Отчет по созданию проекта

1 этап проекта

Гибшер Кирилл Владимирович

Содержание

# 1 Цель работы

* Адаптировать виртуальную машину под дальнейшую работу с сайтом путем создания новых каталогов и внедрения генератора статических сайтов hugo.

# 2 Задание

1. Скачать расширение для работы с сайтом.
2. Используя шаблоны с github создать необходимые каталоги и настроить их работу.
3. Проверить работоспособность сайта после завершения всех настроек.

# 3 Теоретическое введение

**Сайт** - одна или несколько логически связанных между собой веб-страниц; также место расположения контента сервера. Обычно сайт в Интернете представляет собой массив связанных данных, имеющий уникальный адрес и воспринимаемый пользователями как единое целое.

# 4 Выполнение лабораторной работы

Устанавливаем архив hugo-extended на нашу ОС и затем распаковываем этот архив. (рис. [[1](#fig:001)]).

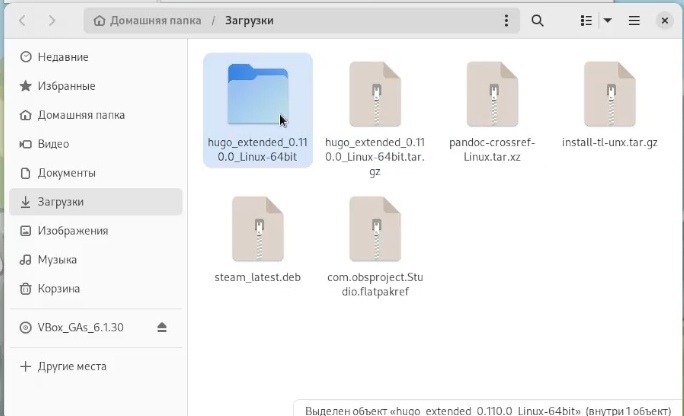


Figure 1: Установка hugo-extended

Создаем в домашнем каталоге пользователя папку bin. Из исходного распакованного архива hugo-extended забираем только лишь исполняемый файл hugo и перемещаем в созданную папку bin. (рис. [[2](#fig:002)]).

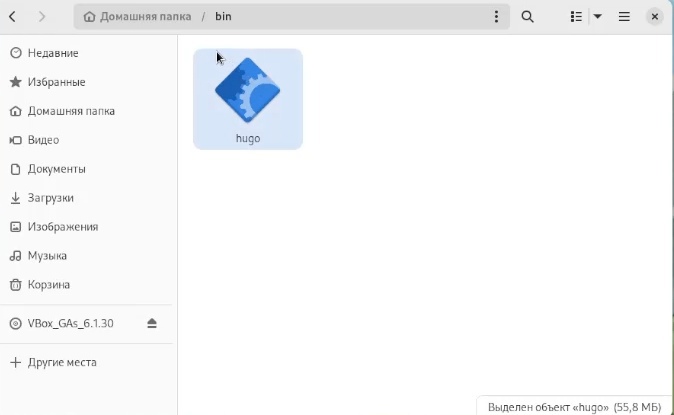


Figure 2: Перемещение исполняемого файла hugo

Используя шаблон с github создаем новый репозиторий blog. (рис. [[3](#fig:003)]).

