

# **Отчёт по 5 этапу проекта**

**Сайт научного работника**

Тимофей Абакумов

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение работы</b>	<b>6</b>
<b>3</b>	<b>Выводы</b>	<b>10</b>

# Список иллюстраций

2.1	Файл о проекте . . . . .	7
2.2	Файл для поста . . . . .	8
2.3	Файл для публикации . . . . .	9

## **Список таблиц**

# 1 Цель работы

Добавить к сайту данные о себе.

## **2 Выполнение работы**

Заполняю файл с информацией о проекте.

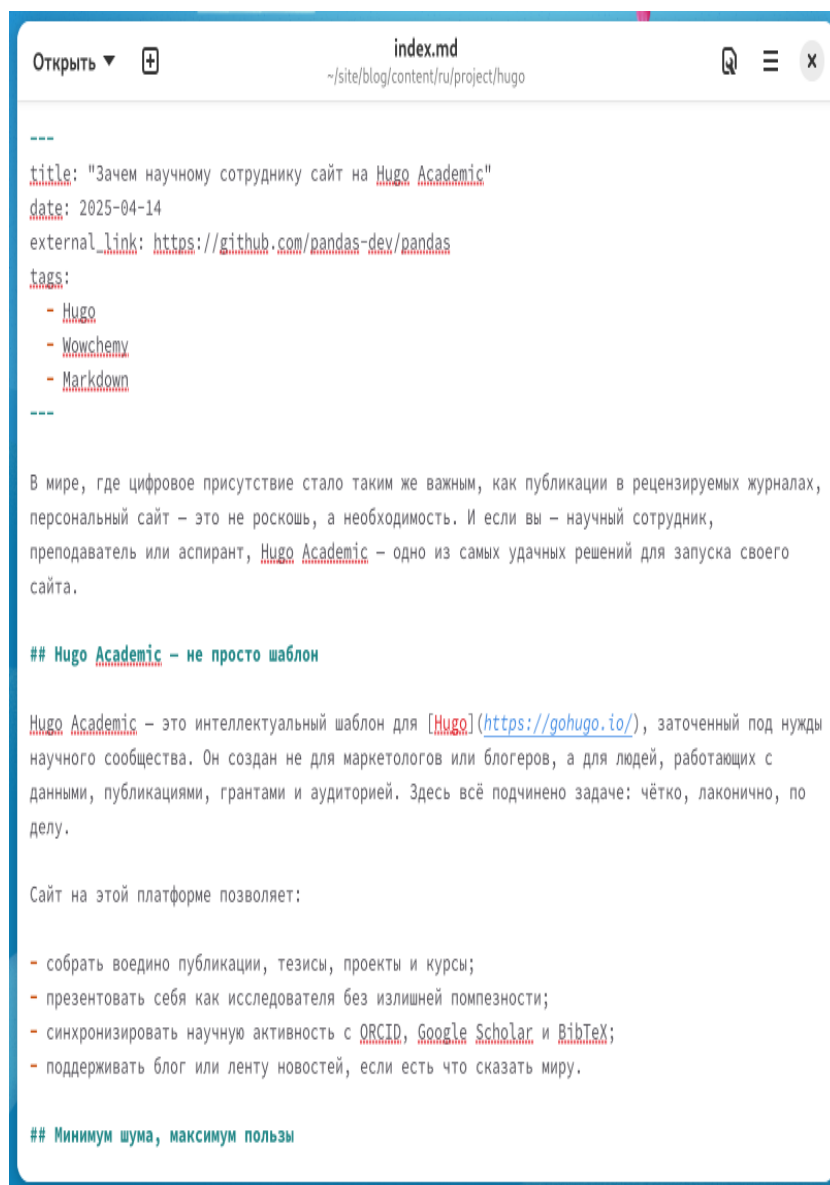
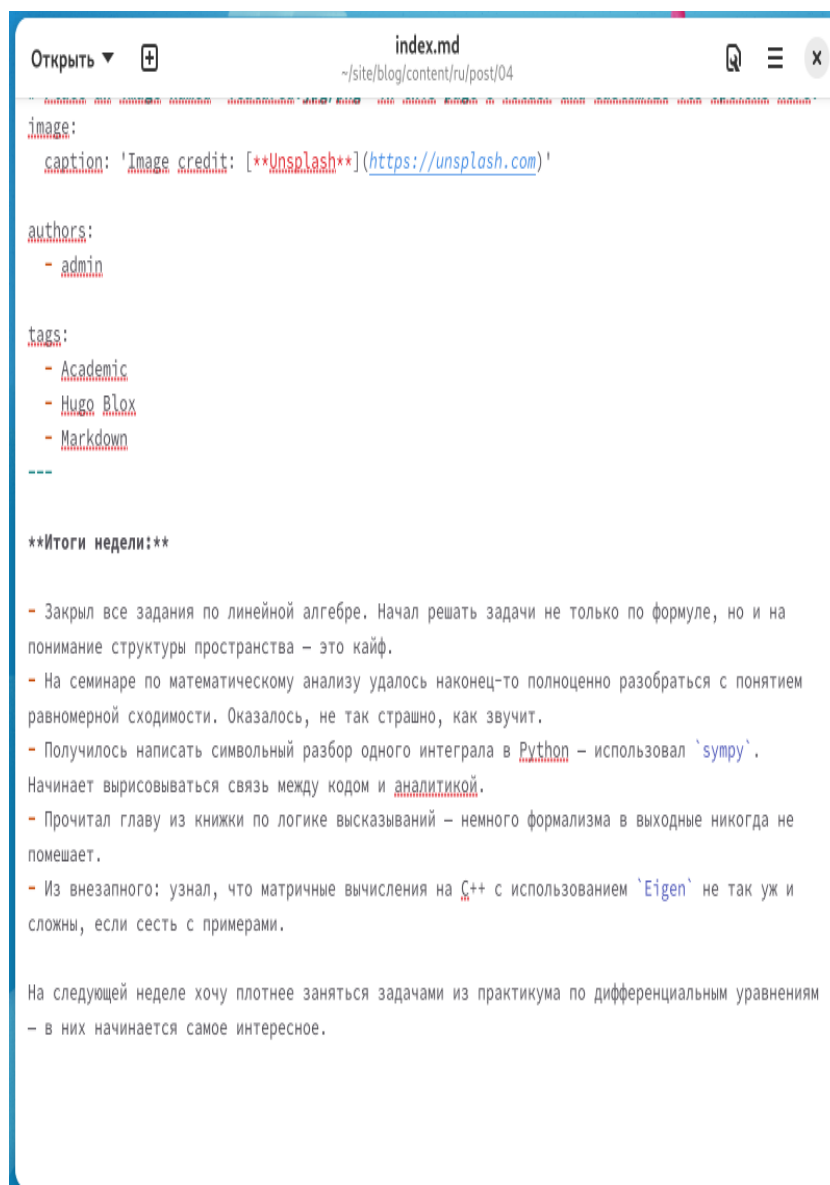


