Отчет по персональному проекту версии 1

Генералов Даниил, НПИ-01-21, 1032212280

Содержание

# 1 Цель работы

Целью данной работы является установка генератора статических сайтов Hugo и развертка шаблона сайта на GitHub Pages.

# 2 Задание

Для выполнения работы требуется установить программу Hugo, создать репозиторий для сайта на GitHub Pages, затем провести развертку шаблона сайта на GitHub Pages.

# 3 Теоретическое введение

Один из удобных способов создания статических сайтов (то есть тех, которые в основном состоят из неизменного контента) – это применить генератор статических сайтов. Это программа, которая принимает на вход документы, написанные на промежуточном языке разметки (например Markdown), и выдают набор HTML-страниц, которые затем можно разместить на хостинг.

Один из таких хостингов – GitHub Pages. С помощью инфраструктуры CI можно сделать так, чтобы содержимое вашего репозитория публиковалось в интернете на определенном доменном имени – без использования своей инфраструктуры и не требуя никакой настройки серверов.

# 4 Выполнение лабораторной работы

Первым делом нужно создать новый репозиторий, внутри которого будут храниться данные сайта. Это можно сделать из командной строки или с помощью веб-интерфейса на рис. 1.

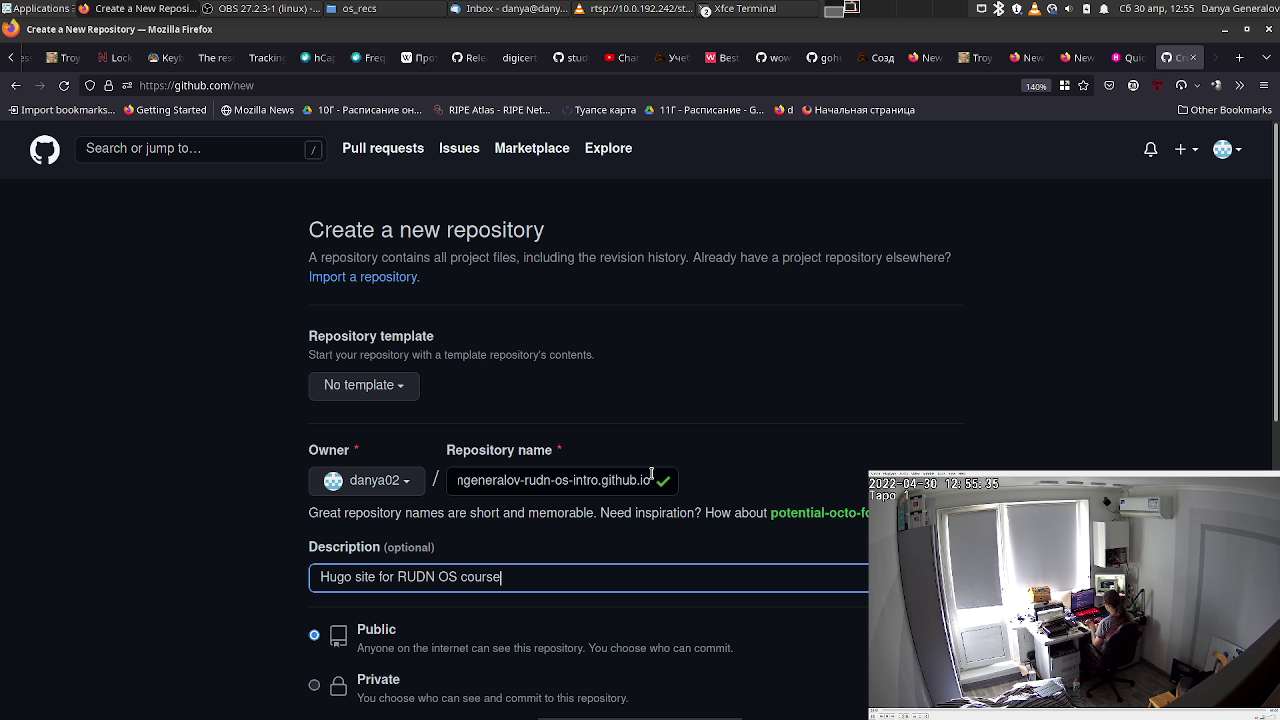


Рис. 1: Создание репозитория в GitHub

Дальше я добавил этот репозиторий в основной репозиторий как подмодуль и использовал команду hugo new site, чтобы создать шаблон сайта, как показано на рис. 2.

