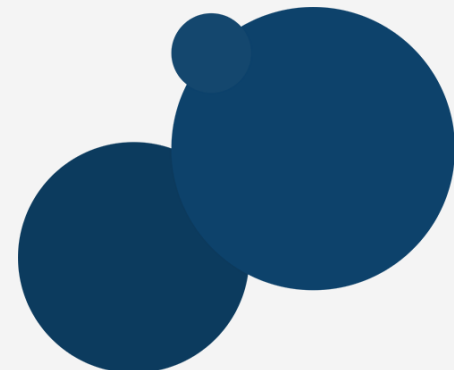
# Django-CMS configuration (6)

# ADD: where to look for additional template files

```
TEMPLATE_DIRS = (  
    os.path.join(PROJECT_DIR, 'templates'),  
)
```

# ADD: which HTML page templates are available for new Django-CMS pages

```
CMS_TEMPLATES = (  
    ('start.html', 'Start page'),  
    ('content.html', 'Content page'),  
)
```



# Django-CMS configuration (7)

# ADD: which languages will be available within Django-CMS for page translation

```
LANGUAGES = (  
    ('en', 'English'),  
)
```

# CHANGE: default language to use

```
LANGUAGE_CODE = 'en'
```

# CHANGE: default timezone to use

```
TIME_ZONE = 'Europe/Berlin'
```



# Syncing the database (again)

django **CMS**

```
$ python manage.py syncdb
```

```
Syncing...
```

```
...
```

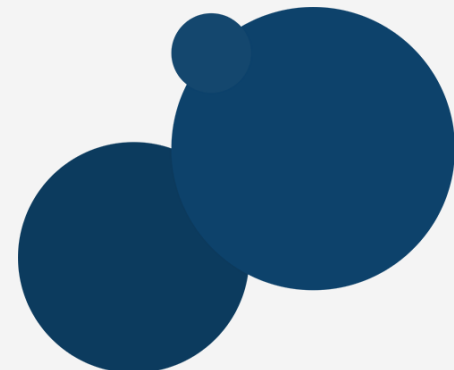
```
Synced:
```

```
> ...
```

```
Not synced (use migrations):
```

- .djangocms\_text\_ckeditor
- djangocms\_link
- cms
- menus

```
(use ./manage.py migrate to migrate these)
```



# Migrating the database changes

```
$ python manage.py migrate
```

```
Running migrations for.djangocms_textckeditor:
```

```
- Migrating forwards to 0002_rename_plugin.
```

```
> cms:0001_initial
```

```
...
```

```
- Loading initial data for cms.
```

```
Installed 0 object(s) from 0 fixture(s)
```

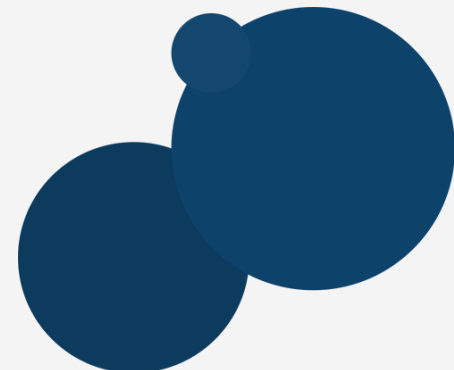
```
Running migrations for menus:
```

```
- Migrating forwards to 0001_initial.
```

```
> menus:0001_initial
```

```
- Loading initial data for menus.
```

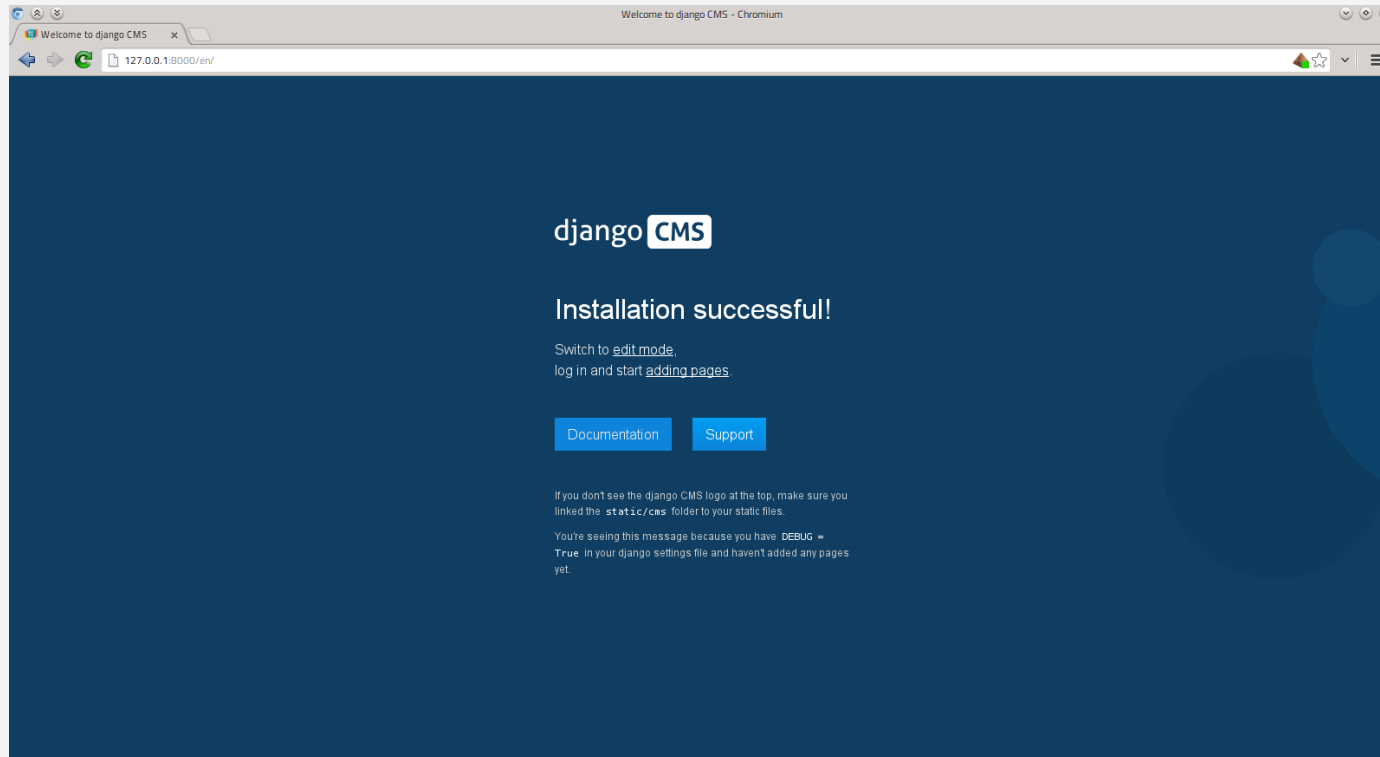
```
Installed 0 object(s) from 0 fixture(s)
```



