Отчёт по выполнению первого этапа индивидуального проекта

Персональный сайт научного работника

Дарья Эдуардовна Ибатулина

Содержание

# 1 Цель работы

Размещение на Github pages заготовки для персонального сайта.

# 2 Задание

1. Установить необходимое программное обеспечение.
2. Скачать шаблон темы сайта.
3. Разместить его на хостинге git.
4. Установить параметр для URLs сайта.
5. Разместить заготовку сайта на Github pages.

# 3 Теоретическое введение

Для сайта не всегда нужна работа с базами данных, запросы к API и сложный бэкенд. Во многих случаях достаточно обычного HTML-документа, но красивого и с единым стилем для всех страниц. Разумеется, можно сверстать что-то подходящее самостоятельно, но можно воспользоваться инструментом, который получает на вход текст, картинки и другой контент и генерирует из этого страницу. Такой подход позволяет сэкономить время и получить результат в сжатые сроки. К тому же, результат будет высокого качества.

В Сети есть множество генераторов, различающихся по различным параметрам. Я буду использовать генератор Hugo, обладающий рядом преимуществ [1] :

* Hugo написан на Go, что обеспечивает высокую скорость работы. Разработчики заявляют о скорости генерации менее 1 мс на страницу, а средний веб-сайт генерируется за 1 секунду;
* включены расширенные возможности языка разметки Markdown, позволяющие встраивать новые типы контента;
* встроенные шаблоны для работы с SEO, оптимизацией и аналитикой;
* открытый исходный код проекта;
* большая библиотека готовых тем, которые можно брать за основу для собственных проектов;
* строенные средства адаптации сайтов к другим языкам (i18n).

Возможностями генератора Hugo пользуются не только обычные пользователи, но и большие компании. К примеру, с помощью Hugo сделана страница поддержки сервиса 1Password, портал с курсами Godot Tutorials и документация Linode.

# 4 Выполнение лабораторной работы

Начнём с установки необходимого ПО. Для этого переходим в репозиторий на Github и скачиваем нужный архив (Linux-64bit, hugo extended) (рис. [1](#fig:001)):

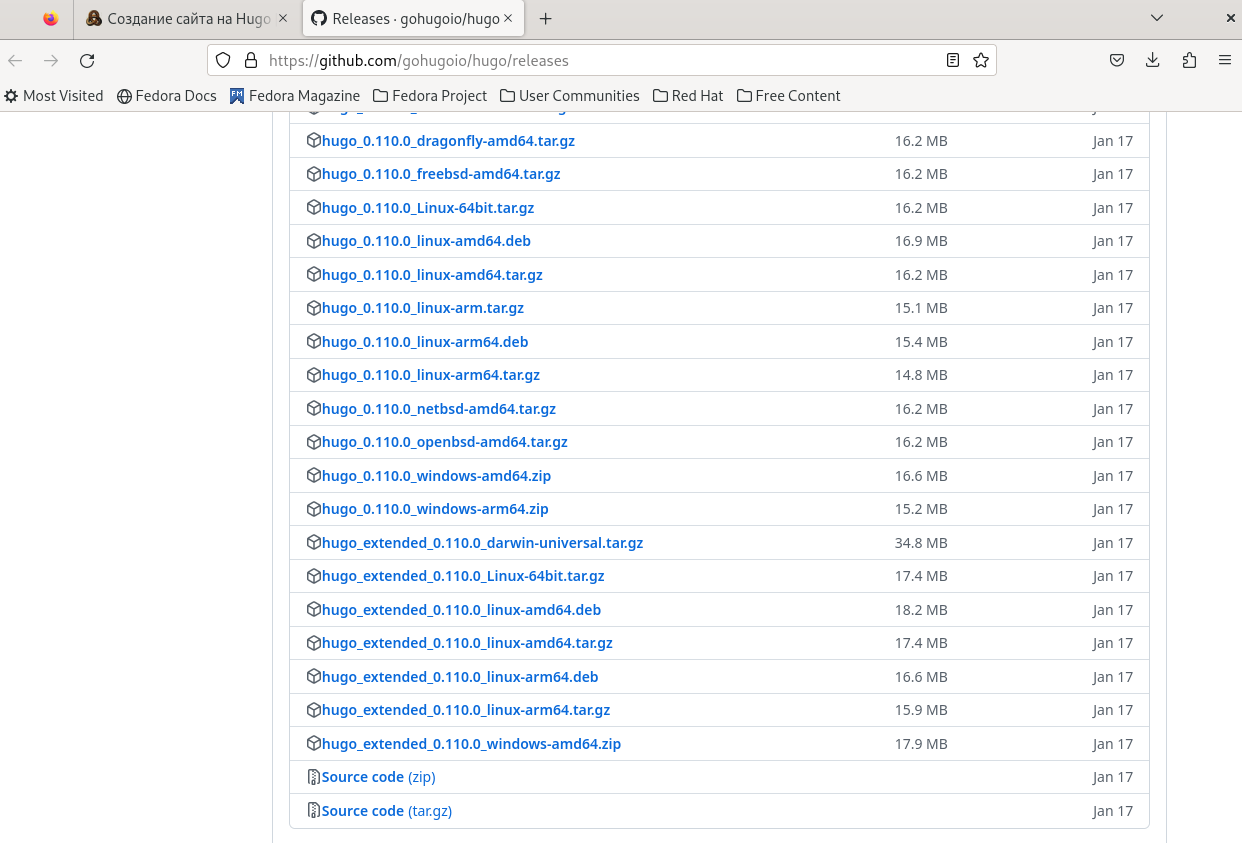


Figure 1: Установка необходимого ПО

Далее в домашне папке мы создаём папку bin, в которую положим исполняемый файл hugo, скачанный в предыдущем пункте (рис. [2](#fig:002)):