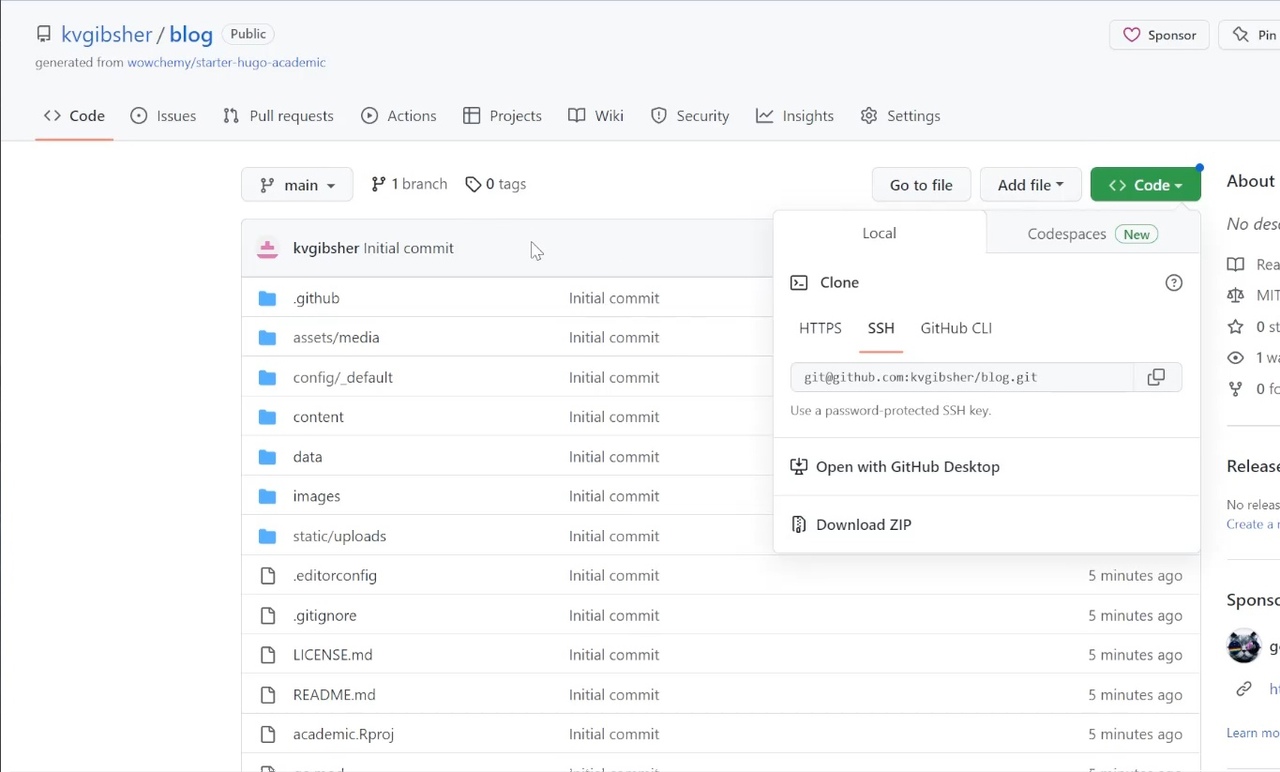


Figure 3: Создание нового репозитория

С помощью команды git clone клонируем репозиторий blog в нашу ОС с последующим созданием каталога соответственно. (рис. [[4](#fig:004)]).

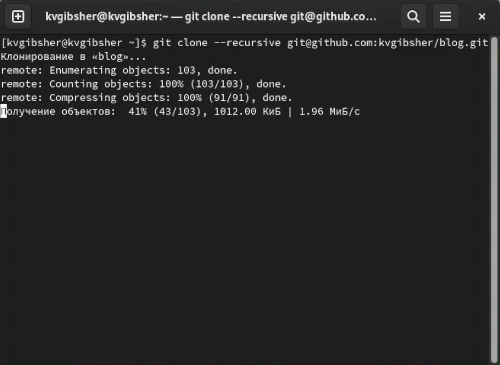


Figure 4: Клонирование репозитория

Проверяем успешность загрузки файлов с github с помощью команды ls (рис. [[5](#fig:005)]).

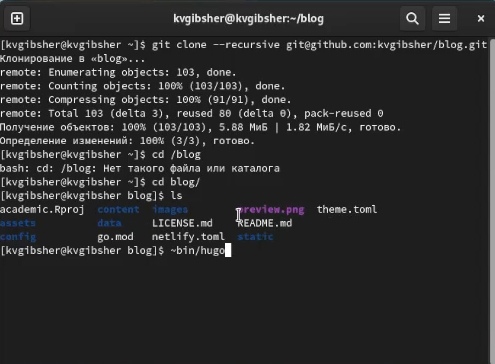


Figure 5: Проверка файлов

Запускаем исполняемый файл hugo раннее перемещенный в созданную нами в домашнем каталоге пользователя папку bin. (рис. [[6](#fig:006)]).

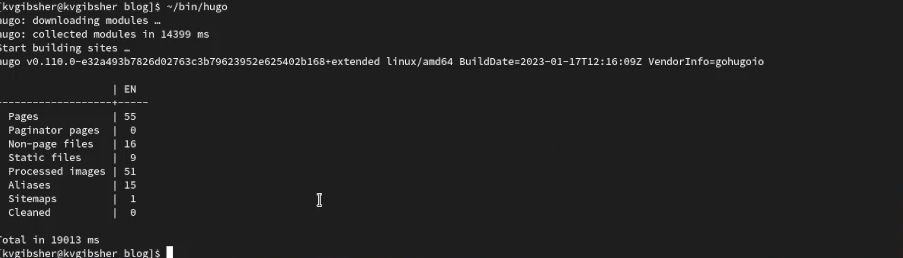


Figure 6: Запуск hugo

Удаляем каталог public находящийся в blog ввиду его ненадобности. (рис. [[7](#fig:007)]).