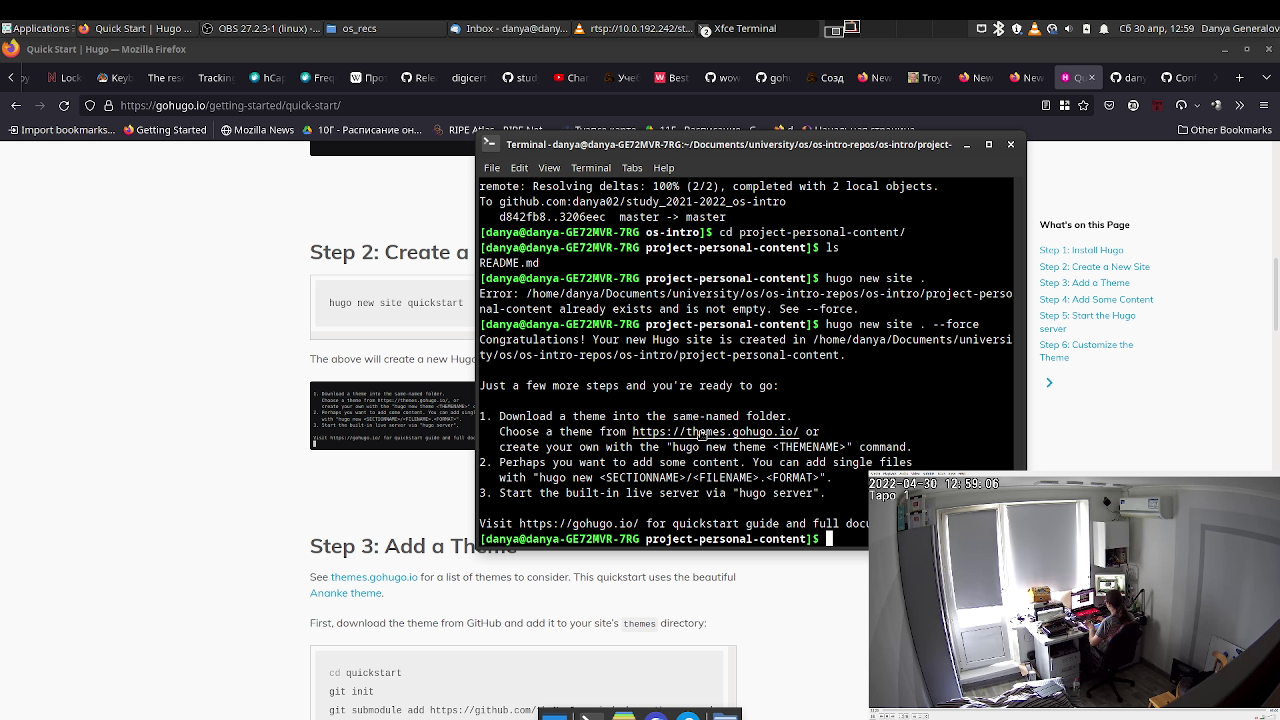


Рис. 2: Создание шаблона сайта

После этого я начал искать, как установить тему Academic, и посредством этого поиска узнал, что она теперь называется Wowchemy, и метод установки изменился. Я узнал это на форуме Wowchemy на GitHub, который показан на рис. 3.

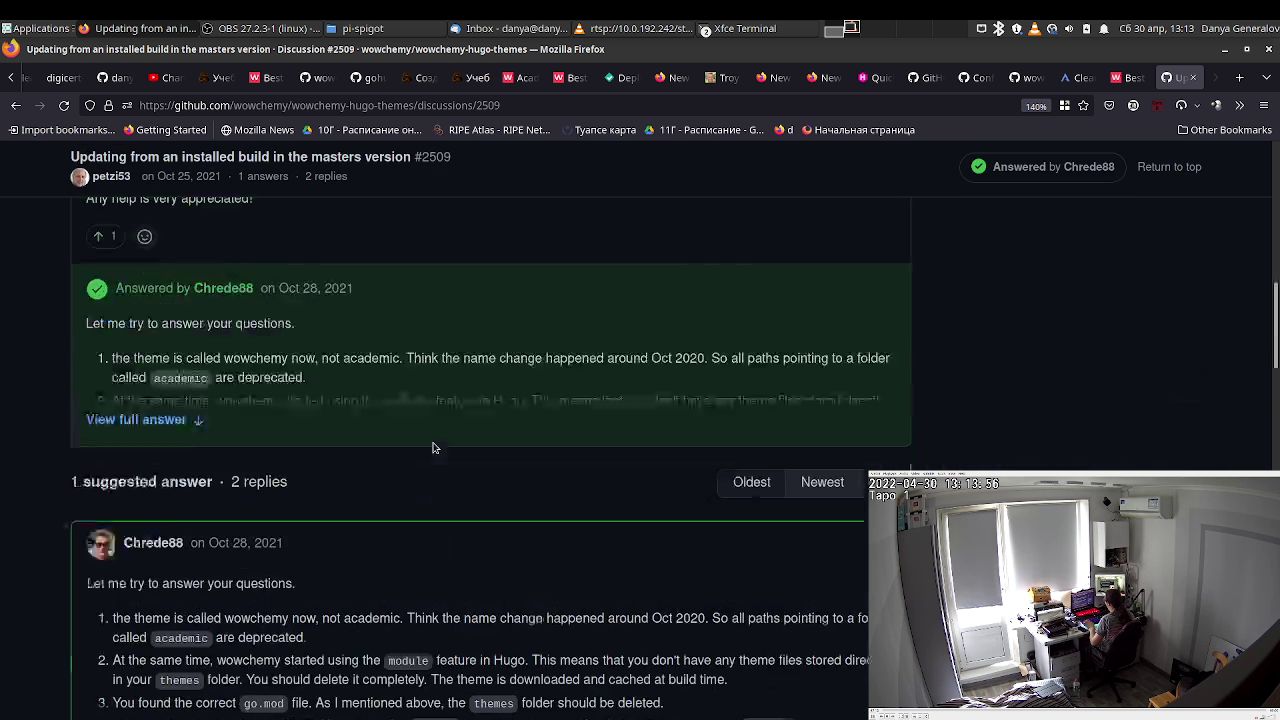


Рис. 3: Форум Wowchemy

Как оказалось, нужно включить его как зависимость в определенном файле в корне репозитория, и после этого шаблон темы был успешно собран, как показано на рис. 4.

