

Федеральное государственное автономное образовательное учреждение высшего
образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра «Смарт-технологии»

Направление подготовки/ специальность: 09.03.01 Разработка инженерного ПО

ОТЧЕТ

по проектной практике

Студент: Чуй Александр Чэнжэнович Группа: 231-324

Место прохождения практики: Московский Политех, кафедра «Смарт-технологии»

Отчет принят с оценкой _____ Дата _____

Руководитель практики: Сивцев Алексей Олегович

Москва 2025

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
1. Изучение Git и GitHub.....	4
2. Изучение синтаксиса Markdown.....	5
3. Разработка сайта, используя генератор статических сайтов Hugo.....	6
4. Взаимодействие с организацией партнером.....	7
ЗАКЛЮЧЕНИЕ.....	9
СПИСОК ЛИТЕРАТУРЫ.....	10

ВВЕДЕНИЕ

Целью данной учебной практики было создание веб-сайта с помощью генератора статических сайтов Hugo, используя для управления версиями - git, для написания документации – Markdown. В качестве платформы для размещения репозитория был выбран GitHub. Репозиторий размещен по ссылке: <https://github.com/Teocad/practice-2025-1/tree/master>.

Материал для создания сайта был взят с дисциплины «Проектная деятельность», а именно проект «Формирование предпрофессиональной образовательной среды в Московском политехе». Куратор проекта – Володина Марина Николаевна. Главной целью проекта является разработка предпрофессиональной навигации молодежи, направленной на раннюю профориентацию школьников, и привлечение их в качестве абитуриентов в Московский Политех. Задачи проекта: участие в проведении мероприятий, связанных со школами; организация выездных мероприятий в школы-партнеры; разработка мастер-классов для проведения занятий; обеспечение логистики при проведении мероприятий.

В качестве заказчика стратегического проекта по дисциплине «Проектная деятельность» выступает «Московский Политех». Взаимодействие с заказчиком и партнерами происходит через куратора проекта.

В качестве задания по учебной практике необходимо было сделать следующие задачи: настройка Git и репозитория; написание документов в Markdown; создание статического веб-сайта; взаимодействие с организацией-партнером; отчет по практике.

В качестве результата учебной практики был создан статический веб-сайт с помощью генератора Hugo. Ссылка на сайт: <https://pre-profedu.netlify.app/>

1. Изучение Git и GitHub

Система контроля версий (git) — это система, записывающая изменения в файл или набор файлов в течение времени и позволяющая вернуться позже к определённой версии. Чтобы использовать git необходимо скачать и установить его, сделать это можно либо скачав его с официального сайта git, или использовать менеджер пакетов, например, «Chocolatey», если мы используем ОС Windows.

Создать репозиторий git, можно используя команду «git init», эта команда создаст в текущем каталоге новый подкаталог с именем «.git», где будут храниться все необходимые файлы репозитория. Так же создать репозиторий можно с помощью команды «git clone <url>».

Команда «git add» добавляет содержимое рабочего каталога в индекс для последующего коммита «git commit». Узнать какие файлы находятся в индексе, или не добавлены в индекс, можно с помощью команды «git status». Если необходимо создать новую ветку, то это можно сделать, используя команду «git branch название_ветки», но после этого git не переключится на новую ветку автоматически, сделать это можно, используя команду «git checkout -b название_ветки».

Отправить изменения в удаленный репозиторий можно с помощью команды «git push», указав имя репозитория и ветки, а получить изменения из удаленного репозитория можно, используя «git pull». Чтобы отправить новую ветку в удаленный репозиторий нужно использовать «git push -u», указав имя репозитория и ветки.

2. Изучение синтаксиса Markdown

Markdown – это язык разметки, который помогает форматировать текст, с помощью простого синтаксиса, который интуитивно понятен. Данный инструмент, очень популярен у специалистов, так как его очень удобно читать даже в исходном виде. Также он совместим со многими платформами и конвертируется в различные форматы, например HTML, DOCX, PDF.

Markdown позволяет создавать заголовки, используя символ #. Уровень заголовка определяется количеством символов #, где # — заголовок первого уровня, ## — второго и далее до шестого уровня.

Новый абзац в Markdown создается просто — нужно оставить пустую строку между блоками текста. Чтобы добавить разрыв строки внутри абзаца, достаточно поставить два пробела в конце строки.

Markdown поддерживает несколько видов выделений, таких как жирный текст, курсив и зачеркнутый текст. Это полезно для акцентов и выделения важной информации. ****Жирный текст****, **Курсивный текст**, ~~~Зачеркнутый текст~~~.