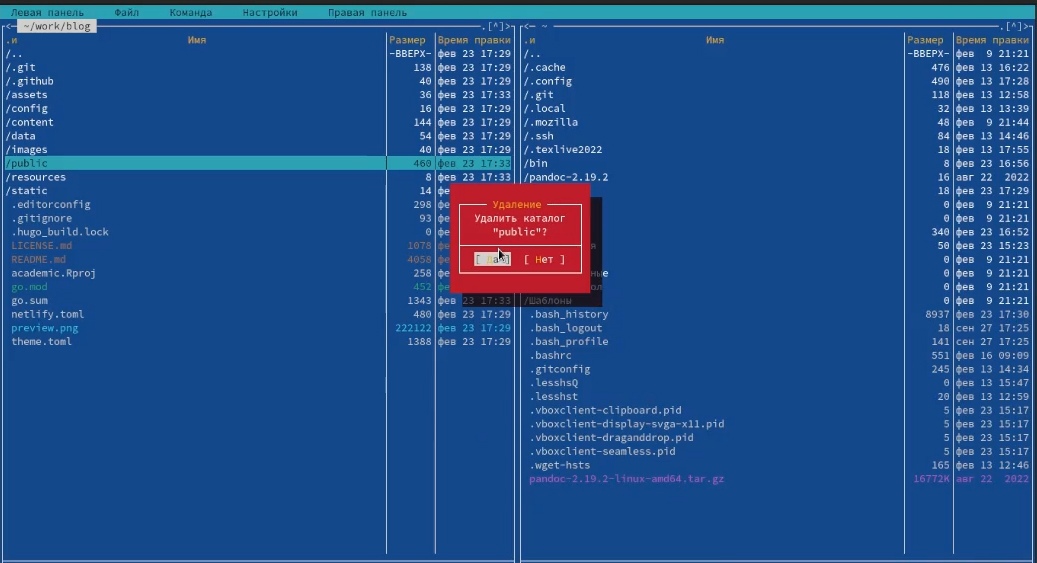


Figure 7: Удаление каталога public

Находясь в каталоге blog прописываем команду hugo server , которая создает локальный хост на сайт и выдает ссылку рабочую лишь в той ОС и на там устройстве, в котором произошла ее генерация (рис. [[8](#fig:008)]).

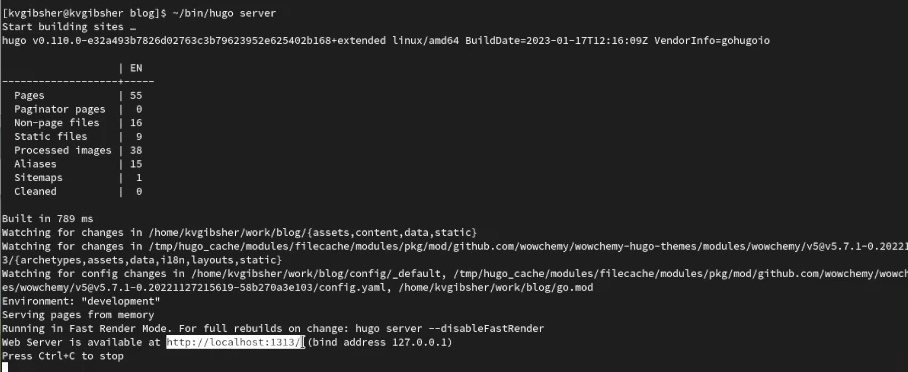


Figure 8: Hugo server

Проверяем работоспособность ссылки и переходим по ней. (рис. [[9](#fig:009)]).