# Ready (2)

django **CMS**

```
# Start Django's development server  
$ python manage.py runserver
```



# Creating CMS HTML templates

# Create some directories we need later on

1. `$ mkdir -p.djangocms_talk/{static,media,templates}`

# Create HTML template files

2. `$ touch.djangocms_talk/templates/{base,start,content}.html`

```
{% extends "base.html" %}
{% load cms_tags %}
{% block content %}
<div class="jumbotron">
  <div class="container">
    {% placeholder jumbotron %}
  </div>
</div>
<div class="container">
  <div class="row">
    <div class="col-lg-4">
      {% placeholder col1 %}
    </div>
    <div class="col-lg-4">
      {% placeholder col2 %}
    </div>
    <div class="col-lg-4">
      {% placeholder col3 %}
    </div>
  </div>
</div>
{% endblock %}
```

# Check Django-CMS installation

```
$ python manage.py cms check
*****
Checking django CMS installation
*****
```

```
...
```

```
OVERALL RESULTS
```

```
=====
```

```
1 checks skipped!
9 checks successful!
```

```
Installation okay
```



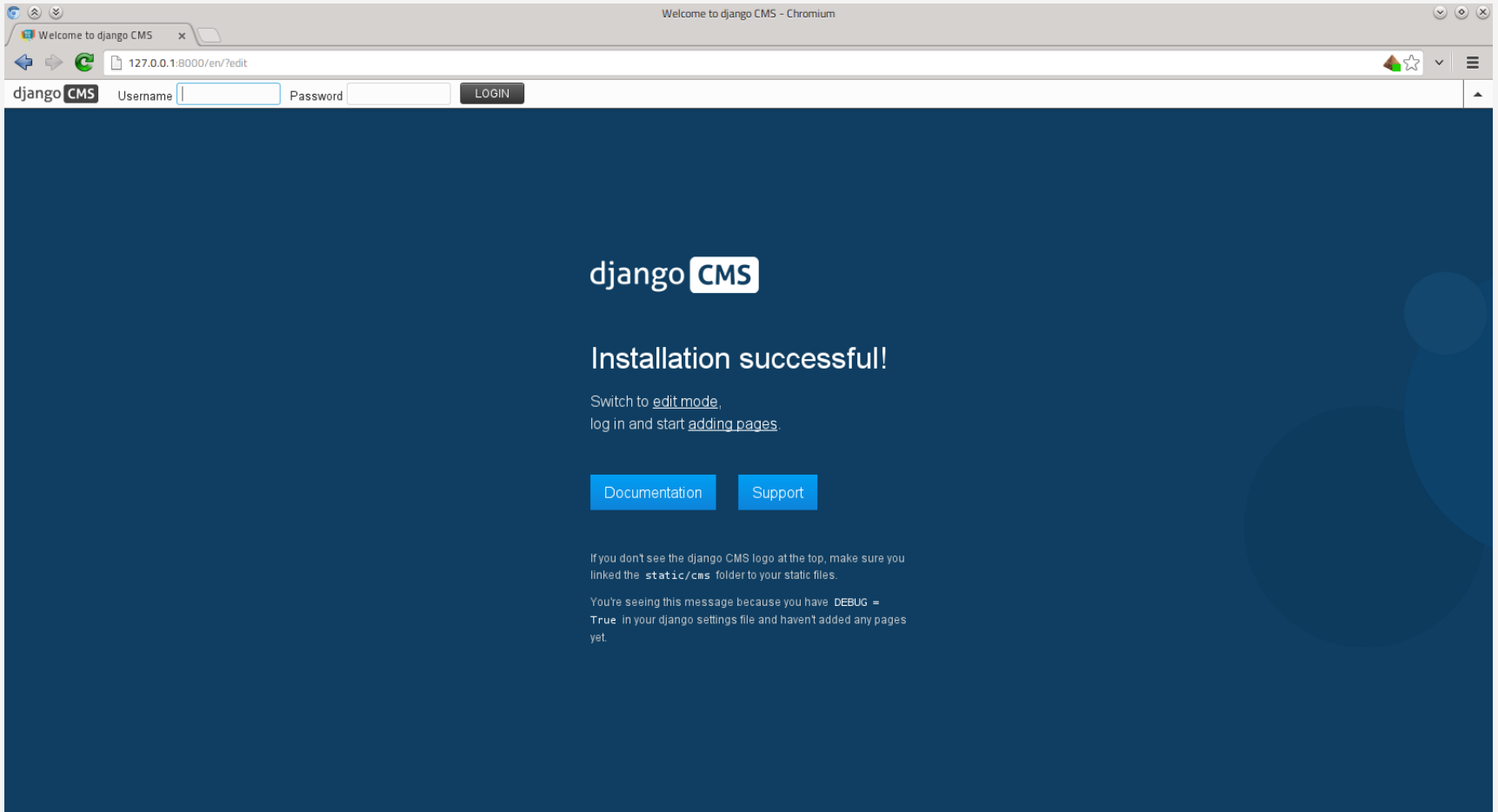
# First steps in our new demo application