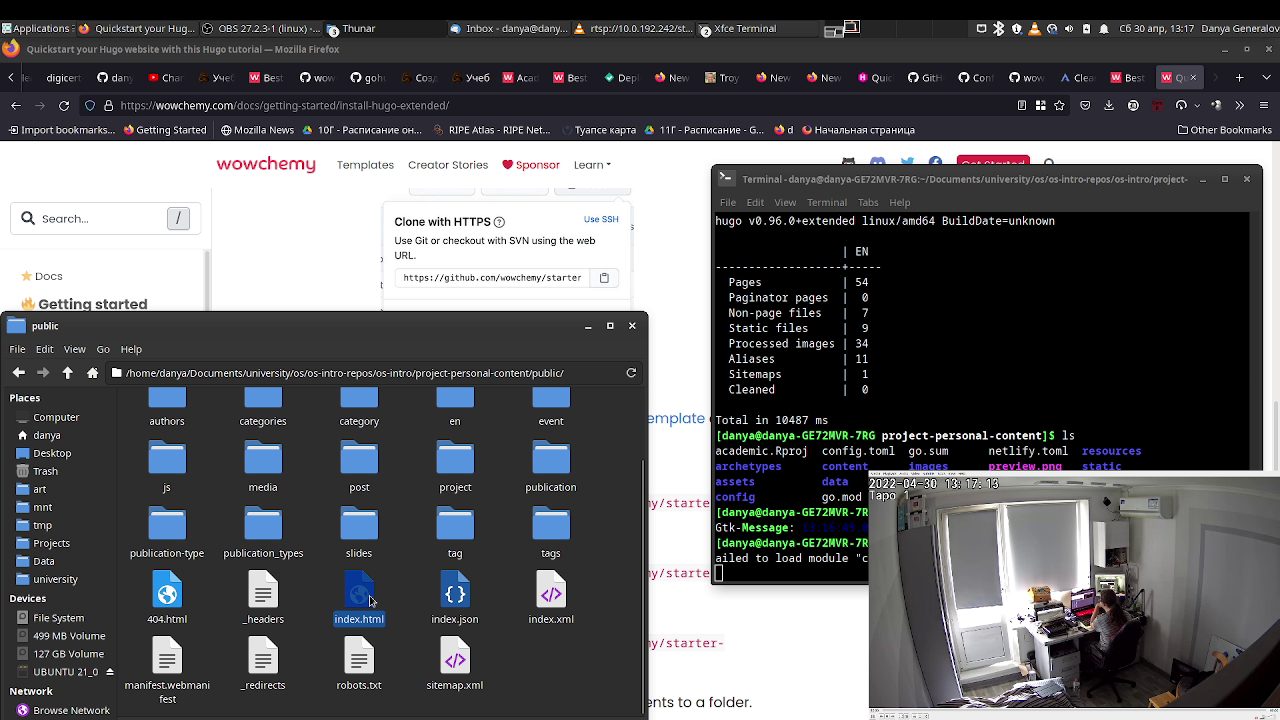


Рис. 4: Компиляция шаблона сайта

После этого была задача опубликовать это на GitHub Pages. Есть две опции, где хранятся файлы, подлежащие публикации – в корне репозитория или в папке docs. По умолчанию Hugo собирает сайт в папку public, поэтому нужно добавить ключ конфигурации, меняющий это поведение – он вносится в файл config.toml в корне репозитория, как показано на рис. 5.

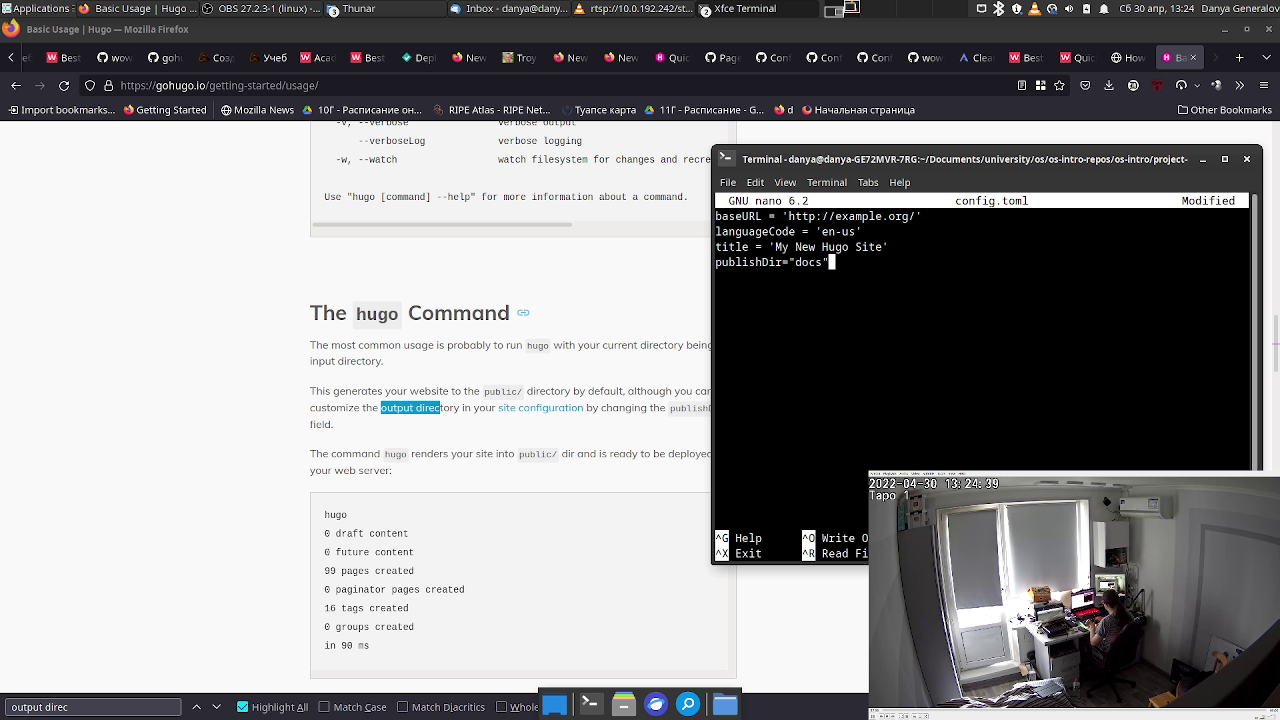


Рис. 5: Конфигурация директории сборки

После этого можно сделать коммит с новым сайтом (рис. 6) и посмотреть на него на GitHub Pages (рис. 7). Но сразу заметно, что что-то не так.