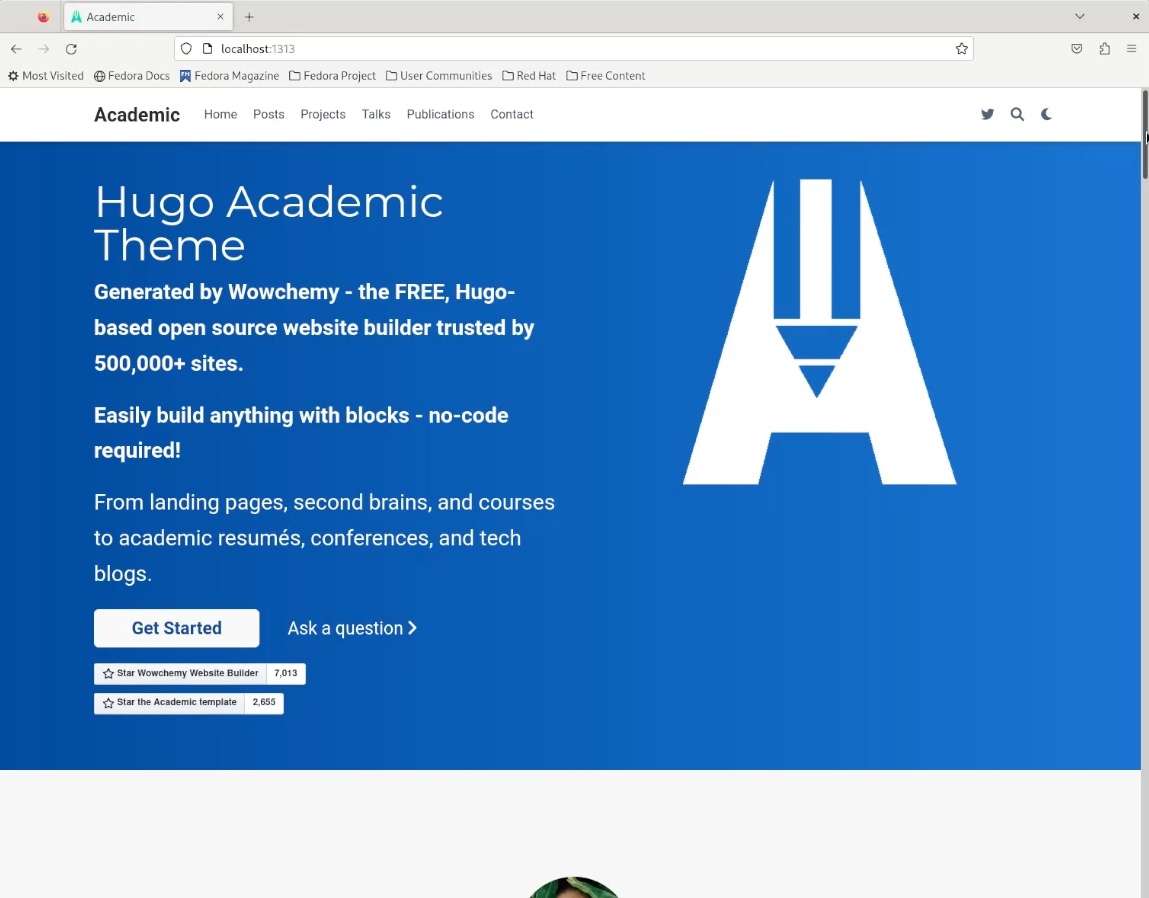


Figure 9: Переход на сайт по сгенерированной ссылке

Создание нового репозитория на github , который будет иметь прямую связь с нашим будущим сайтом. (рис. [[10](#fig:010)]).