django **CMS**



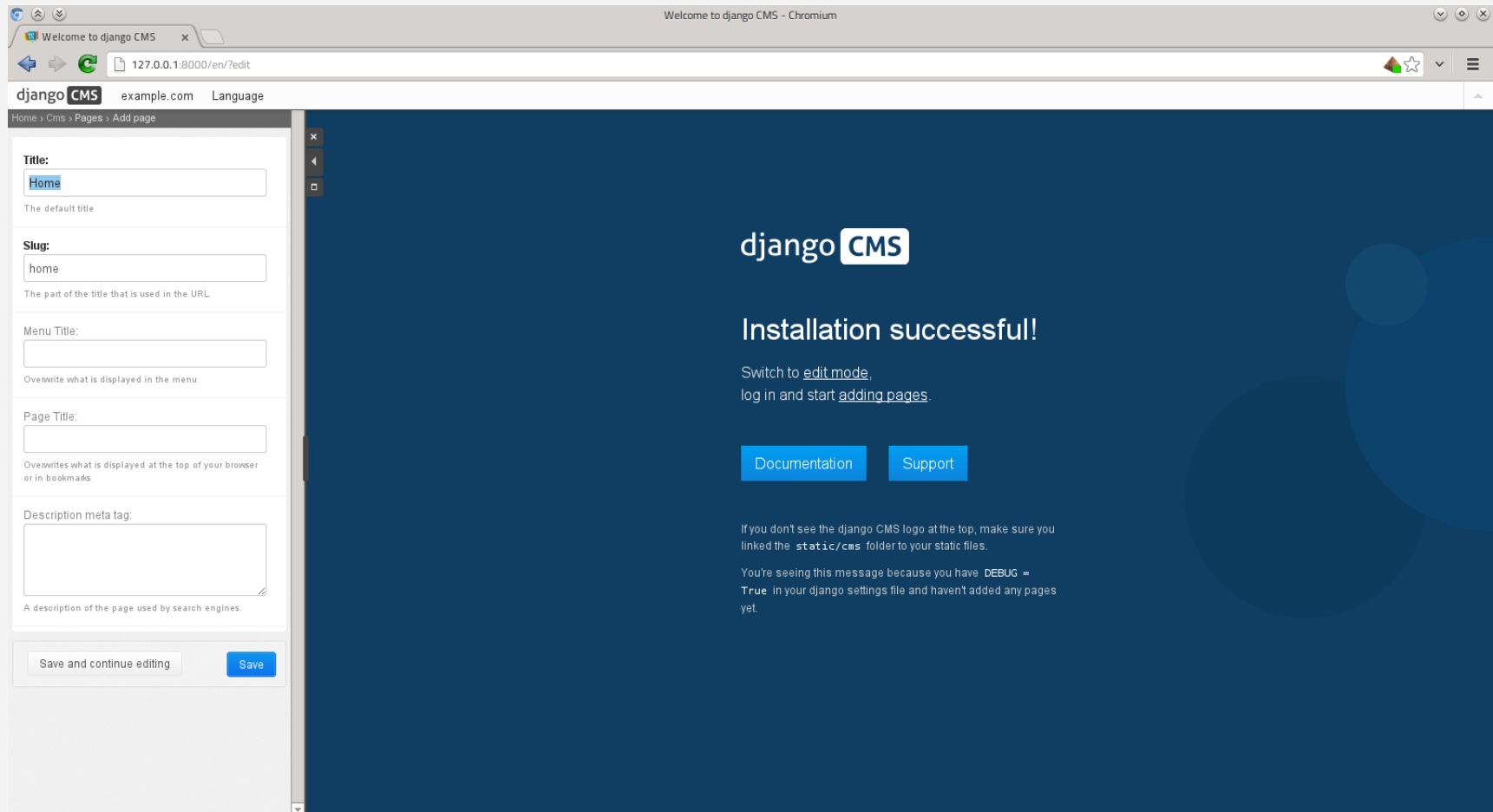
# Django-CMS First Steps

## Log-in



# Django-CMS First Steps

## Adding a start page



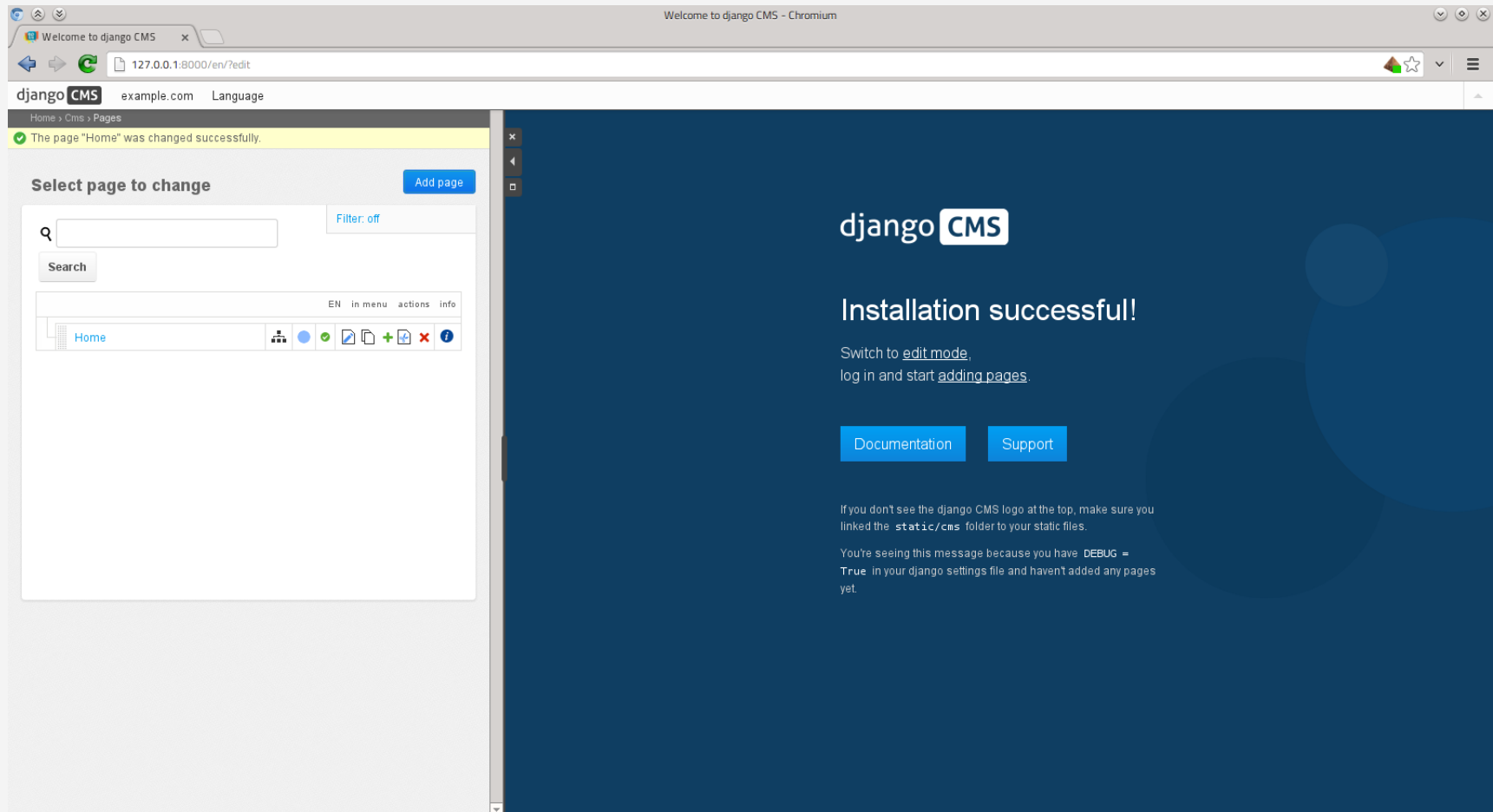
# Django-CMS First Steps

## Adding a start page

A screenshot of a web browser displaying the Django CMS admin interface. The browser's address bar shows the URL "127.0.0.1:8000/en/?edit". The page title is "Welcome to django CMS - Chromium". The interface is divided into two main sections. On the left is a sidebar with a "Home > Cms > Pages > Home" breadcrumb. Below this is the "Advanced Settings" section, which contains several form fields: "Template:" with a dropdown menu set to "Start page"; "Id:" with an empty text input; "Overwrite URL:" with an empty text input; "Redirect:" with an empty text input; a "Soft root" checkbox; "Attached menu:" with a dropdown menu; "Application:" with a dropdown menu; "Application instance name:" with an empty text input; and "X Frame Options:" with a dropdown menu set to "Inherit from parent page". At the bottom of the sidebar are links for "Basic Infos", "Save and continue editing", and a blue "Save" button. The main content area on the right has a dark blue background with the "django CMS" logo at the top. Below the logo, it says "Installation successful!". There are two links: "Switch to [edit mode](#)." and "log in and start [adding pages](#).". Below these are two blue buttons: "Documentation" and "Support". At the bottom, there is a paragraph of text: "If you don't see the django CMS logo at the top, make sure you linked the `static/cms` folder to your static files. You're seeing this message because you have `DEBUG = True` in your django settings file and haven't added any pages yet."