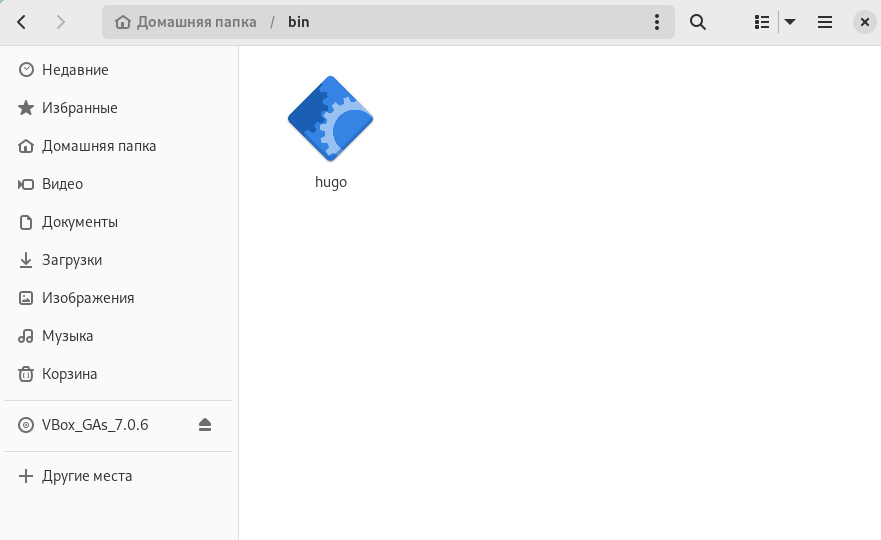


Figure 2: Помещение ПО в нужную папку

Затем переходим на github.com и создаём репозиторий blog (рис. [3](#fig:003)):

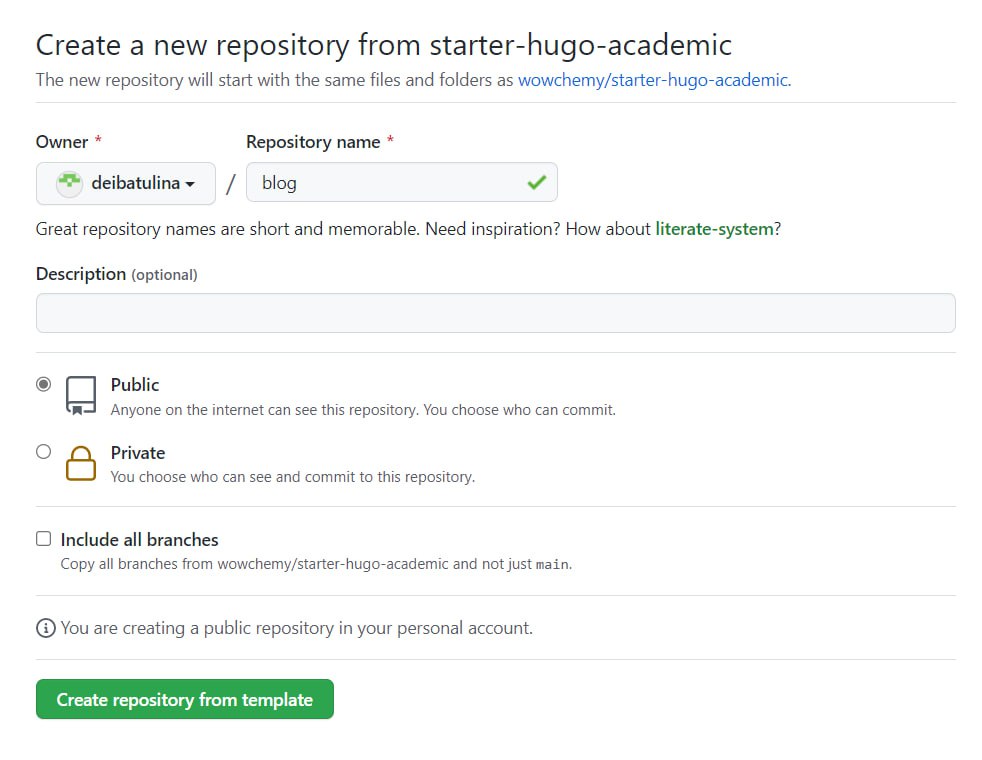


Figure 3: Создание репозитория blog

Следующий шаг - клонировать этот репозиторий (рис. [4](#fig:004)) в каталог work с помощью знакомой нам команды (рис. [5](#fig:005)):