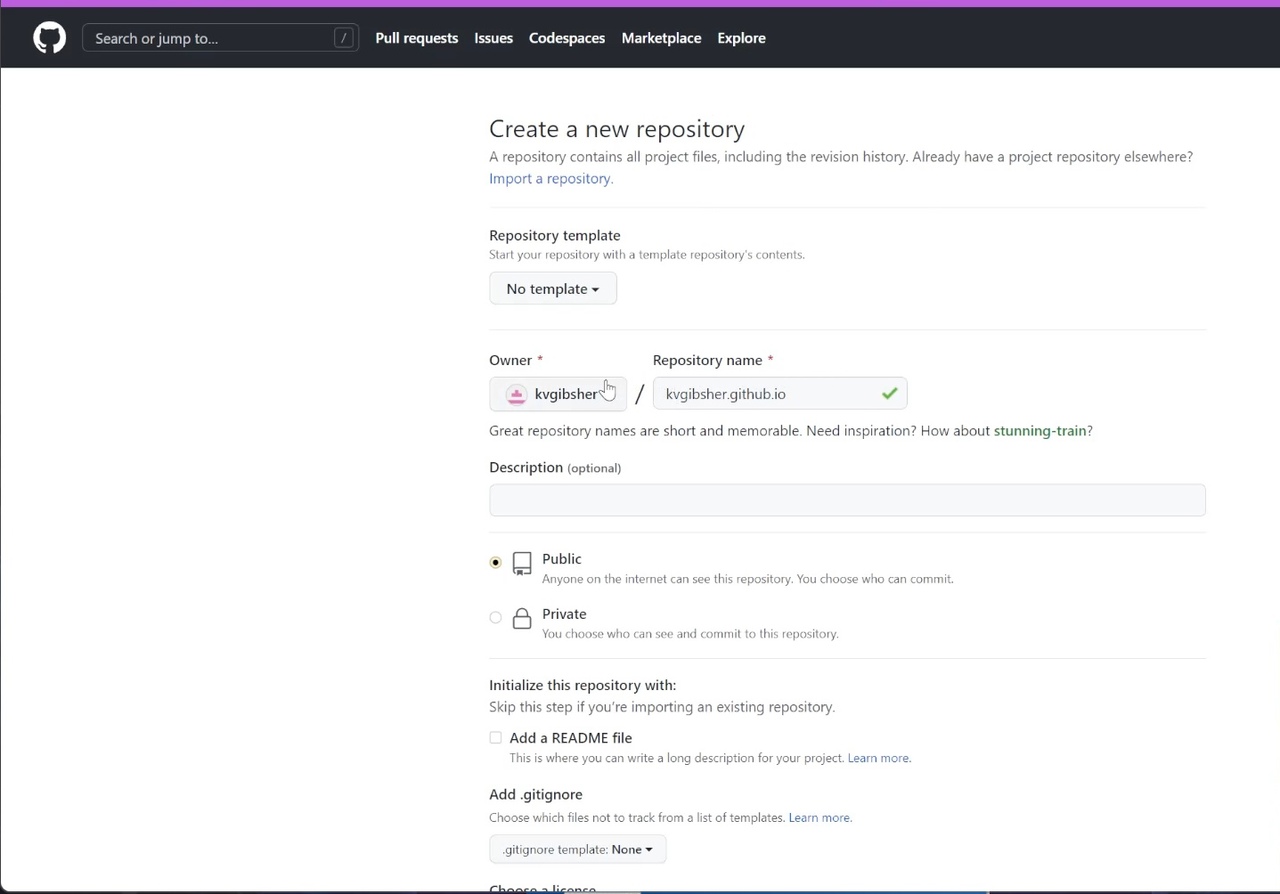


Figure 10: Создание репозитория github.io

Клонируем только что созданный нами репозиторий в нашу ОС и с помощью команды ls проверяем, что все успешно получилось. Каталог склонировался, но пустой.(рис. [[11](#fig:011)]).

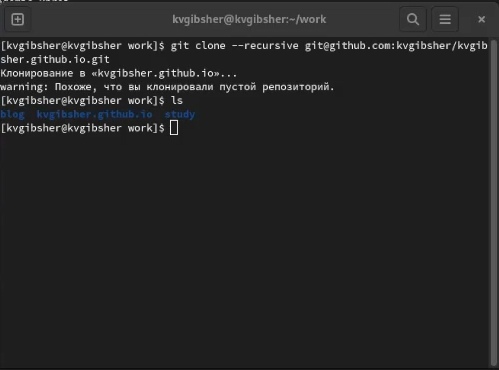


Figure 11: Клонирование репозитория github.io

Создаем главную ветку в каталоге и называем ее main. Затем, чтобы активировать репозиторий создаем пустой файл README.md и прописываем git push origin main для выгрузки на github. (рис. [[12](#fig:012)]).

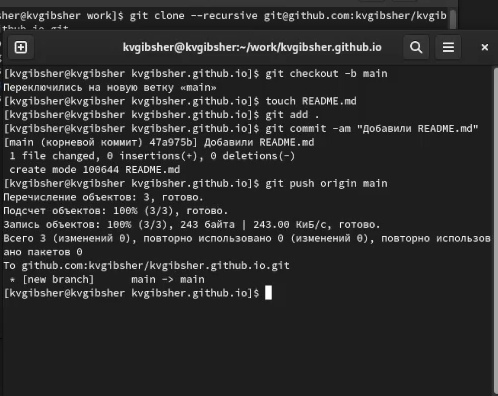


Figure 12: Активация репозитория

Прописываем git submodule add -b main находясь в каталоге blog. Но выполнение этой команды не очень успешно, т.к. некоторые пути игнорируются файлом gitignore. Потому попытка добавить submodule public проваливается. (рис. [[13](#fig:013)]).