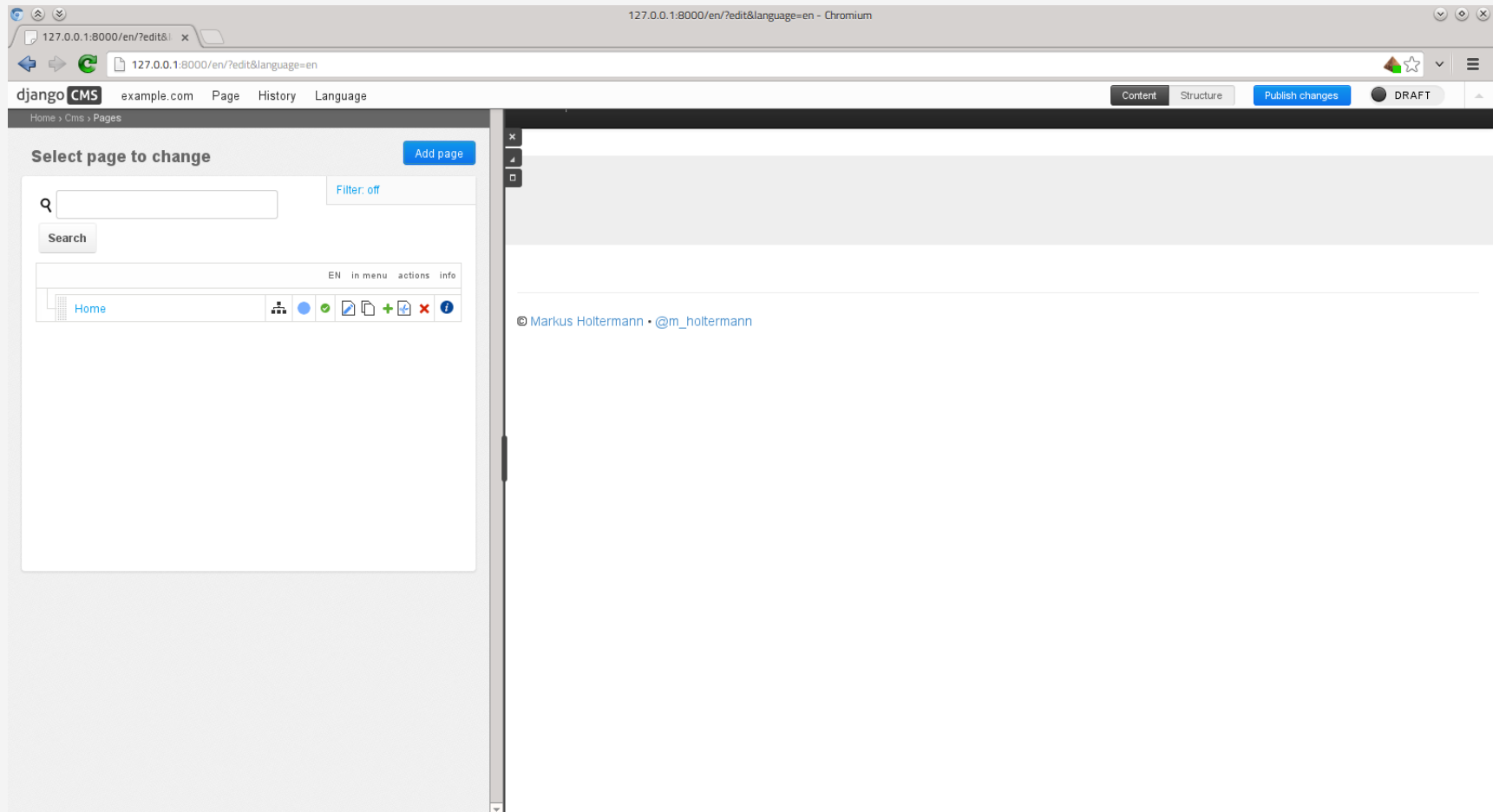
# Django-CMS First Steps

## Adding a start page



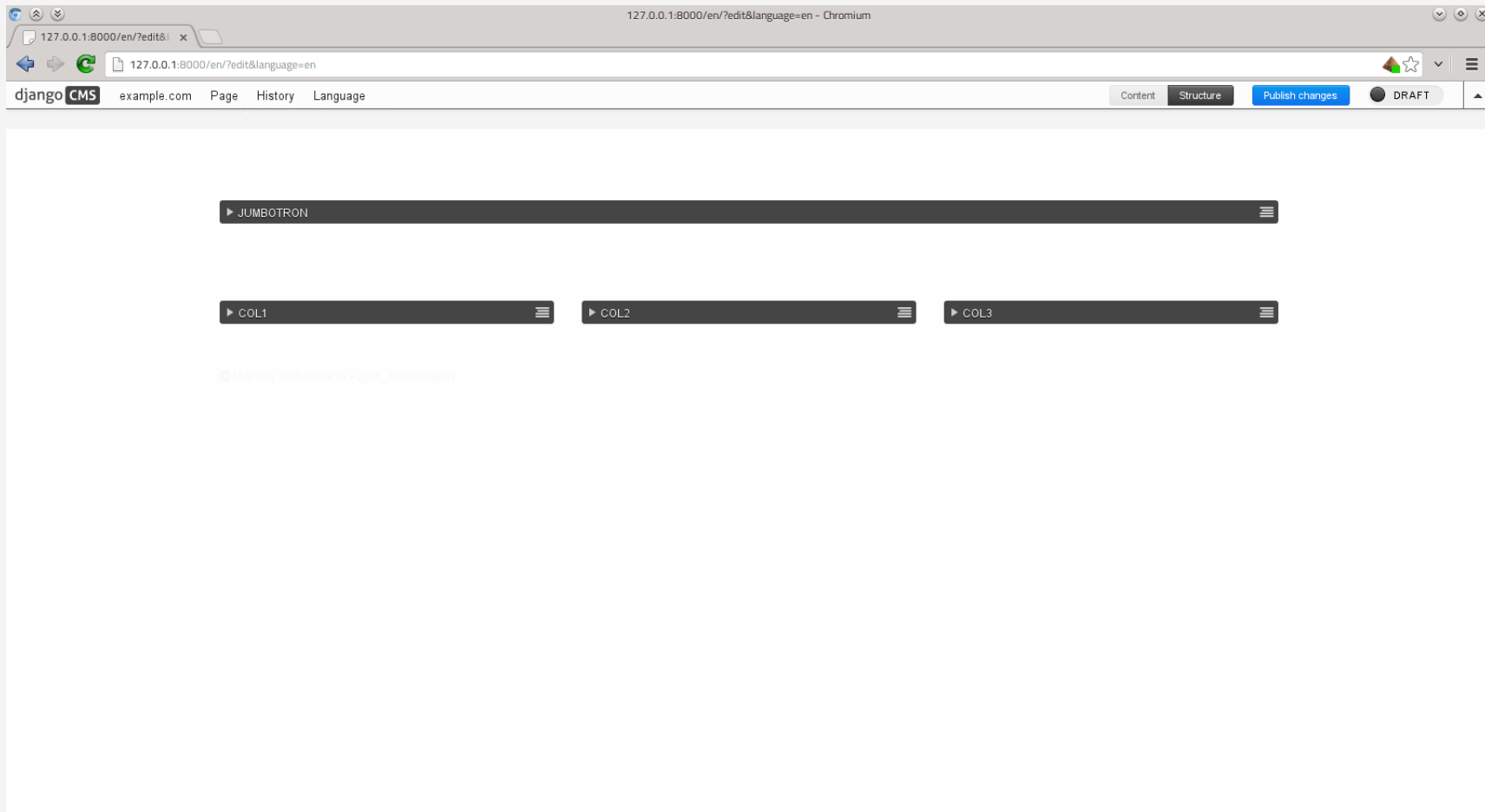
# Django-CMS First Steps

## Our new start page



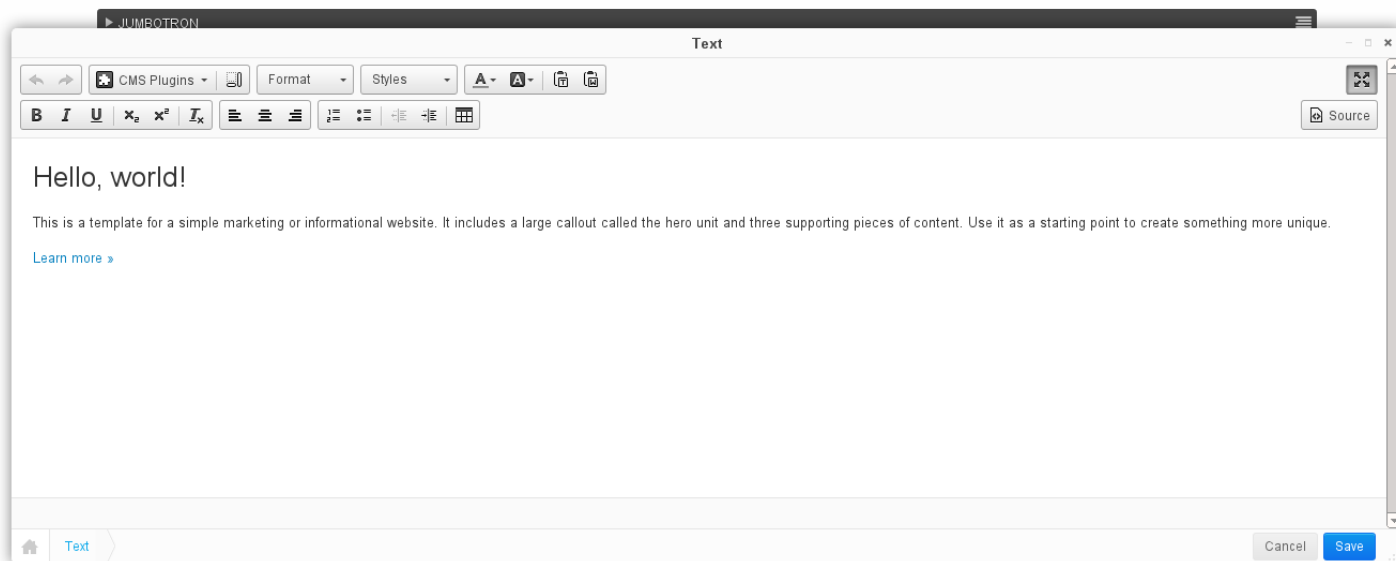
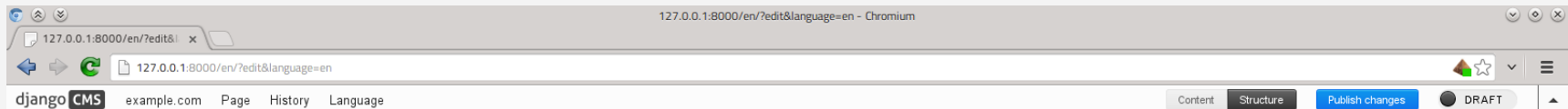
# Django-CMS First Steps

