

ООО «Открытые Библиотечные Системы»

---

**СИСТЕМА УПРАВЛЕНИЯ КОНТЕНТОМ И СЕРВИСАМИ  
БИБЛИОТЕЧНОГО ПОРТАЛА (САЙТА)**

# **БИБЛИОПОРТАЛ**

---

**Руководство по установке**

## Содержание

Содержание.....	2
Введение.....	3
1. Подготовка к установке.....	4
2. Установка программного обеспечения.....	5
3. Установка обновлений программного обеспечения.....	10
3.1 Обновление.....	10
3.2 Отмена обновления.....	10
3.3 Резервное копирование данных.....	10

## Введение

Документ содержит описание основных действий по установке программного продукта «Система управления контентом и сервисами библиотечного портала (сайта) «Библиопортал»» (далее – Система, Библиопортал) для ОС Ubuntu Server 18.04 x64, рекомендуемой платформы для функционирования ПП.

Установку выполняет администратор Системы или системный администратор.

## 1. Подготовка к установке

Требуется убедиться, что оборудование, на котором будет проводиться установка, соответствует минимальным системным требованиям:

ОС Ubuntu Server  $\geq$  18.04 x64, CPU: 2 ядра, ОЗУ: 2 Гб, 40 Гб дискового пространства.

Выполнить установку дополнительного программного обеспечения (ПО) для организации программного окружения Библиопортала.

Python, PostgreSQL, Nginx, Supervisor устанавливаются с использованием менеджера пакетов.

Команды выполняются от имени системного пользователя root.

Необходимо установить пакеты для сборки python библиотек:

```
$ apt install python-pip python-dev g++
```

## 2. Установка программного обеспечения

1. Распаковать в директорию `/var` архив с дистрибутивом системы. В результате система будет находиться по следующему пути:

`/var/portal_systems`.

2. Создать пользователя, от имени которого будет работать система. В данном руководстве будет использоваться пользователь **portal**.

3. Создать директорию `/var/portal_systems/var` и сделать её владельцем пользователя **portal**. В этой директории будут располагаться файлы, создаваемые в процессе работы системы.

4. Создать директории:

`/var/portal_systems/var/static` - хранит статические ресурсы шаблонов: css, js и т.п.

`/var/portal_systems/var/media` - хранит загружаемый пользователями контент.

5. Рекомендуется использовать виртуальное окружение **virtualenv** для изоляции python библиотек, необходимых для работы системы. Установка `virtualenv` производится следующей командой:

```
$ python -m pip install virtualenv
```

6. Перейти в директорию `/var/portal_systems/` и выполнить команду для создания виртуального окружения:

```
$ python -m virtualenv venv
```

7. Для включения контекста виртуального окружения необходимо выполнить команду:

```
$ source /var/portal_systems/venv/bin/activate
```

В результате выполнения команды приглашение командной строки будет выглядеть следующим образом:

```
(venv) $
```

8. Для установки python зависимостей, необходимых для работы системы, выполните команду:

```
(venv) $ pip install -r /var/portal_systems/libcms/requirements.txt
```

9. В СУБД PostgreSQL необходимо создать пользователя `portal` и базу `portal` с кодировкой `utf-8`.

10. Необходимо переименовать файл `/var/portal_systems/libcms/libcms/local_settings.py.tpl` в `local_settings.py`. В этом файле хранятся локальные настройки системы.

11. Открыть файл `local_settings.py` в редакторе и указать следующие параметры:

`SECRET_KEY` - изменить текущий на подобный с другими значениями. На его основе работают алгоритмы хеширования и подписей.

DEBUG - в режиме промышленной эксплуатации - False, при диагностике ошибок - True  
DOMAIN - основной домен работы системы.  
ALLOWED\_HOSTS - домены, которые разрешены при обращении к системе.  
ADMINS - список администраторов, которым будет отправляться уведомление об ошибках.  
DATABASES - настройка базы данных.

Пример:

```
DATABASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.postgresql',  
        'NAME': 'portal', # имя базы в PostgreSQL  
        'USER': 'portal', # пользователь  
        'PASSWORD': '123456',  
        'HOST': '127.0.0.1', # адрес сервера СУБД  
        'PORT': '5432', # порт  
        'CONN_MAX_AGE': None # постоянное подключение к базе  
    }  
}
```

LANGUAGE\_CODE - локализация системы по умолчанию

TIME\_ZONE - временная зона работы системы

Настройки отправки почты:

```
EMAIL_HOST = 'localhost'  
EMAIL_HOST_USER = 'user'  
EMAIL_HOST_PASSWORD = 'secret'  
EMAIL_PORT = 587  
EMAIL_USE_TLS = True
```

DEFAULT\_FROM\_EMAIL - почтовый адрес, который будет виден при отправке писем.