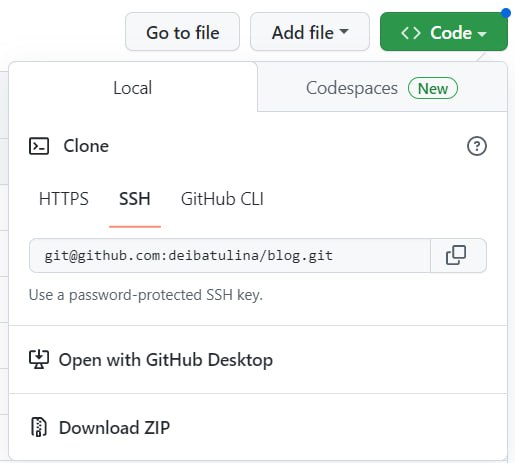


Figure 4: Копирование названия репозитория

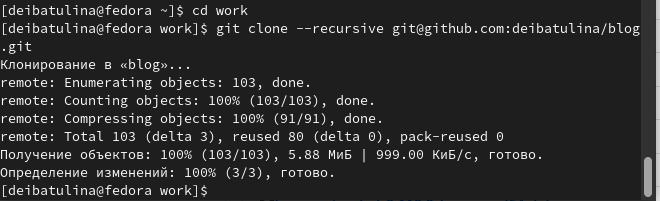


Figure 5: Клонирование репозитория в каталог work

Далее переходим в каталог blog и выводим список файлов (рис. [6](#fig:006)):

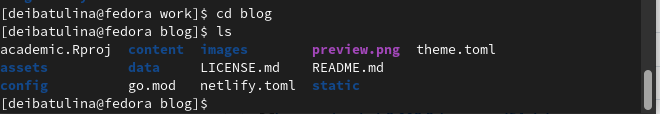


Figure 6: Список файлов каталога blog

Загружаем модули hugo (рис. [7](#fig:007)):

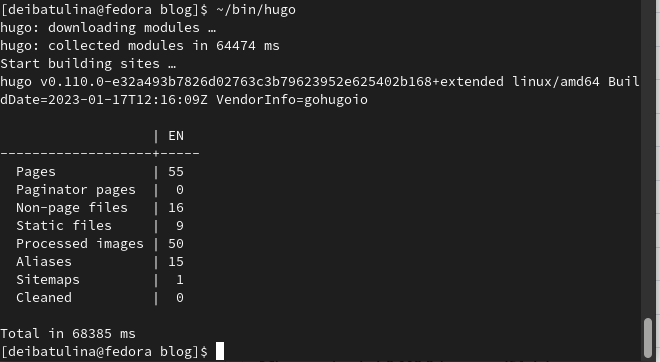


Figure 7: Загрузка модулей hugo

В каталоге blog мы видим каталог public, который в данный момент нам не понадобится, поэтому удаляем его (рис. [8](#fig:008)):

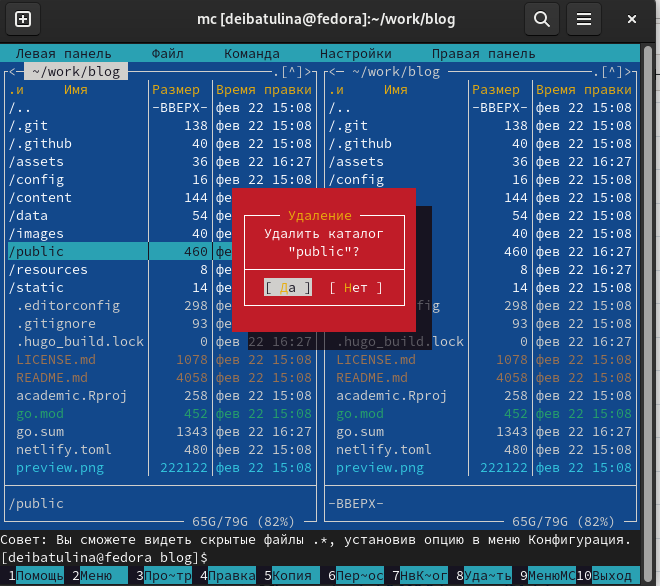


Figure 8: Удаление каталога public

Теперь мы имеем возможность открывать созданный нами сайт с нашего компьютера (сайт будет доступен только с него) (рис. [9](#fig:009)), (рис. [10](#fig:010)):



Figure 9: Открытие сайта с компьютера

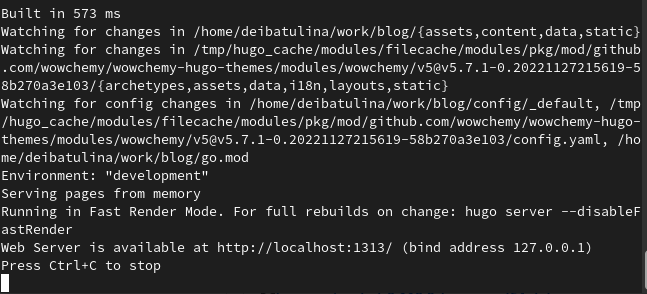


Figure 10: Открытие сайта с компьютера

Копируем ссылку localhost и вставляем в браузер (рис. [11](#fig:011)):

