

# Проект часть 6

---

Матюшкин Денис Владимирович (НПИбд-02-21)

30.05.2022

RUDN University, Moscow, Russian Federation

## Цель работы

---

- Размещение двуязычного сайта на Github.

## Ход работы

---

## 1. Редактирование конфига

- Для добавления нового языка сайта необходимо перейти к каталогу сайта. После перейдем в `config/_default/`. Откроем файл `config.yaml` и разрешим добавление второго языка (`defaultContentLanguageInSubdir: true`), еще сделаем язык сайта по умолчанию русским, чтобы при заходе открывался русский сайт, а также изменим имя сайта, чтобы не было потом проблем при добавлении заголовка в файле `languages.yaml` (рис. 1).

```
title: Матюшкин Д.В. # Website name
baseURL: 'https://stifell.github.io/' # Website URL

#####
## LANGUAGE
#####

defaultContentLanguage: ru
hasCJKLanguage: false
defaultContentLanguageInSubdir: true
removePathAccents: true
```

Рис. 1: Редактирование `config.yaml`

## 2. Добавление заголовков для русского языка

- В том же каталоге откроем файл *languages.yaml* и добавим менюшку для русского языка, а также заголовок в виде имени автора (рис. 2).

```
en:
  languageCode: en-us
  contentDir: content/en
  title: Matyushkin D.V.

# Default language
ru:
  languageCode: ru-ru
  contentDir: content/ru
  title: Матюшкин Д.В.
  menu:
    main:
      - name: Обо мне
        url: '#about'
        weight: 10
      - name: Посты
        url: '#posts'
        weight: 20
      - name: Проекты
        url: '#projects'
        weight: 30
      - name: Мероприятия
        url: '#talks'
        weight: 40
      - name: Публикации
        url: '#featured'
        weight: 50
      - name: Контакты
        url: '#contact'
        weight: 60
```

Рис. 2: Редактирование languages.yaml

### 3. Создание папок для языков

- В каталоге *content* добавим 2 папки для разных языков соответственно. Так как все это время заполнял сайт на английском, пришлось все посты и данные переводить на русский язык (рис. 3).

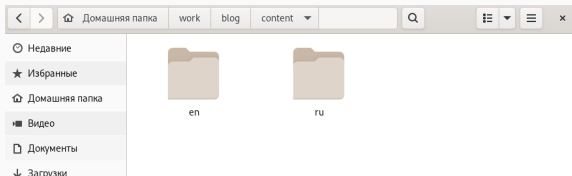


Рис. 3: Создание папок для языков

## 4. Создание поста по прошедшей неделе

- Создадим пост по прошедшей неделе на двух языках (рис. 4 и рис. 5).

```
# Article

## How I spent the last week

1. The beginning of the week turned out to be easy, so on Monday I went in for sports and read a book.

2. Tuesday and Wednesday were generally no different, except that I started doing labs on operating systems.

3. Friday is a hard day. I spent the first two classes at the computer, learning C++. From 15:00 to 19:20 I studied English.

4. On Saturday from 12:00 to 14:50 there were couples for computer practice. We have done lab work on Arduino.

5. On Sunday, I completely rested mentally and physically.
```

Рис. 4: Создание поста по прошедшей неделе en

```
## Как я провел прошлую неделю

1. Начало недели выдалось легким, поэтому в понедельник я занимался спортом и читал книгу.

2. Вторник и среда в целом ничем не отличались, разве что я начал делать лабораторные работы по операционным системам.

3. Пятница – тяжелый день. Первые две пары я провел за компьютером, изучая C++. С 15:00 до 19:20 учил английский язык.

4. В субботу с 12:00 до 14:50 по компьютерному практику были пары. Мы сделали лабораторную работу на Arduino.

5. В воскресенье полностью отдохнул морально и физически.
```

Рис. 5: Создание поста по прошедшей неделе ru



## 5. Создание поста на выбор

- Создадим пост на тему: “статический генератор сайтов hugo” на двух языках (рис. 6 и рис. 7).

### **## Overview**

Hugo is a static HTML and CSS website generator written in [Go][].  
It is optimized for speed, ease of use, and configurability.  
Hugo takes a directory with content and templates and renders them into a full HTML website.

Hugo relies on Markdown files with front matter for metadata, and you can run Hugo from any directory.  
This works well for shared hosts and other systems where you don't have a privileged account.

Hugo renders a typical website of moderate size in a fraction of a second.  
A good rule of thumb is that each piece of content renders in around 1 millisecond.

Hugo is designed to work well for any kind of website including blogs, tumblers, and docs.

### **### Supported Architectures**

Рис. 6: Создание поста на выбор en

### **## Обзор**

Хьюго – это статический генератор веб-сайтов HTML и CSS, написанный на Go.  
Он оптимизирован для обеспечения скорости, простоты использования и возможности настройки.  
Хьюго берет каталог с контентом и шаблонами и превращает их в полноценный HTML-сайт.

Хьюго полагается на файлы Markdown с исходным кодом для метаданных, и вы можете запустить Hugo из любого каталога.  
Это хорошо работает для общих хостов и других систем, где у вас нет привилегированной учетной записи.

Хьюго рендерит типичный веб-сайт среднего размера за доли секунды.  
Хорошее эмпирическое правило заключается в том, что каждый фрагмент контента отображается примерно за 1 миллисекунду.

Хьюго разработан так, чтобы хорошо работать с любым типом веб-сайта, включая блоги, tumblers и docs.

### **### Поддерживаемые архитектуры**

Рис. 7: Создание поста на выбор ru

## 6. Проверка постов на сайте

- Зайдем в наш сайт и проверим изменения (рис. 8 и рис. 9).

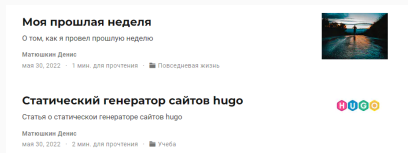


Рис. 8: Проверка постов ru

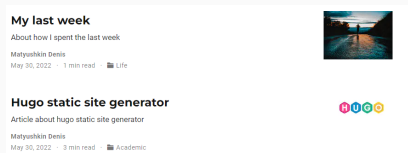


Рис. 9: Проверка постов en

## Заключение

---

В ходе этой лабораторной работы мы разместили двуязычный сайта на Github.

Спасибо за внимание!