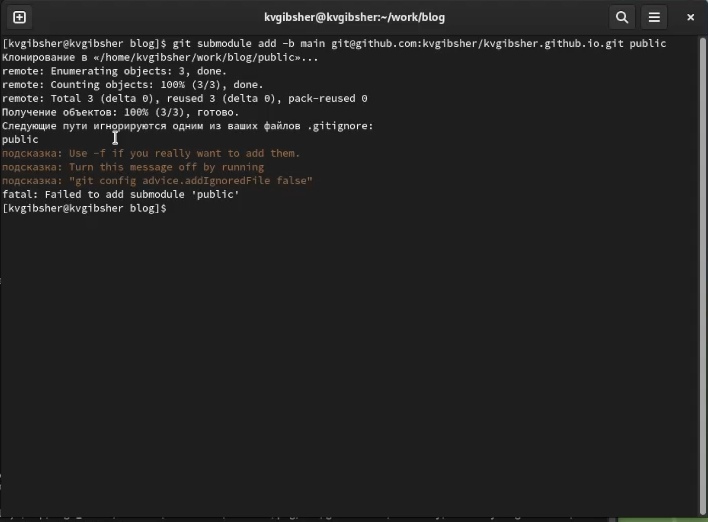


Figure 13: Git add submodule public

С помощью MightCommander переходим в файл gitignore и коммитим с помощью знака # необходимую строку. Затем повторяем команду из прошлого пункта и на этот раз все получается и автоматически у нас создается каталог public в blog. (рис. [[14](#fig:014)]).



Figure 14: Исправление файла gitignore и повтор команды

Запускаем вновь исполняемый файл hugo в каталоге blog и все содержимое автоматически выгружается в каталог public. (рис. [[15](#fig:015)]).

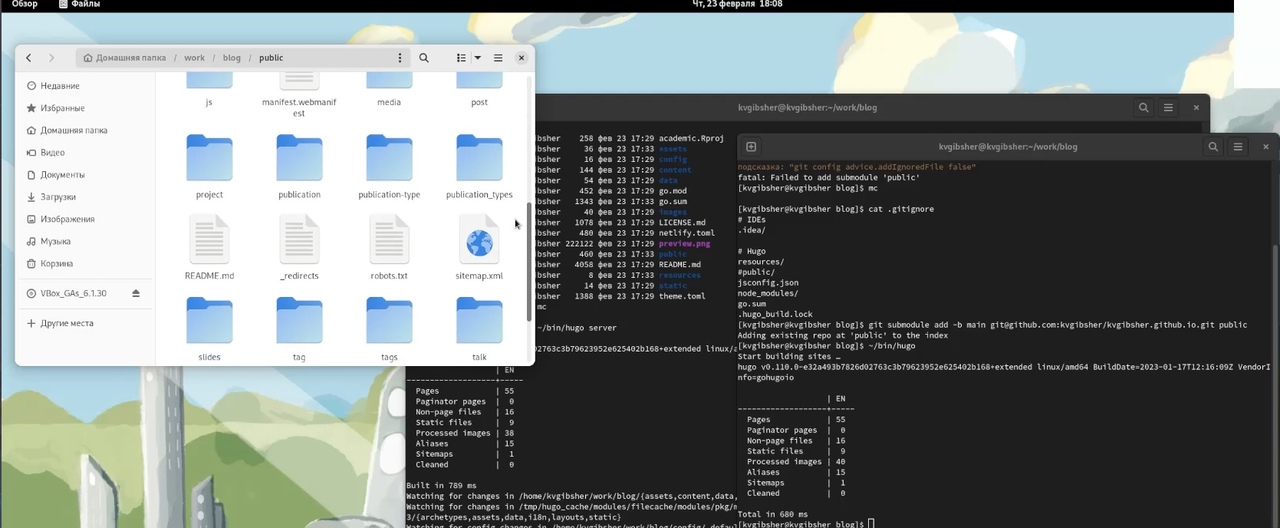


Figure 15: Запуск файла hugo в blog

С помощью команды git remote -v проверяем точно ли необходимый для сайта репозиторий подключен к нашей ОС. Затем с помощью уже знакомых команд добавляем сайт и все сопутствующие файлы на гитхаб. (рис. [[16](#fig:016)]).