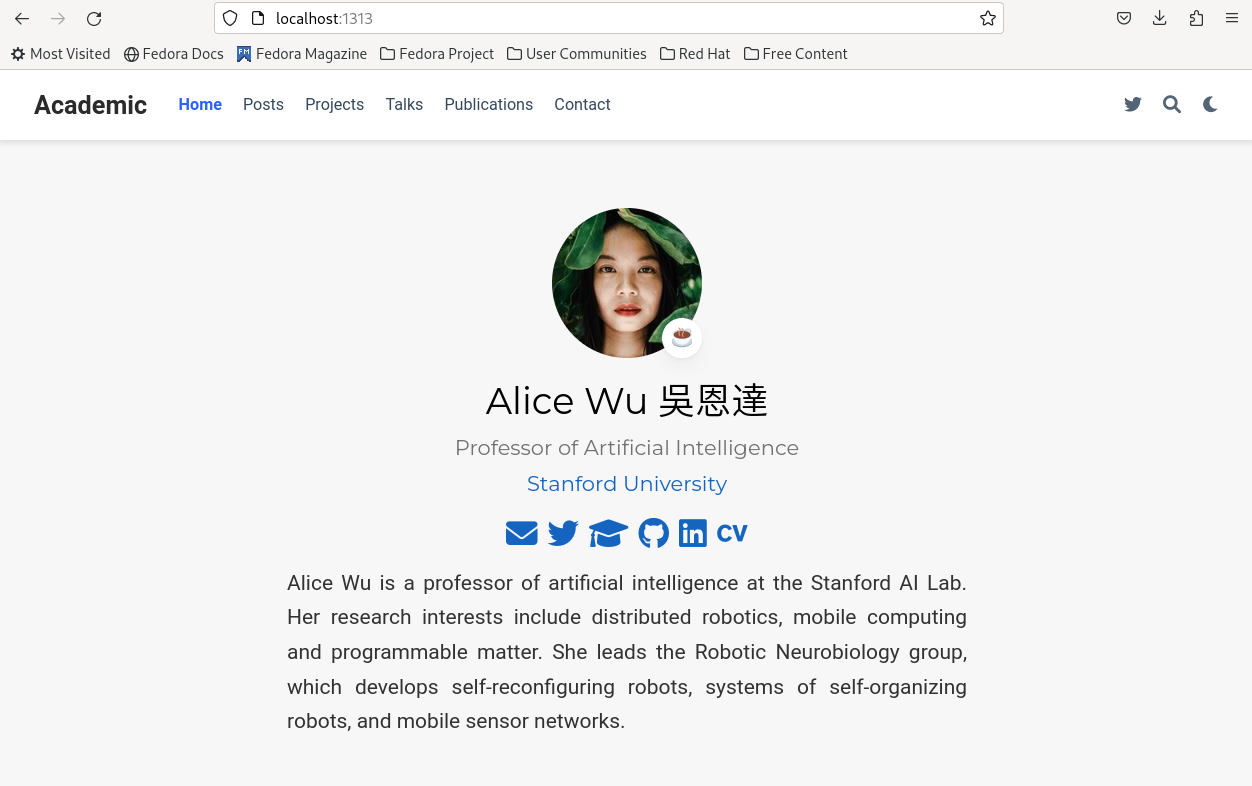


Figure 11: Открытие сайта в браузере

Просматривать данный сайт пока имею возможность только я с моего компьютера. Теперь создадим репозиторий на github, в котором будет размещён общедоступный сайт (рис. [12](#fig:012)):