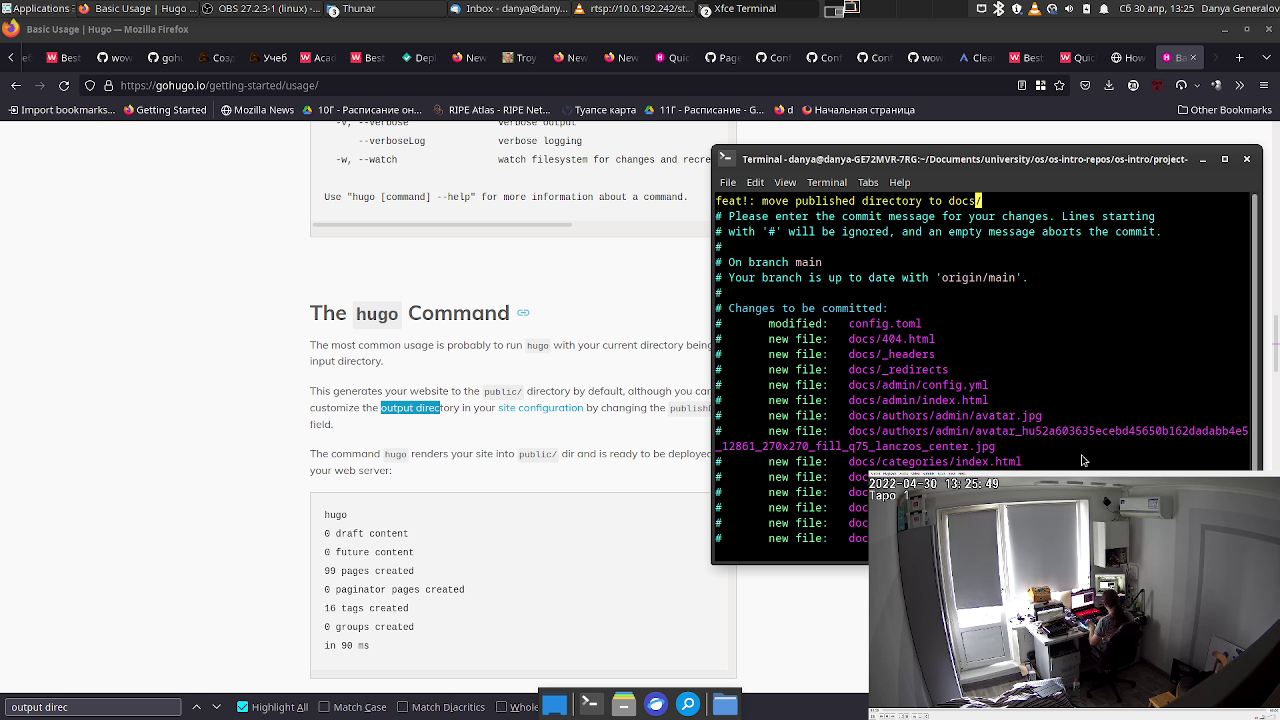


Рис. 6: Коммит с содержимым сайта

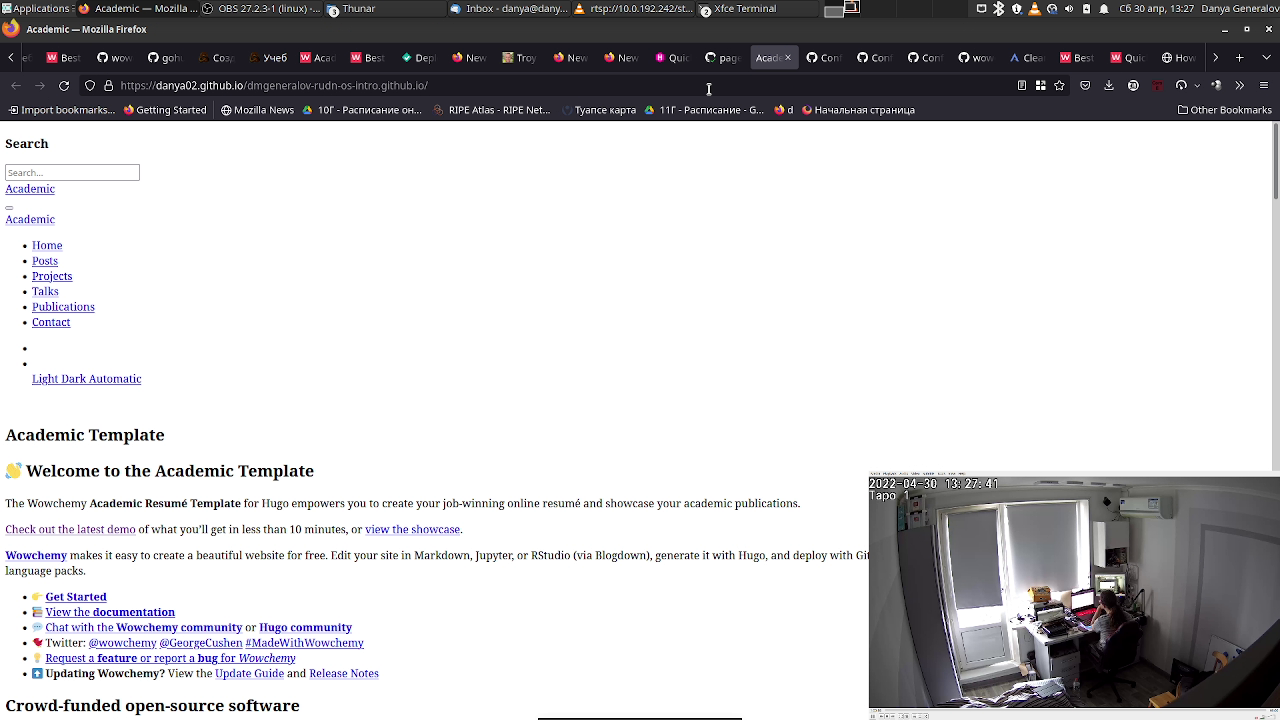


Рис. 7: Сайт на GitHub Pages без форматирования

Как нетрудно увидеть, на странице полностью нет форматирования. Если посмотреть в консоли запросов на рис. 8, то можно увидеть проблему – все ресурсы сайта выходят с ошибкой 404, хотя в папке сборки присутствуют. Дело в том, что пути к ресурсам начинаются с корня сайта, а GitHub Pages публикует проект в субдиректории, что ломает адресацию.