Markdown поддерживает два типа списков: упорядоченные (нумерованные) и неупорядоченные (маркированные). Нумерованные создаются с помощью цифр, а маркированные — символами - или *

Markdown синтаксис позволяет легко добавлять ссылки и изображения в текст. Для создания ссылки используется квадратные и круглые скобки: [текст ссылки](<https://example.com>). Изображения оформляются аналогично ссылкам, но перед текстом ссылки добавляется !.

Таблицы в Markdown создаются с использованием символа |, который разделяет столбцы. Для выравнивания заголовков столбцов применяются двоеточия.

Чтобы вставить код, можно использовать тройные кавычки ``` . Для подсветки синтаксиса укажите язык программирования.

3. Разработка сайта, используя генератор статических сайтов Hugo

Hugo – это генератор статических сайтов, написанный на Go, оптимизированный для быстрой работы и гибкий в настройке. Для использования Hugo необходимо его скачать, сделать это лучше всего используя пакетный менеджер, например Chocolatey, написав в PowerShell команду «choco install hugo-extended». После этой команды установится расширенная версия Hugo.

Установив Hugo, можно приступить к разработке сайта. Первым шагом будет клонирование репозитория practice-2025-1. После этого используя команду «hugo new site site», создаем папку «site», где будет размещена стандартная структура hugo. Основные папки для нас это «theme», где будет размещена тема hugo, «content», где хранится контент сайта, «hugo.toml» - это файл конфигурации, где мы можем задать параметры нашего сайта, «assets» - здесь можно хранить изображения и файлы с кодом.

Для своего проекта я использовал тему «BlowFish». Для этого я перешел в каталог сайта «cd site» и добавил тему как подмодуль «git submodule add -b main <https://github.com/nunocoracao/blowfish.git> themes/blowfish». После этого в папке «themes» была создана папка «blowfish». Чтобы использовать шаблон темы необходимо удалить наш базовый файл конфигурации «hugo.toml», и создать папку «config», после этого скопировать в нее папку «_default» из темы «blowfish» со всеми файлами конфигурации. Используя команду «hugo serve», мы получим результат как на рисунке 1.

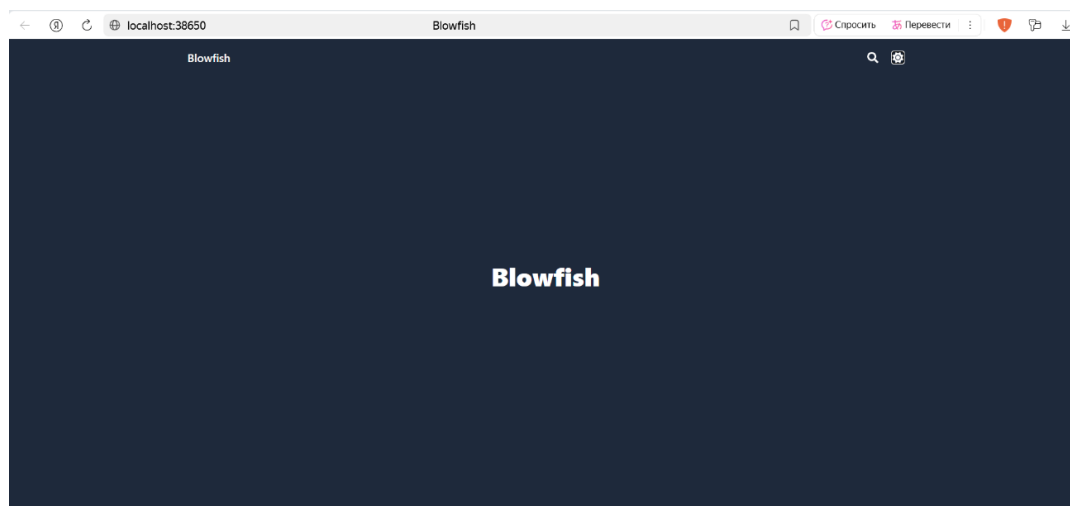


Рисунок 1 – Пустая страница Blowfish

Теперь, когда мы установили тему Blowfish, мы должны изменить файлы конфигурации на необходимые нам параметры. После чего, добавив новые каталоги под наши разделы сайта, и создав в них файлы markdown, мы сможем добавить контент на сайт. Пример статьи, размещенной на сайте, можно увидеть на рисунке 2.