## Adding content to our page



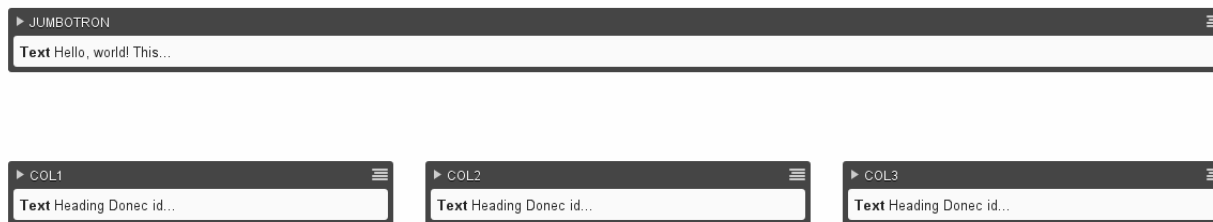
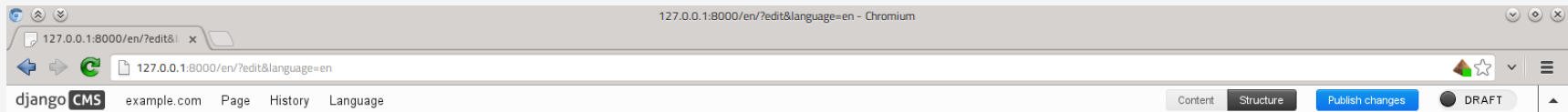
# Django-CMS First Steps