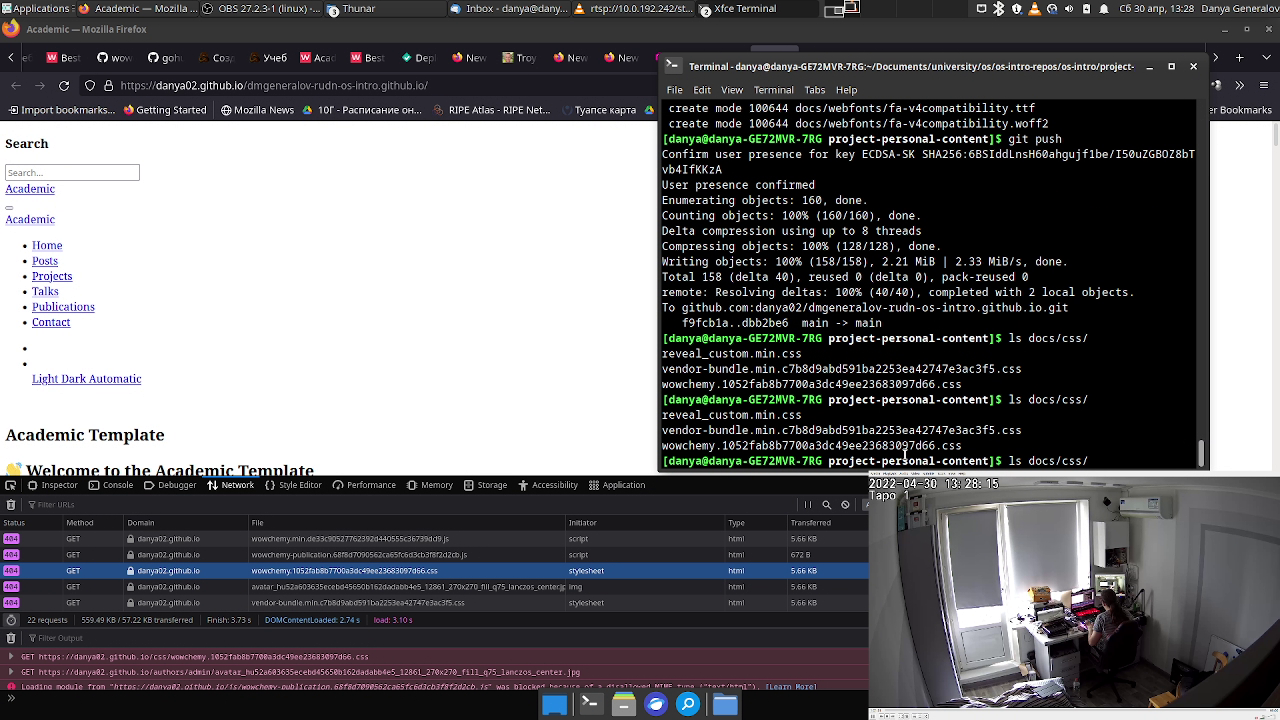


Рис. 8: Ошибка при загрузке CSS

Причина того, что GitHub Pages делает так, заключается в том, что страницы проектов являются подстраницами пользователей или организаций. Я не хотел использовать свою страницу пользователя только для этого проекта, поэтому я вместо этого подумал создать организацию. К сожалению, страница “создать организацию” делает впечатление, будто для того, чтобы это сделать, необходимо заплатить – когда я делал работу, я полностью пропустил бесплатную опцию на странице на рис. 9.