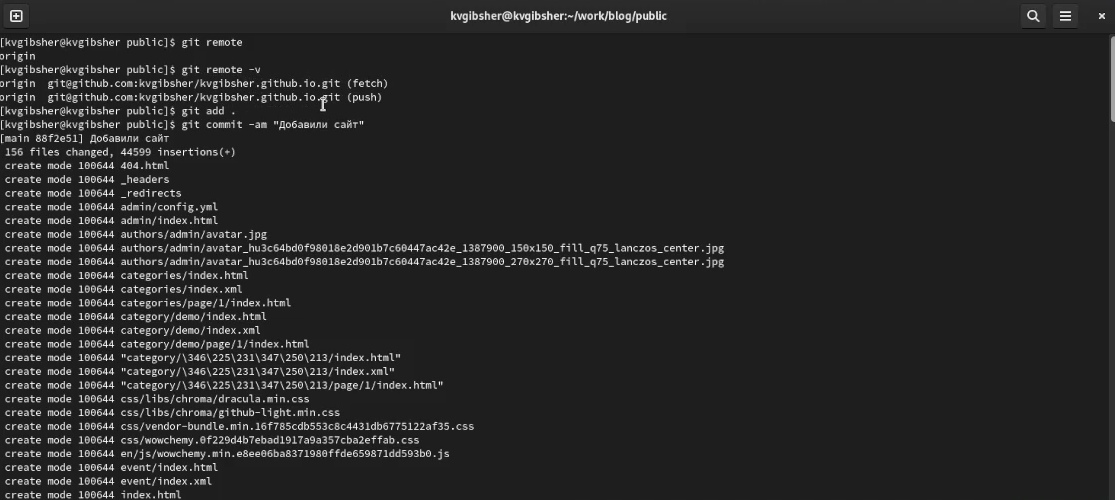


Figure 16: Выгрузка сайта на github

Заходим на гитхаб и убеждаемся в том, что все файлы сайта успешно загружены. (рис. [[17](#fig:017)]).

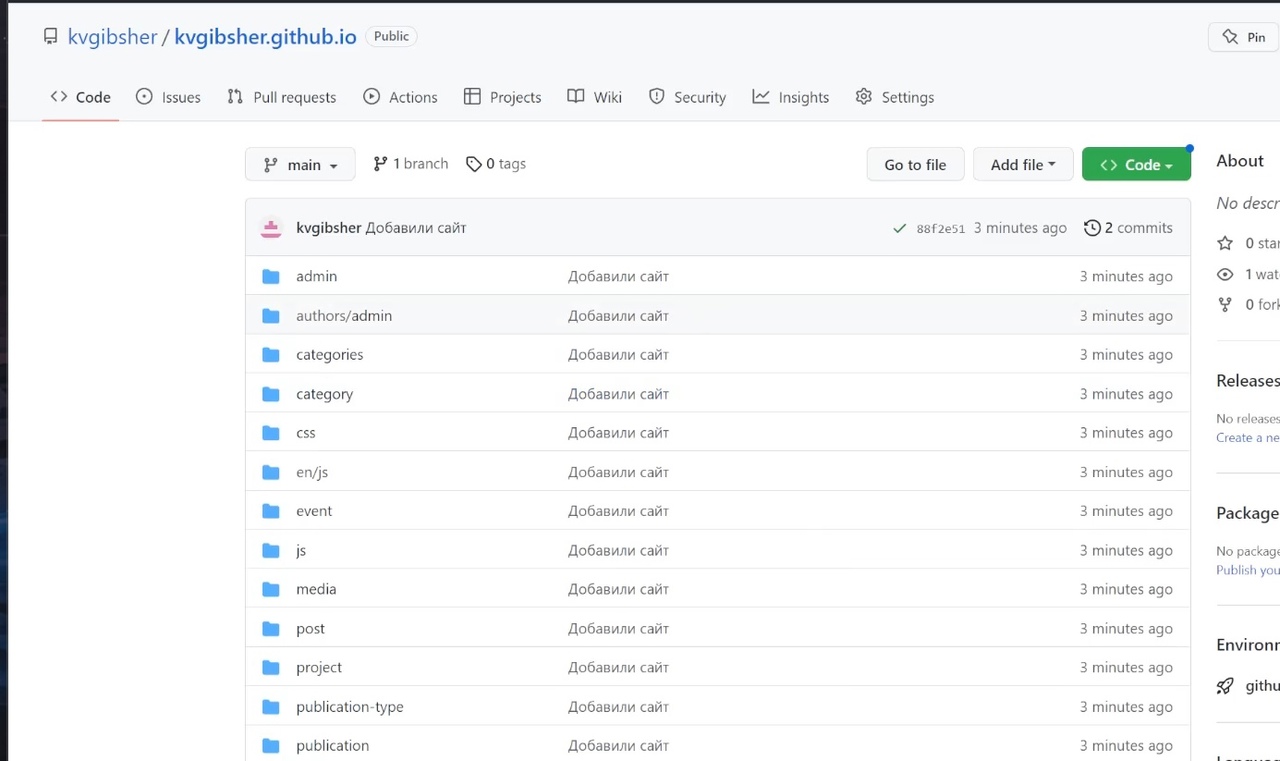


Figure 17: Проверка загрузки

Копируем часть ссылки на одноименный репозиторий и проверяем работоспособность нашего личного сайта. (рис. [[18](#fig:018)]).