SESSION\_COOKIE\_DOMAIN - поддомен, на который распространяются куки.

Более подробную информацию по настройке и параметрам можно найти в руководстве фреймворка Django <https://docs.djangoproject.com/en/2.2/ref/settings/> (на англ. языке).

12. После изменения настроек необходимо инициализировать БД. Для этого необходимо перейти в директорию **/var/portal\_systems/portal** и выполнить команду:

```
(venv) $ python manage.py migrate
```

13. Далее необходимо создать суперпользователя:

```
(venv) $ python manage.py createsuperuser
```

и внести необходимые данные учетной записи.

14. Произвести деплой статических файлов командой:

```
(venv) $ python manage.py collectstatic
```

В результате выполнения, статические файлы копируются в `/var/portal_systems/var/static`, и размещенные там файлы будут доступны через Вэб-сервер.

15. Для проверки работоспособности установленной системы следует запустить встроенный веб сервер и проверить работу системы из браузера:

```
(venv) $ python manage.py runserver 0.0.0.0:8080
```

**Если установка прошла успешно, то появится интерфейс главной страницы портала.**

16. Настройка автозапуска. Система работает с использованием сервера **gunicorn**, который взаимодействует с сервером **nginx**. Для автозапуска и восстановления после сбоя необходимо настроить **supervisor**. Путь до файла настроек: `/var/portal_systems/portal$ cat /etc/supervisor/conf.d/portal.conf`

Содержимое файла:

```
[program:portal]
command=/var/portal_systems/venv/bin/gunicorn --workers=3 portal.wsgi:application --
bind 0.0.0.0:8080
directory=/var/portal_systems/libcms
user=portal
autorestart=true
redirect_stderr=true
```

Подробное описание параметров можно найти на сайте <http://supervisord.org/>

После изменения файла настроек необходимо инициализировать supervisor с новыми данными:

```
$ service supervisor reload
```

Для перезапуска системы, например, после обновления или смены конфигурации следует использовать команду:

```
$ supervisorctl restart portal
```

Вместо supervisor можно использовать systemd или другой диспетчер процессов.

17. Настройка сервера nginx производится по усмотрению администратора.

Пример конфигурации.

```
server {

    index index.html index.htm;
    server_name example.com;

    client_max_body_size 32m;

    gzip on;
```

```

gzip_disable "msie6";

gzip_vary on;
gzip_proxied any;
gzip_comp_level 6;
gzip_buffers 16 8k;
gzip_http_version 1.1;
gzip_types text/plain text/css application/json application/x-javascript text/xml
application/xml application/xml+rss text/javascript;

location / {
proxy_pass http://127.0.0.1:8080;
proxy_set_header Host $server_name;
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
}

location = /favicon.ico { access_log off; log_not_found off; }

location /static/ {
alias /var/portal_systems/var/static/;
}

location /media/ {
alias /var/portal_systems/var/media/;
}

listen 80;
listen [::]:80;
}

```

18. Настройка отправки почты по расписанию. От имени пользователя **portal** выполнить команду:

```
$ crontab -e
```

и ввести следующие команды, для периодического запуска:

```

* * * * /var/portal_systems/venv/bin/python /var/portal_systems/libcms/manage.py send_mail >>
/dev/null 2>&1
0,20,40 * * * * /var/portal_systems/venv/bin/python /var/portal_systems/libcms/manage.py
retry_deferred >> /dev/null 2>&1
0 0 * * * /var/portal_systems/venv/bin/python /var/portal_systems/libcms/manage.py
purge_mail_log 7 >> /dev/null 2>&1
0 * * * * /var/portal_systems/venv/bin/python /var/portal_systems/libcms/manage.py
portal_logger_run_watchers >> /dev/null 2>&1

```

Процесс настройки завершен.

Более подробную документацию по фреймворку Django можно найти на сайте <https://docs.djangoproject.com/en/2.2/>

## 3. Установка обновлений программного обеспечения

### 3.1 Обновление

Обновление происходит путем извлечения из архива программного кода новой версии и замены существующего в соответствии со структурой файлов и директорий.

### 3.2 Отмена обновления

Перед обновлением рекомендуется сделать резервную копию директории **/var/portal\_systems/libcms**. В случае обнаружения ошибок или некорректной работы при установке новой версии Системы, следует восстановить директорию из резервной копии. Далее необходимо выполнить деплой статических файлов, которые (см. п. 14 в разделе «Установка»).

### 3.3 Резервное копирование данных

Резервное копирование данных осуществляется средствами СУБД. Файлы и медиаданные, загруженные во время эксплуатации, хранятся в директории **/var/portal\_systems/var/media**