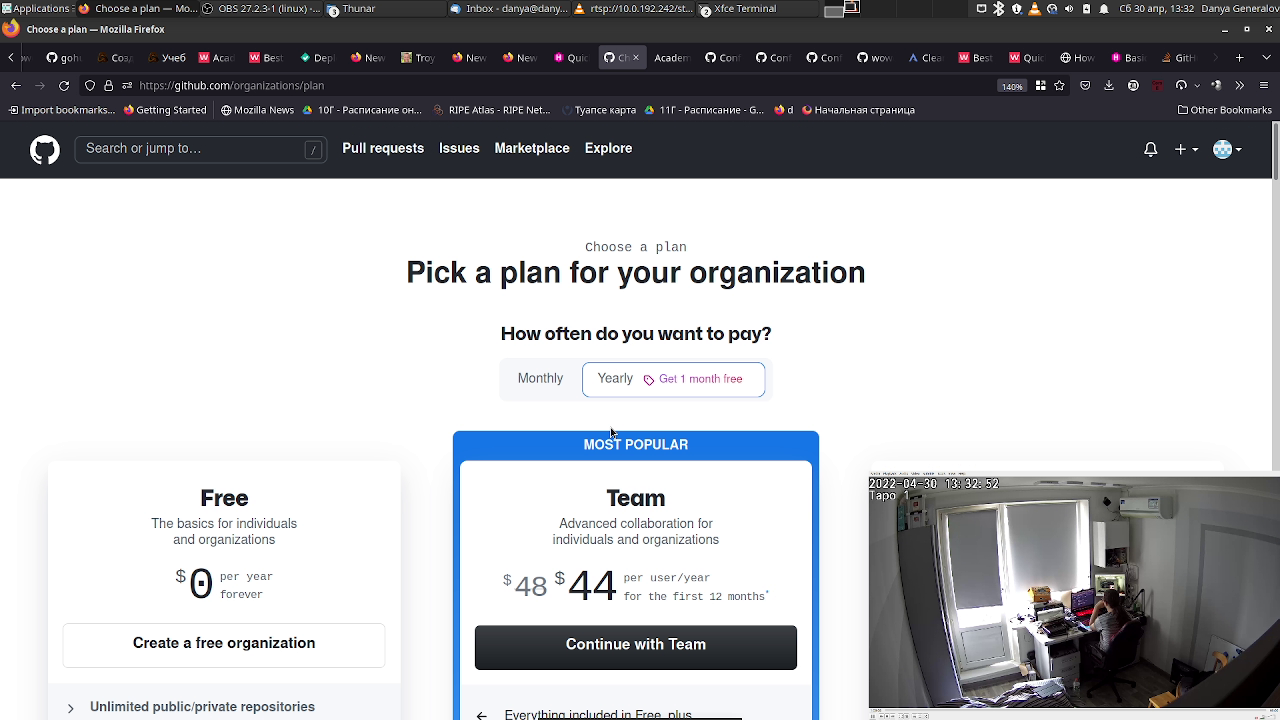


Рис. 9: Страница создания организации

Вместо этого я решил пойти другим путём. У меня есть доменное имя, danya02.ru, и я могу направить один из поддоменов на GitHub Pages. Первый шаг для этого – это прописать DNS-записи в моей записи, чтобы этот поддомен оказался перенаправленным на GitHub Pages. Я добавил для начала A-записи, как показано на рис. 10.

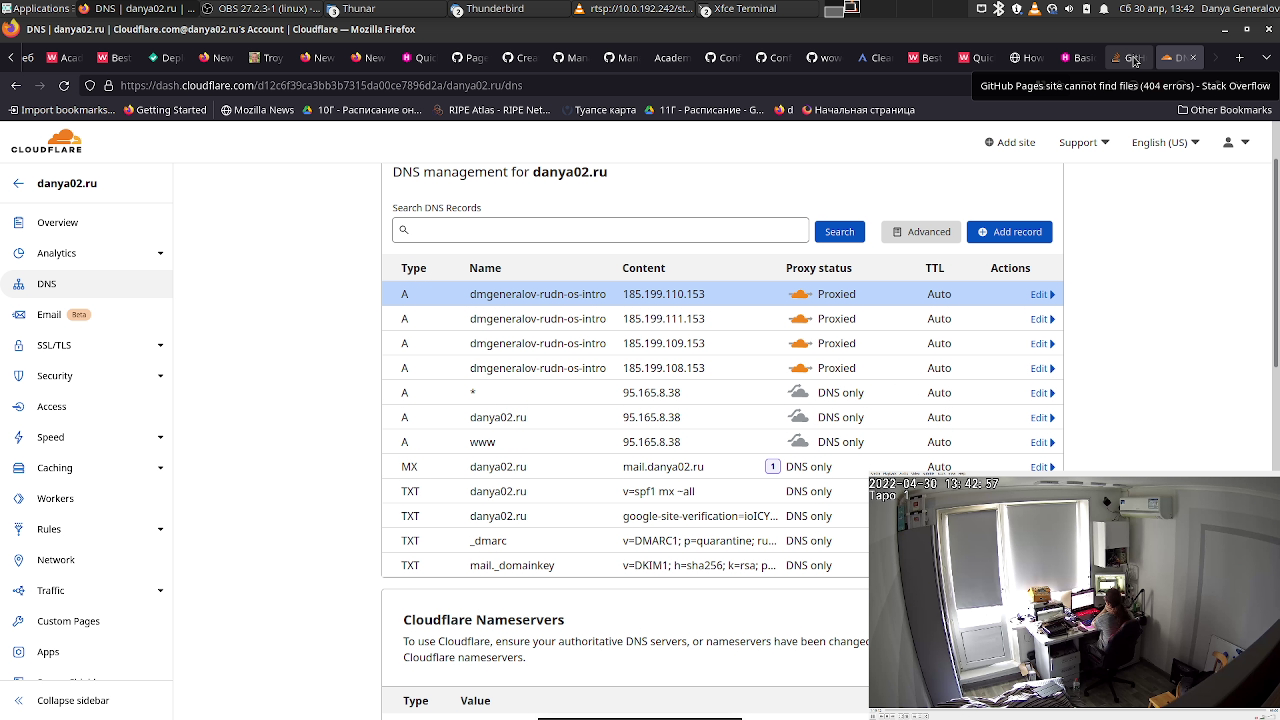


Рис. 10: Доменные записи A

К сожалению, сделав это, я сделал невозможным для GitHub получить TLS-сертификат на этот домен, и поэтому при открытии этого поддомена появлялось окно, говорящее о неправильном сертификате (рис. 11).