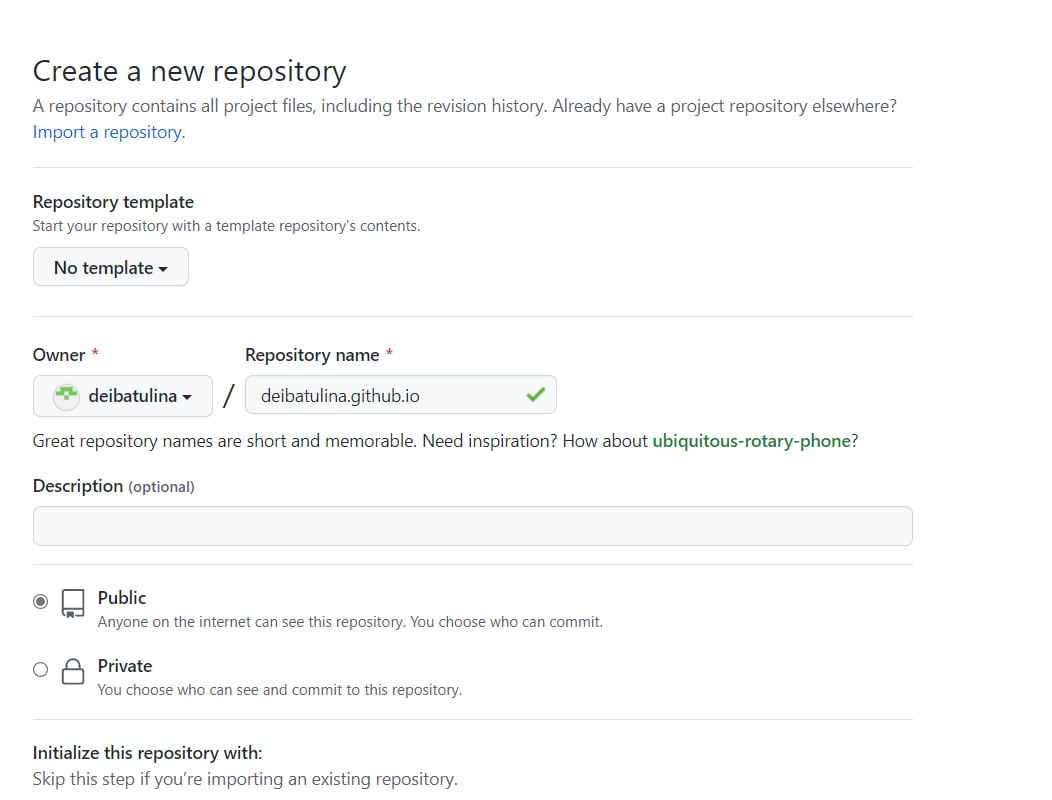


Figure 12: Создание репозитория для общедоступного сайта

Поднимаемся в терминале на структуру выше и выводим список файлов и видим, что в каталоге work помещены каталоги study (для лабораторных работ) и work (для индивидуального проекта) (рис. [13](#fig:013)):

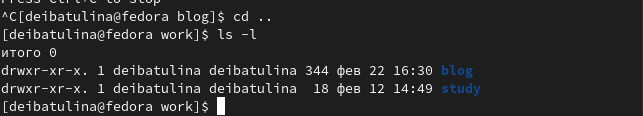


Figure 13: Вывод списка файлов каталога work

Теперь клонируем репозиторий для общедоступного сайта (github.io) в каталог work (рис. [14](#fig:014)):

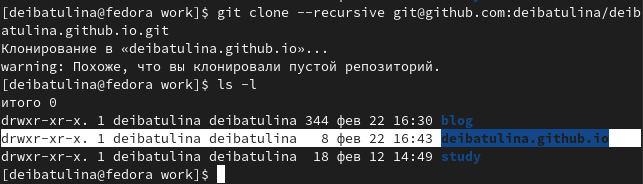


Figure 14: Клонирование репозитория c сайтом в каталог work

Переключаемся на ветку main (рис. [15](#fig:015)):

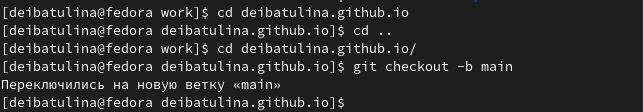


Figure 15: Переключение на ветку main

Создаём в репозитории файл README.md, а затем отправляем изменённые файлы на Github, используя хорошо знакомые нам команды (рис. [16](#fig:016)):

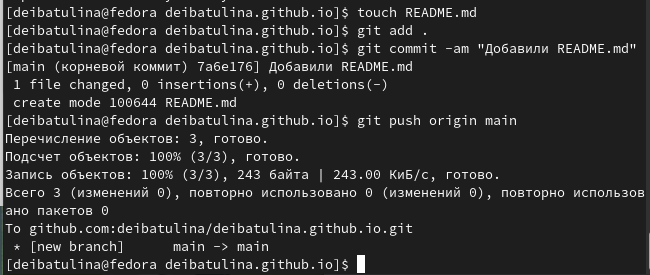


Figure 16: Отправка изменённых файлов на Github

Переходим в браузер и проверяем, что файл README.md находится в нужном репозитории на гитхабе (рис. [17](#fig:017)):

