

---

# SOFTWARE REQUIREMENTS SPECIFICATION

Release 1.0

# Оглавление

1	Структура проекта	3
2	Общее описание проекта	4
3	Этапы разработки	6
4	Mindmap	7

# 1 Структура проекта

Библиотеки, фреймворки и пр. используемое в проекте

Routing, Api:

- Slim micro-framework  
<https://www.slimframework.com/>

Basic access authentication:

- Slim-basic-auth middleware  
<https://github.com/tuupola/slim-basic-auth>

ORM, Event dispatcher:

- Illuminate  
<https://github.com/illuminate/database>  
<https://laravel.com/docs/5.7/eloquent>

Mailer:

- PHPMailer  
<https://github.com/PHPMailer/PHPMailer>

Templating engine:

- Twig  
<https://twig.symfony.com/>

Frontend:

- AdminLTE  
<https://github.com/almasaeed2010/AdminLTE>
- Webpack bundler  
<https://webpack.js.org/>
- Vue.js framework  
<https://vuejs.org/>

Datamining, Web-Sites Monitoring:

- Requests  
<http://docs.python-requests.org>

## 2 Общее описание проекта

Скрипт на питоне будет проверять состояние сайтов из списка урлов в базе, отправляя им http-реквесты. Получение списка страниц для проверки и сохранение ответов производится соответствующими реквестами к RESTful api административной панели. Скрипт стартует крон-задачей. За рассылку http-реквестов отвечает библиотека Requests. Данные проверяемых страниц хранятся в таблице SMPages.

Для административной панели используется микрофреймворк Slim. Он особенно удобен при создании RESTful api и в будущем его возможности можно будет использовать для координации микросервисов. Особенно - если вдруг потребуются создание демонов на внешних серверах (их нельзя разворачивать на бегете).

Для авторизации запросов используется HTTP Basic authentication. Все запросы к api также проходят эту аутентификацию. Из-за того что передача авторизационных данных идёт открытым текстом необходимо включить https с принудительной переадресацией.

Административная панель содержит следующие формы:

1. Общие настройки - почтовый адрес для уведомлений, доменное имя сайта, чекбокс для включения https при авторизации
2. Список сайтов - просмотр статусов сайтов. Если та или иная страница сайта имеет статус отличный от 200, статус сайта будет подсвечен соответствующим цветом. Есть возможность удалить все страницы сайта и отредактировать имя домена
3. Список страниц - просмотр статусов страниц с сортировкой по статусу и имени, редактирование и удаление страниц из списка, фильтр для поиска страниц по имени или урлу. Есть возможность отметить и удалить группу страниц.
4. Добавление новой страницы - url, домен сайта, имя/title страницы, использование https.
5. Массовый импорт - текстовое поле для построчного перечисления страниц для импорта. В конце импорта обнаруженные ошибки в импортируемых данных будут отображены в этом же текстовом поле

В списке страниц статусы будут подсвечиваться красным (404), оранжевым(301,302) и зеленым(200). Если у сайта есть страницы с красным или оранжевым статусом он будет подсвечен соответствующим цветом. Красный в приоритете над оранжевым. В случае переадресации можно будет просмотреть конечный адрес.

RESTful api позволяет просматривать (в ответ придет application/json) и изменять состояние сайтов из списка в базе. В момент обновления состояния той или иной страницы, выстреливает событие. Если новый статус ответа 404 - будет отправлено email уведомление на почтовый ящик администратора.

Административная панель разбита на зоны доступа (**Realm в терминологии HTTP Basic auth**). При переходе между ними потребуется подтвердить авторизацию. Соответствие “зона доступа : пользователь” хранится в таблице Permissions. В таблице Realms хранится имя зоны доступа и входящие в неё url-адреса.

### 3 Этапы разработки

№	Время(ч)	Описание этапа
1	2	Разработка структуры таблиц БД
2	6	Создание структуры проекта и настройка основных модулей
3	2	Создание шаблона главной темы административной панели
4	2	Форма общих настроек
5	3	Форма просмотра списка сайтов
6	5	Форма просмотра списка страниц
7	2	Форма добавления страницы
8	5	Форма массового импорта
9	4	Скрипт для проверки состояния страниц
10	6	Деплой и тестирование на боевом сервере
Σ	37	

## 4 Mindmap