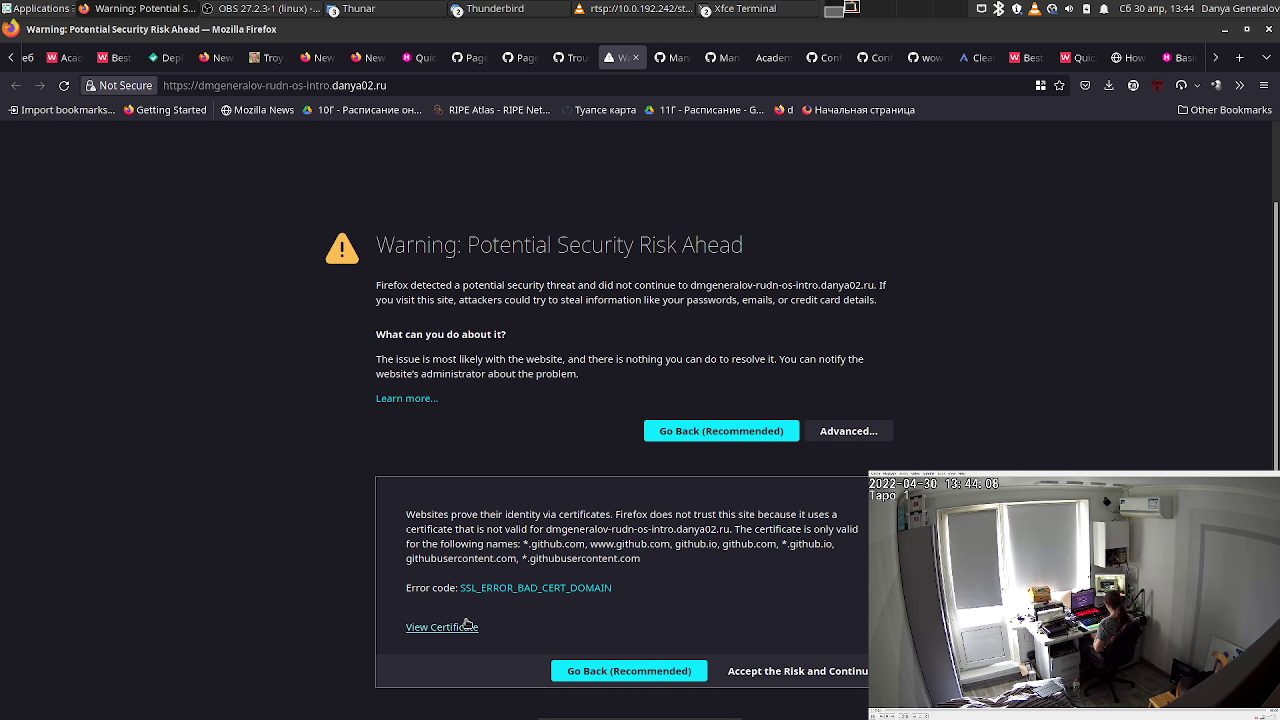


Рис. 11: Ошибка сертификата для домена

Чтобы справиться с этим, как оказалось, нужно указывать не простые A-записи, а CNAME-записи, фактически осуществляющие перенаправление с одного домена на другой. Я добавил CNAME-запись, указывающую на свой поддомен на GitHub Pages, как показано на рис. 12.

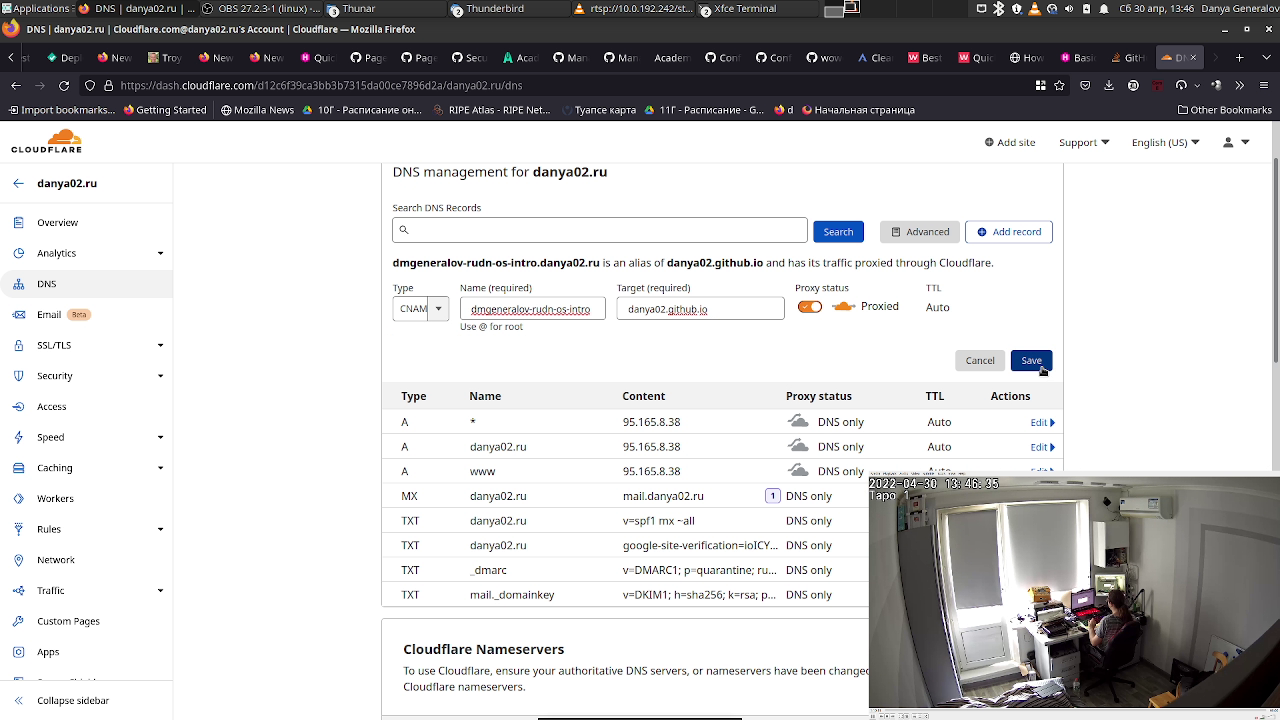


Рис. 12: Создание доменной записи CNAME

После этого интерфейс GitHub Pages подтвердил, что процесс создания сертификата начат (рис. 13), и после нескольких минут сайт стал открываться без предупреждений (рис. 14).