Рис. 2.1: Файл о проекте

Заполняю файл с текстом поста.



```
index.md
~/site/blog/content/ru/post/04

image:
  caption: 'Image credit: [Unsplash](https://unsplash.com)'

authors:
  - admin

tags:
  - Academic
  - Hugo Blog
  - Markdown
  ---

**Итоги недели:**

- Закрыв все задания по линейной алгебре. Начал решать задачи не только по формуле, но и на понимание структуры пространства – это кайф.
- На семинаре по математическому анализу удалось наконец-то полноценно разобраться с понятием равномерной сходимости. Оказалось, не так страшно, как звучит.
- Получилось написать символический разбор одного интеграла в Python – использовал `sympy`.
  Начинает вырисовываться связь между кодом и аналитикой.
- Прочитал главу из книжки по логике высказываний – немного формализма в выходные никогда не помешает.
- Из внезапного: узнал, что матричные вычисления на C++ с использованием `Eigen` не так уж и сложны, если сесть с примерами.

На следующей неделе хочу плотнее заняться задачами из практикума по дифференциальным уравнениям
– в них начинается самое интересное.
```

Рис. 2.2: Файл для поста

Заполняю файл с текстом публикации.



```
---

### **1. Fortran – вечная классика**

Fortran – старейший из живущих языков, разработан в 1950-х. И несмотря на возраст, до сих пор используется в задачах вычислительной физики, численного моделирования и гидродинамики. Почему? Потому что он:

- невероятно быстр на операциях с массивами и числами с плавающей точкой;
- отлично распараллеливается;
- поддерживается в большинстве научных библиотек, особенно в старых.

Но учить его с нуля – дело спорное. Скорее, стоит знать о нём и уметь читать код, особенно если работать с "наследием".

---

### **2. Python – универсальный солдат**

Python – любимец научного сообщества. Его плюсы:

- простота синтаксиса;
- богатейшая экосистема ('NumPy', 'SciPy', 'SymPy', 'Pandas', 'Matplotlib', 'Jupyter');
- легко соединяется с C/C++ и Fortran через 'ctypes', 'cython', 'f2py' и другие мосты.

Но важно понимать: Python – это не про скорость. Он медленный. Зато его удобно использовать как «склеивающий» язык, который организует весь научный процесс.

---

### **3. C и C++ – точность и контроль**

```

Рис. 2.3: Файл для публикации

Перекомпилирую сайт

## **3 Выводы**

Добавили к сайту данные о себе.