

# **Отчёт по 2 этапу проекта**

**Сайт научного работника**

Викторов Егор

# Содержание

1	Цель работы	4
2	Выполнение работы	5
3	Выводы	8

# List of Figures

2.1	Файл об авторе . . . . .	5
2.2	Файл для поста . . . . .	6
2.3	Файл для публикации . . . . .	6
2.4	hugo сборка . . . . .	7

# 1 Цель работы

Добавить к сайту данные о себе.

## 2 Выполнение работы

Заполняю файл с данными о владельце сайта.



```
Открыть ▾ ⊕ _index.md
~/site/blog/content/authors/admin

# Full name (for SEO)
first_name: Егор
last_name: Викторов

# Status emoji
status:
  icon: 🧑

# Is this the primary user of the site?
superuser: true

# Highlight the author in author lists? (true/false)
highlight_name: true

# Role/position/tagline
role: Студент

# Organizations/Affiliations to display in Biography block
organizations:
  - name: RUdо
    url: https://rudo.ru/

# Social network links
# Need to use another icon? Simply download the SVG icon to your `assets/media/icons/` folder.
profiles:
  - icon: brands/github
    url: https://github.com/34T0N4P0P4N/
  - icon: academics/google-scholar
    url: https://scholar.google.com/

interests:
  - Криптовалюта
  - Трейдинг
  - Python

education:
```

Figure 2.1: Файл об авторе

Заполняю файл с текстом поста.

```
---
title: Моя неделя с 30.09 по 5.10
summary: Моя неделя
date: 2024-10-03

# Featured image
# Place an image named `featured-image.png` in this page's folder and customize its options here.
#image:
# caption: 'Image credit: [**Unsplash**](https://unsplash.com)'

authors:
- admin

tags:
- Academic
- MyWeek
---

На этой неделе я выбрал научного руководителя и утвердил тему RKP.
|
```

Figure 2.2: Файл для поста

Заполняю файл с текстом публикации.

```
Управление версиями — это процесс отслеживания изменений в файлах проекта. Благодаря этому можно:
- Сохранять историю изменений
- Возвращаться к предыдущим версиям
- Работать с несколькими участниками над одним проектом

## Git — основа управления версиями

Git — это распределённая система управления версиями, которая позволяет:
- Локально хранить всю историю изменений
- Работать с ветками
- Интегрировать изменения от разных участников

### Основные команды Git

1. **git init** — Инициализация репозитория в текущей директории.
2. **git clone <url>** — Копирует удалённый репозиторий на локальную машину.
3. **git add <файл>** — Добавляет файл в индекс (staging area) для последующего коммита.
4. **git commit -m "Сообщение о коммите"*** — Создаёт снимок состояния файлов и сохраняет его в истории.
5. **git branch <имя_ветки>** — Создаёт новую ветку.

Получает последние изменения из удалённого репозитория и объединяет их с локальными.

### Основные рабочие процессы в Git:

1. **Git Flow**: Популярная модель, которая предполагает использование двух основных веток: `master` для стабильных релизов и `develop` для текущей разработки.
2. **Feature Branch Workflow**: Ветвление для каждой новой функции, с последующим слиянием в основную ветку.
```

Figure 2.3: Файл для публикации

Перекомпилирую сайт

```
egorviktorov@fedora:~/site/blog$ hugo
WARN deprecated: site config key paginate was deprecated in Hugo v0.128.0 and will be removed in a future release. Use pagination.pageSize instead.
Start building sites ...
hugo v0.134.2-1c74abd26070b0c12849550c974a9f3f1e7afb06+extended linux/amd64 BuildDate=2024-09-10T10:46:33Z VendorInfo=gohugoio

-----+-----+
Pages | 28
Paginator pages | 0
Non-page files | 2
Static files | 1
Processed images | 11
Aliases | 6
Cleaned | 0

Total in 650 ms
egorviktorov@fedora:~/site/blog$
```

Figure 2.4: hugo сборка

## **3 Выводы**

Добавили к сайту данные о себе.