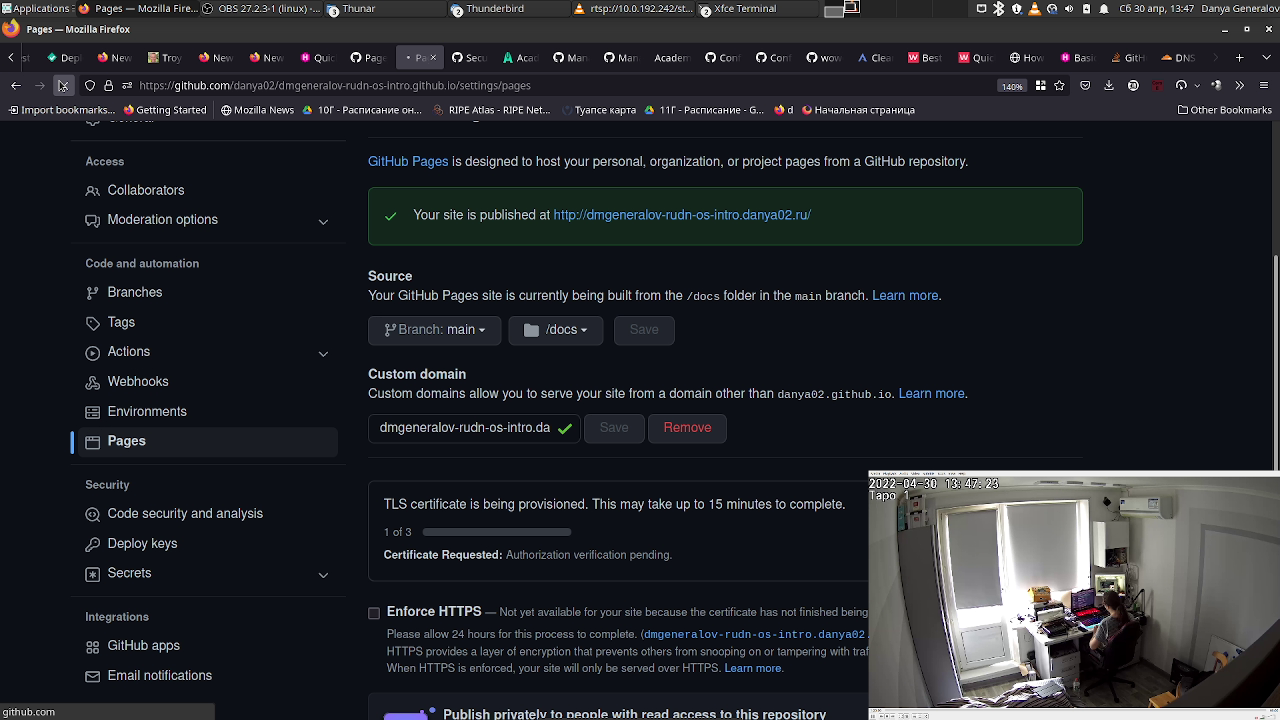


Рис. 13: Создание сертификата

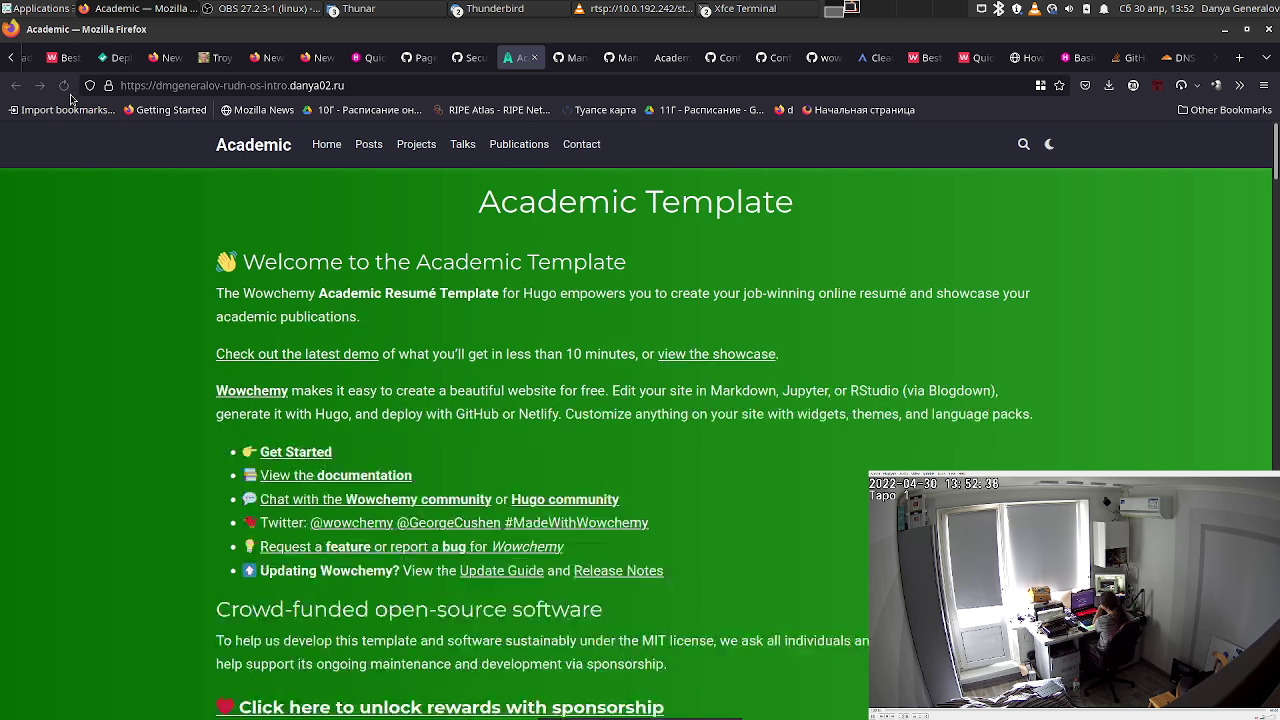


Рис. 14: Теперь сайт работает

Текущее состояние сайта можно увидеть по адресу https://dmgeneralov-rudn-os-intro.danya02.ru/.

# 5 Выводы

В этой работе мы успешно установили Hugo, сделали шаблон сайта, и опубликовали его на GitHub Pages. Используя эти инструменты, можно быстро создавать сайты без написания кода HTML или развертки собственных серверов. Поэтому такие сайты идеально подходят для проектов, которые не подразумевают высокой сложности пользовательского взаимодействия и в основном служат для публикации контента. В следующих шагах этого проекта мы сможем добавить контента к этому сайту.