## Adding content to our page



# Django-CMS First Steps

## Adding content to our page

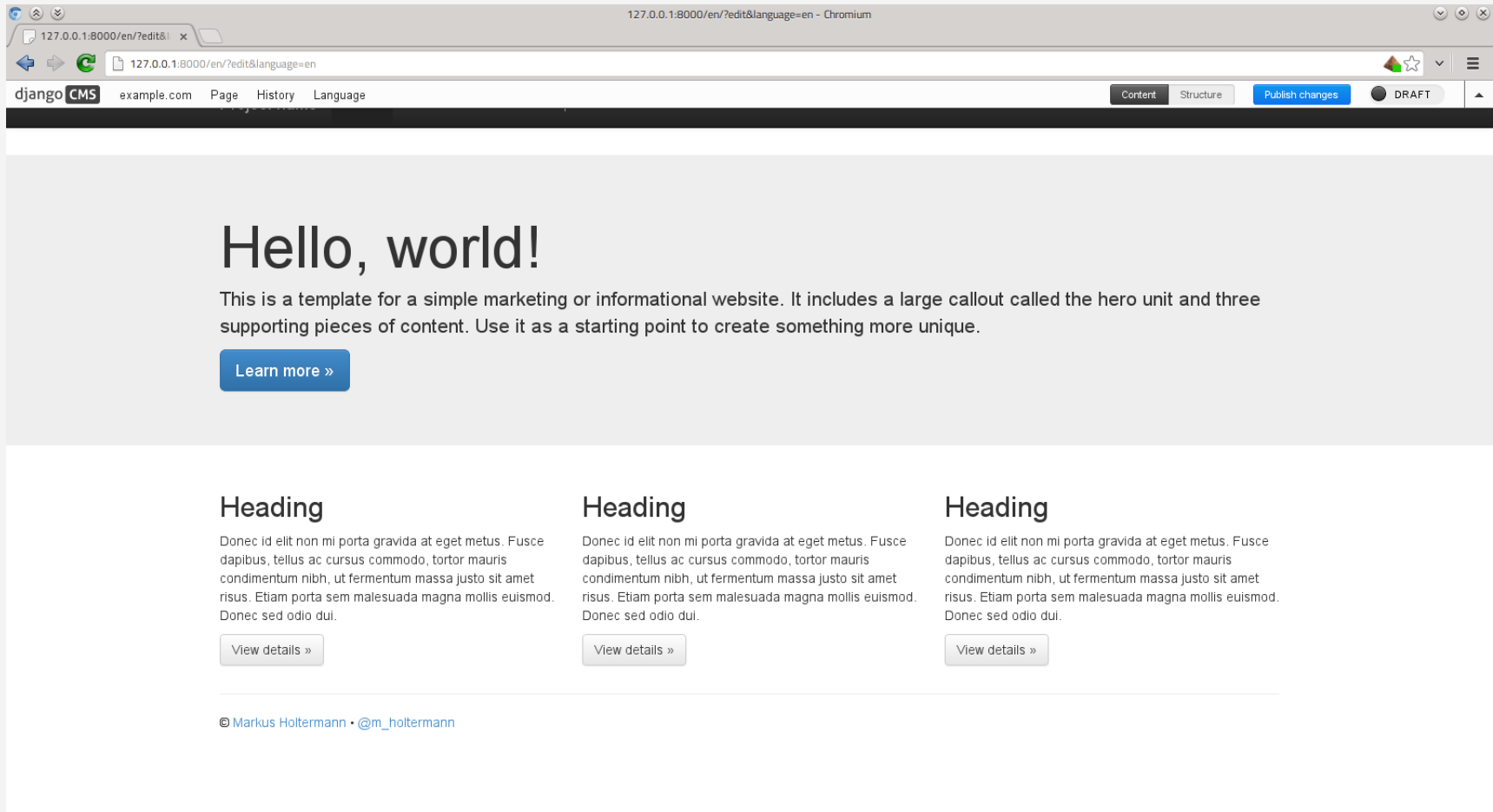


© Markus Holtermann • @m\_holtermann



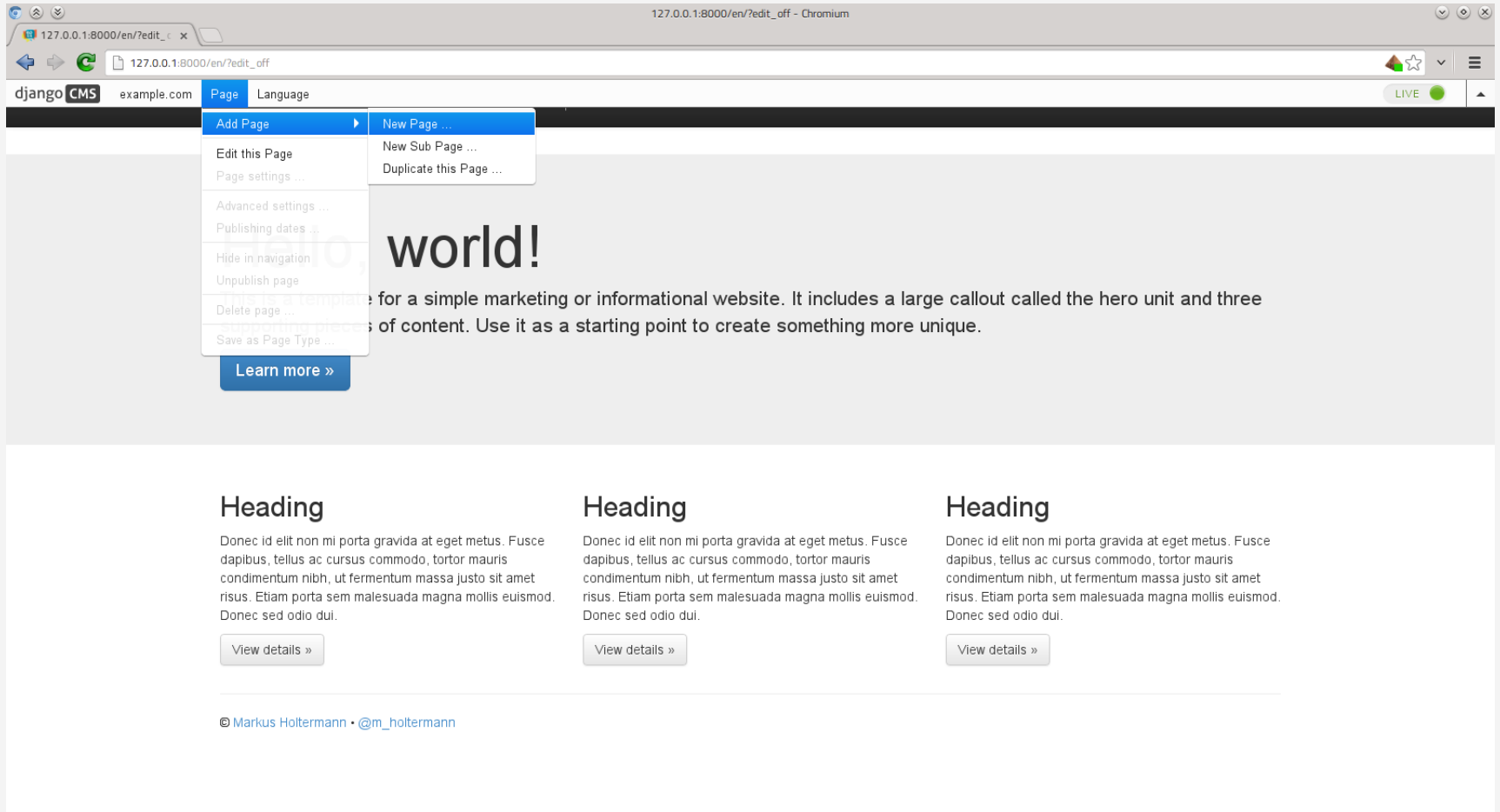
# Django-CMS First Steps

## The start page is ready



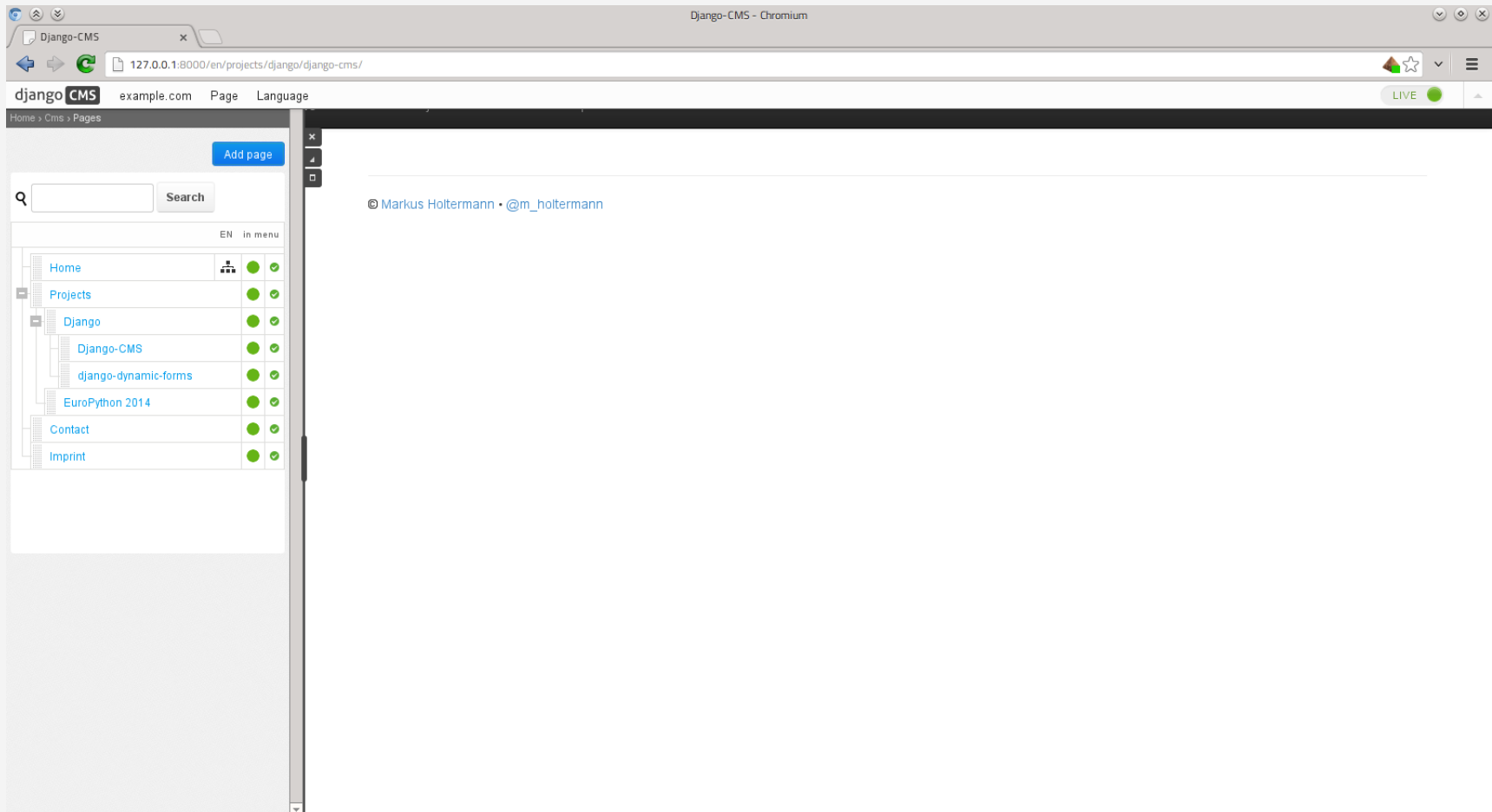
# Django-CMS Next Steps

## Creating some sub-pages



# Django-CMS Next Steps

## Complex page hierarchy



# Django-CMS Next Steps

## Getting the page in the menu

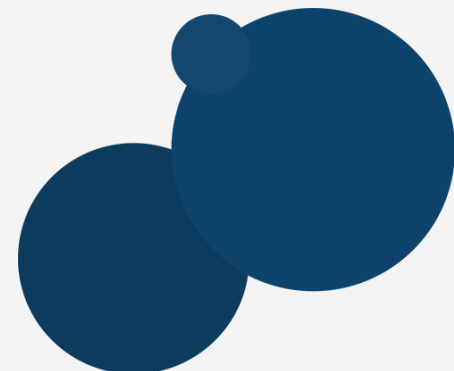