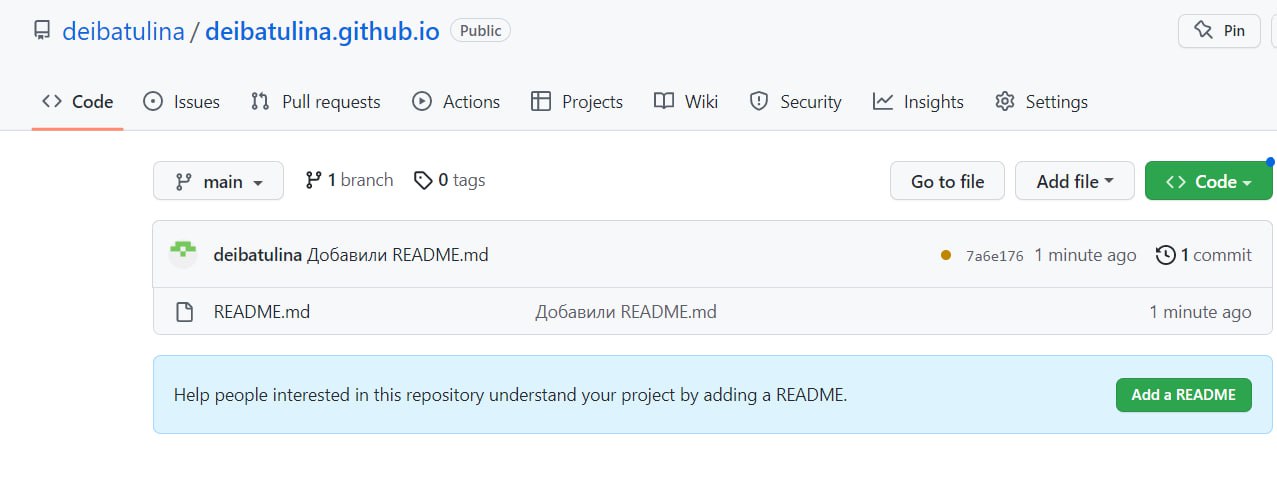


Figure 17: Проверка отправленных файлов

Теперь добавляем подмодуль, в нашем репозитории должен появиться каталог public (рис. [18](#fig:018)):

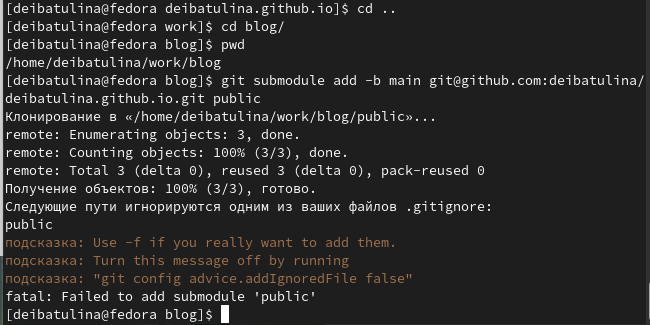


Figure 18: Добавление подмодуля в репозиторий

Однако, этого не происходит, поскольку в файле .gitignore прописано игнорирование файлов с именем public. Это необходимо исправить, отредактировав данный файл (рис. [19](#fig:019)):

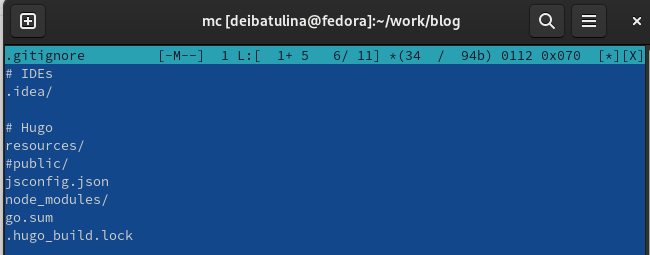


Figure 19: Редактирование файла .gitignore

Сохраняем изменения в файле и проверяем, что всё в порядке (рис. [20](#fig:020)):



Figure 20: Вывод содержимого файла .gitignore

Повторяем комнаду, отвечающую за добавление подмодуля (рис. [21](#fig:021)):

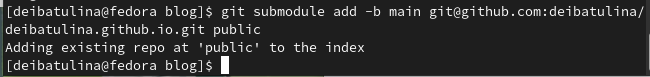


Figure 21: Добавление подмодуля в репозиторий (повторно)

На этот раз всё прошло успешно и тепер мы выполняем построение нашего сайта (рис. [22](#fig:022)):

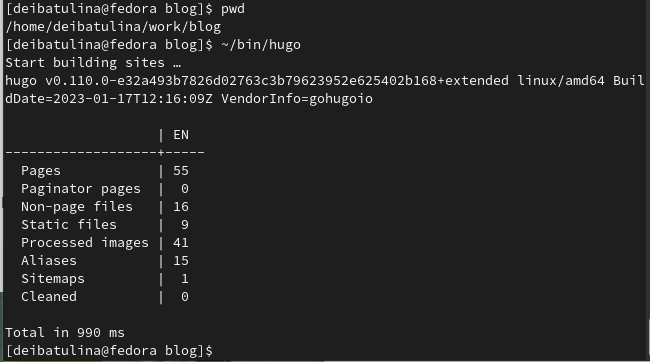


Figure 22: Построение сайта

Проверяем, что в блоге появиля каталог public (рис. [23](#fig:023)):

