

---

# Rynda Documentation

*Выпуск 0.1*

Valery A. Ilychev

июн. 13, 2017



<b>1</b>	<b>О проекте</b>	<b>3</b>
1.1	Возможности . . . . .	3
1.2	История . . . . .	3
<b>2</b>	<b>Быстрый старт</b>	<b>5</b>
<b>3</b>	<b>Установка Рынды в «боевом» режиме</b>	<b>7</b>
3.1	Настройки VirtualHost . . . . .	7
3.2	Настройки обработки статических файлов . . . . .	8
3.3	Пример описания виртуального хоста для Apache . . . . .	8
<b>4</b>	<b>Indices and tables</b>	<b>11</b>



“Виртуальная Рында” - это средство для координации взаимопомощи. Сайт позволяет каждому сообщить о своей беде или желании помочь через различные каналы (Интернет, мобильная связь). Затем такие сообщения систематизируются и наносятся на карту. “Виртуальная рында” написана на python с использованием фреймворка Django.

Содержание:



Краткое описание проекта

### Возможности

- Создание и хранение разовых сообщений (просьб о помощи и предложений).
- Отображение сообщений на карте (на главной странице и на странице сообщения).
- Жизненный цикл сообщения (новое - непроверенное - проверенное - закрытое).
- Поиск по просьбам и предложениям - по категориям, типу, содержанию (поиск по тексту сообщения)
- Регистрация на сайте в качестве постоянно действующего волонтера.
- Информационные страницы (можно размещать произвольный текст)
- Новости сайта - можно вести микроблог.
- Интерфейс администратора и модератора, позволяющий полноценно управлять сообщениями и пользователями.

### История

Деятельность блоггеров и пользователей Рунета во время пожаров лета 2010 года показала огромный потенциал взаимопомощи среди российских граждан в целом и пользователей Интернета в частности. Не менее важно, что для раскрытия этого потенциала необходима единая информационная платформа для координации действий людей, позволяющая сделать оказание помощи более эффективным.

Именно тогда родился наш первый проект, “[Карта помощи пострадавшим от пожаров](#)”. Этот сайт стал базой данных, с одной стороны, позволяющей каждому предоставить информацию, а с другой — систематизировать ее согласно актуальности, времени, месту и типу сообщения. Координационный

центр, работавший на базе «Карты», позволил значительно повысить эффективность оказания помощи через согласование действий между теми, кто хотел помочь, и теми, кто в помощи нуждался.

Затем была открыта «Карта помощи в холода». После трагедии в Японии — «Карта радиации».

Со временем на эти сайты стали приходить сообщения, не связанные с темами пожаров, холодов и техногенных катастроф. Стало очевидно, что нужна единая база, куда человек может обратиться за помощью и советом. Так родилась идея «Виртуальной Рынды».



---

### Быстрый старт

---

0. Создаем и активируем virtualenv:

```
$ virtualenv rynda  
$ source rynda/bin/activate
```

1. Клонировем репозиторий:

```
$ git clone https://github.com/sarutobi/Rynda.git
```

2. Переходим в клонированный репозиторий:

```
$ cd Rynda
```

3. Устанавливаем все зависимости:

```
$ pip install -r requirements\test.txt
```

4. Копируем mysettings.py.example в mysettings.py

5. Задаем структуру базы данных и пароль суперпользователя:

```
$ bash createdb.sh
```

Дважды вводим пароль суперпользователя. Имя суперпользователя по умолчанию: *admin*

6. Запускаем локальный сервер:

```
$ python manage.py runserver
```

7. Открываем в браузере <http://localhost:8000>

8. Чтобы протестировать систему запускаем:

```
$ python manage.py test
```



---

## Установка Рынды в «боевом» режиме

---

Для установки в боевом режиме под веб сервером Apache потребуется установить модуль `mod-wsgi`. В ОС Debian это можно сделать командой:

```
$ sudo aptitude install libapache2-mod-wsgi
```

**Примечание:** Стоит заметить, что если у вас стоит модуль `mpm-itk`, то он может быть причиной того, что модуль `mod-wsgi` не будет работать.

---

## Настройки VirtualHost

Запуск в режиме демона является рекомендуемым методом запуска `mod_wsgi` на не Windows платформах. Чтобы создать процесс, в описание `VirtualHost` потребуется добавить директивы `WSGIDaemonProcess`, `WSGIProcessGroup` и `WSGIScriptAlias`.

- `WSGIDaemonProcess` - основная директива настройки демона, её формат:

```
WSGIDaemonProcess имя_процесса processes= threads= python-path=
```

где:

```
python-path - это путь к библиотекам виртуального окружения сайта в формате:  
/path/to/mysite.com:/path/to/your/venv/lib/python2.X/site-packages
```

```
processes - определяет количество процессов демона, которые должны быть запущены  
в этой группе процессов, определенной директивой WSGIProcessGroup. По умолчанию  
(когда кол-во не определено) будет запущен только один процесс.
```

```
threads - определяет кол-во потоков для обработки запросов по каждому из процессов  
демона внутри группы процессов, определенных директивой WSGIProcessGroup.
```

- WSGIProcessGroup - группирует процессы демона:

```
WSGIProcessGroup имя_процесса
```

- WSGIScriptAlias - местоположение скрипта, обрабатывающего запросы к приложению:

```
WSGIScriptAlias site_root /path/to/mysite.com/wsgi.py
```

Пример:

```
<Directory /path/to/mysite.com/mysite/Rynda/scripts>
<Files wsgi.py>
    Require all granted
</Files>
</Directory>
```

---

**Примечание:** если вы используете Apache версии 2.4 или старше, замените одну строку `Require all granted` на

```
Order deny, allow
Allow from all
```

---

## Настройки обработки статических файлов

В отличие от сайтов, написанных на php, Django не обрабатывает статические файлы (изображения, таблицы стилей, скрипты). Их обработка должна настраиваться отдельно.

Для сервера Apache:

```
Alias /static/ /path/to/mysite.com/static
```

Для сервера nginx:

```
location /static {alias /path/to/mysite.com/static; }
```

---

**Примечание:** рекомендуется использовать nginx в качестве backend'а для раздачи статических файлов, так как он делает это более эффективно, нежели Apache

---

Теперь запустим команду по сбору статических файлов, которая в автоматическом режиме разместит все статические файлы в правильном месте:

```
./manage.py collectstatic
```

---

## Пример описания виртуального хоста для Apache

```
<Directory /path/to/my/site/www/demo.openrynda.com/Rynda/rynda/Rynda>
    <Files wsgi.py>
        Order allow,deny
        Allow from all
```

```
</Files>
</Directory>

<VirtualHost 1.2.3.4:81 >
    ServerName demo.openrynda.com
    DocumentRoot /path/to/my/site/www/demo.openrynda.com
    WSGIDaemonProcess openrynda processes=5 threads=1 python-path=/path/to/my/site/www/demo.
↪openrynda.com/Rynda:/path/to/my/site/www/demo.openrynda.com/rynda/lib/python2.7/site-packages:/
↪path/to/my/site/www/demo.openrynda.com/rynda/lib/python2.7/
    WSGIProcessGroup openrynda
    WSGIScriptAlias / /path/to/my/site/www/demo.openrynda.com/rynda/Rynda/wsgi.py

    Alias /static/ /path/to/my/site/www/demo.openrynda.com/static # расположение статика (при
↪необходимости измените)
    ServerAlias www.demo.openrynda.com
</VirtualHost>
```



---

Indices and tables

---

- genindex
- modindex
- search