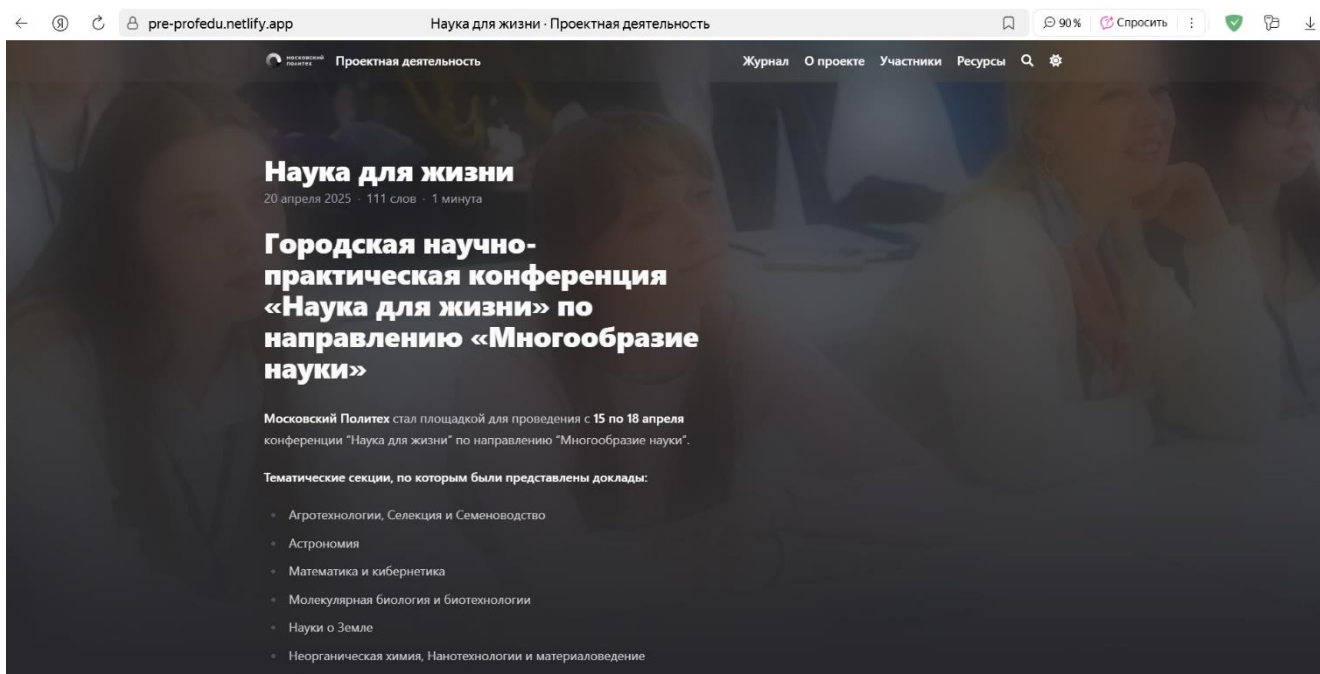


Рисунок 2 - Статья из блога сайта

4. Взаимодействие с организацией-партнером

В рамках проектной деятельности, во время конференции, которая проходила на территории Московского Политеха с 15 по 18 апреля 2025 года, помогал в организации мероприятия, а именно: оказывал помощь в обеспечении логистики участников мероприятия; помощь в подготовке места и настройке оборудования для выступления; взаимодействовал с организаторами мероприятия и участниками.



Рисунок 3 – Благодарность за помощь на конференции

ЗАКЛЮЧЕНИЕ

В рамках текущей учебной практики были успешно выполнены следующие задачи:

1. Изучение базовых команд git, создание репозитория на GitHub.
2. Изучение синтаксиса markdown и использование его для написания контента.
3. Изучение и применение на практике статического генератора сайтов Hugo. Создание сайта по проектной деятельности.
4. Участие в мероприятии в рамках проектной деятельности.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Что такое Markdown и зачем он нужен? [сайт] // Хекслет (Hexlet). — URL: <https://ru.hexlet.io/blog/posts/что-такое-markdown-i-zachem-on-nuzhen> (дата обращения: 09.07.2025).
2. Pro Git [сайт] / S. Chacon, B. Straub; пер. на рус. яз. // Git. — 2-е изд. — URL: <https://git-scm.com/book/ru/v2> (дата обращения: 09.07.2025).
3. Documentation [сайт] // Hugo. — URL: <https://gohugo.io/documentation/> (дата обращения: 09.07.2025).
4. Создание содержимого веб-сайта [сайт] // MDN Web Docs. — URL: https://developer.mozilla.org/ru/docs/Learn_web_development/Getting_started/Your_first_website/Creating_the_content (дата обращения: 09.07.2025).
5. CSS: каскадные таблицы стилей [сайт] // MDN Web Docs. — URL: <https://developer.mozilla.org/ru/docs/Web/CSS> (дата обращения: 09.07.2025).