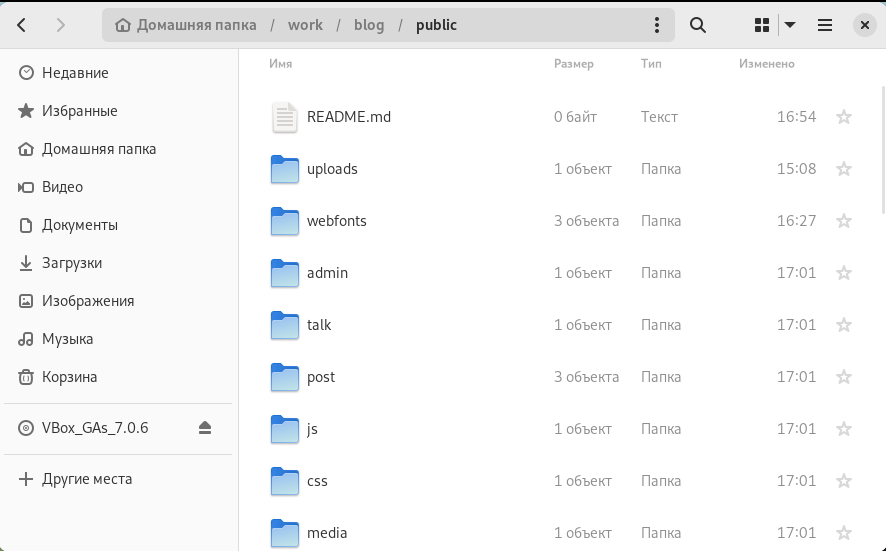


Figure 23: Проверка существования каталога public

Убеждаемся, что каталог для редактирования сайта на компьютере связан с нужной веткой на гитхабе (рис. [24](#fig:024)):

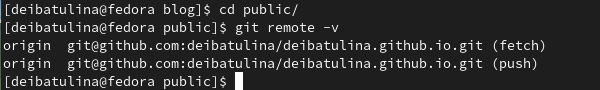


Figure 24: Проверка связывания сайта с нужной веткой на гитхабе

Добавляем на гитхаб все изменённые файлы и каталоги, отправляем изменения на сервер (рис. [25](#fig:025), [26](#fig:026)):

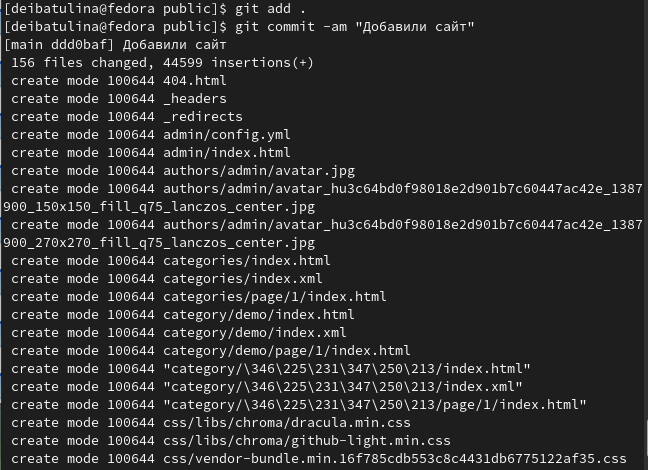


Figure 25: Отправка изменений на гитхаб

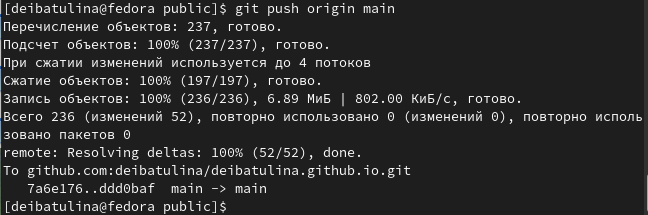


Figure 26: Отправка изменений на гитхаб

Теперь удостоверяемся, что все изменённые каталоги добавлены на гитхаб (рис. [27](#fig:027)):