show\_menu template tag:

```
{% show_menu 0 100 100 1000 "menu/menu.html" %}
```

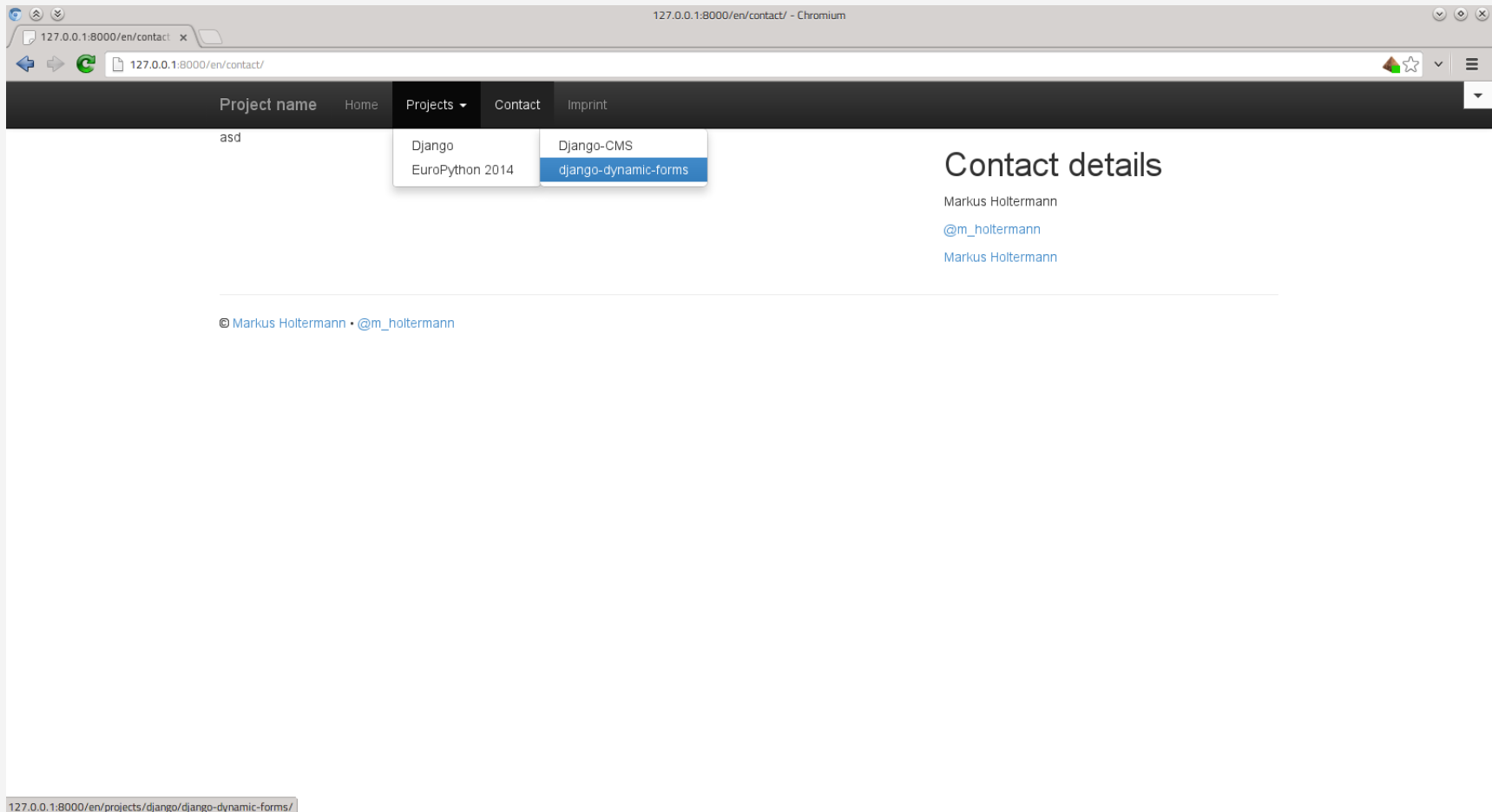
<http://django-cms.readthedocs.org/en/develop/advanced/templatetags.html>

```
{% show_menu from_level to_level extra_inactive extra_active  
"menu/menu_item.html" "" "" child %}
```



# Django-CMS Next Steps

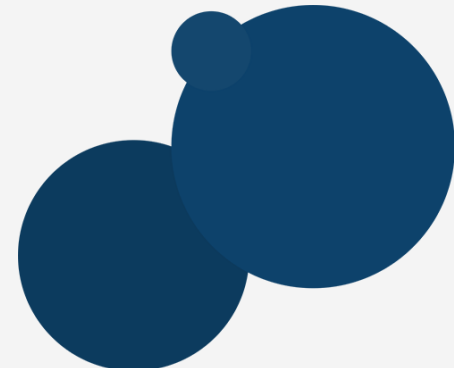
## Getting the page in the menu



# Demo

django **CMS**

- **Django-CMS Logo:**  
<https://github.com/divio/django-cms/blob/19d8441a/cms/static/cms/img/logo-white%402x.png>
- **Background Image:**  
<https://github.com/divio/django-cms/blob/687f4f8e/cms/static/cms/img/welcome-bg.png>
- **Presentation:**  
Creative Commons CC-BY-SA 3.0



- **This presentation and code:**  
<https://github.com/Markush2010/django-cms-introduction>
- **HTML Boilerplate from Initializr:**  
<http://www.initializr.com/>
- **Complex Django-CMS application:**  
<https://ep2014.europython.eu/>  
Code at <https://github.com/EuroPython/djep>