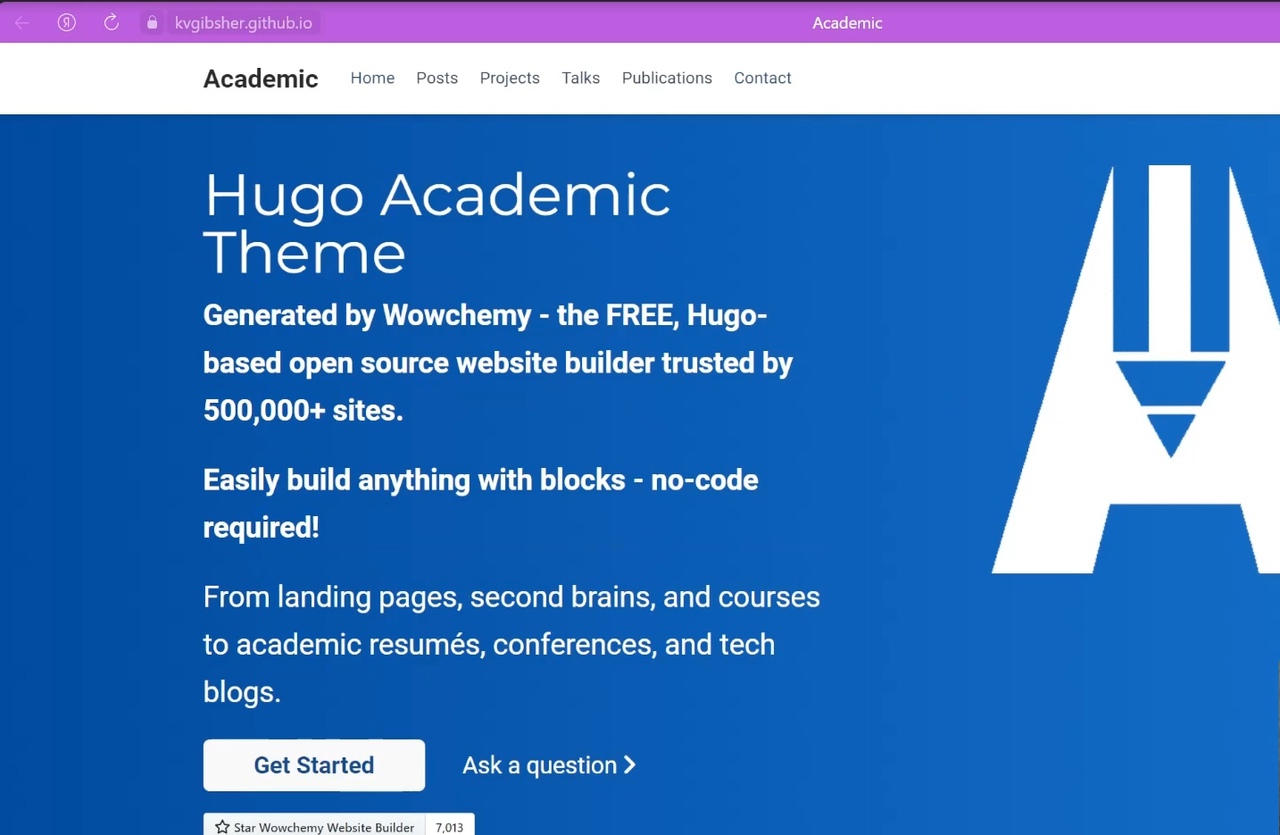


Figure 18: Проверка работоспособности сайта

# 5 Выводы

Мы установили генератор статичных сайтов hugo , создали необходимые для сайта репозитории , связали их с нашей ОС и проверили работоспособность первоначальной версии нашего сайта.

# Список литературы

1. Первый этап индивидуального проекта by Кулябов Д.С. [Электронный ресурс] - Режим доступа:https://www.youtube.com/watch?v=OpsSv0RE3C4