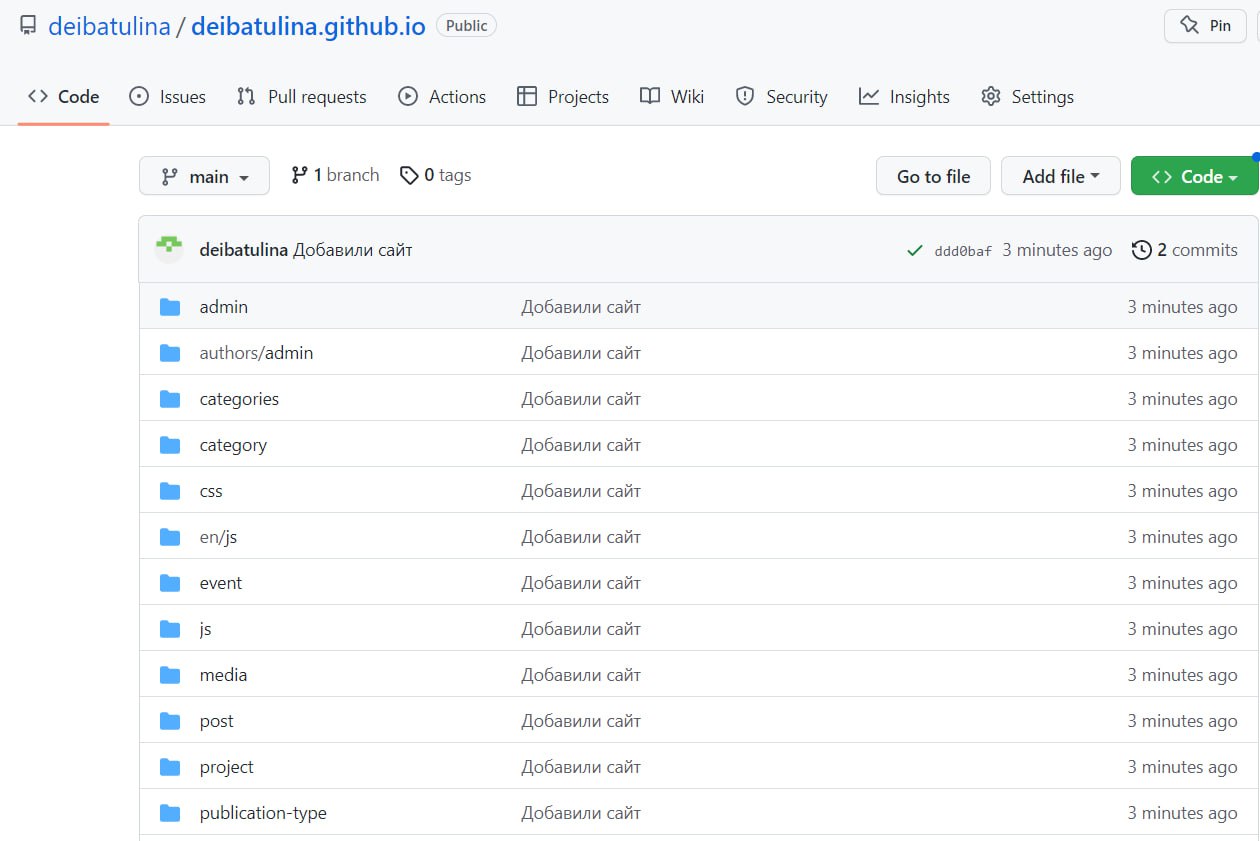


Figure 27: Проверка изменений

Остался поледний шаг: открываем наш сайт в браузере (рис. [28](#fig:028)):

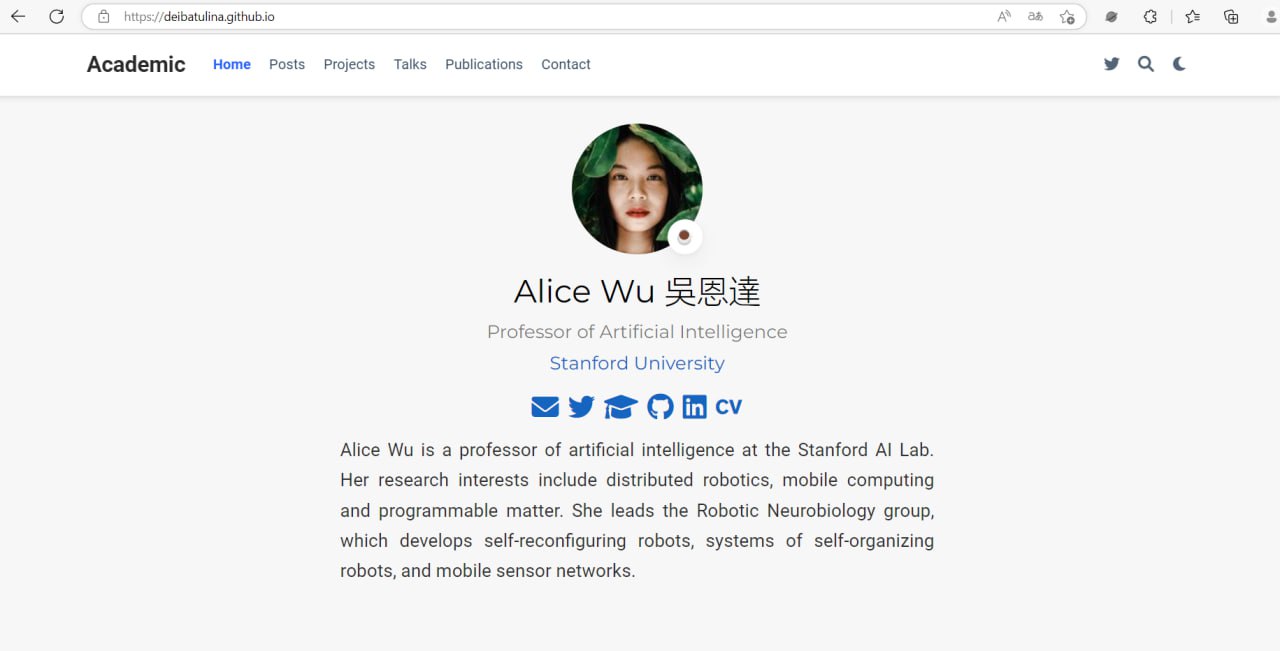


Figure 28: Открытие общедоступного сайта в браузере

Наш сайт можно теперь открывать с любого компьютера! Заполнять его своими данными предстоит нам в следующем этапе проекта.

# 5 Выводы

Цель достигнута - на GithubPages размещена заготовка моего сайта.

# Список литературы

* (Статья на Habr.com - интернет-ресурсе для IT-специалистов)[https://habr.com/ru/post/700640/]