Отчёт по проекту. Первый этап

Королёв Иван Андреевич

Содержание

# 1 Цель работы

Размещение на Github pages заготовки для персонального сайта.

# 2 Задание

* Установить необходимое программное обеспечение.
* Скачать шаблон темы сайта.
* Разместить его на хостинге git.
* Установить параметр для URLs сайта.
* Разместить заготовку сайта на Github pages.

# 3 Теоретическое введение

Один из самых популярных генераторов статических сайтов с открытым исходным кодом, написан на языке Go. Благодаря своей удивительной скорости и гибкости, Hugo делает создание веб-сайтов увлекательным.

# 4 Выполнение лабораторной работы

## 4.1 Установка необходимого программного обеспечения.

1. Установка Hugo Extended и Go(sudo dnf install go hugo) (рис. [1](#fig:001)).

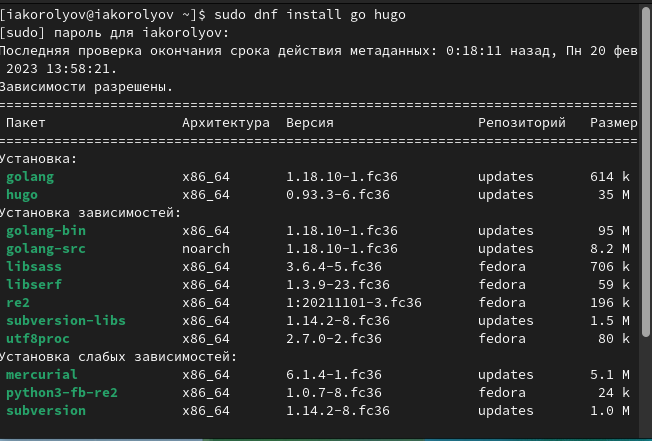


Figure 1: Hugo Extended и Go

1. После установки с помощью команды “sudo dnf install go hugo” - hugo установился версии 0.93.0 и не Extended. Нам же необходима версия 0.95 и выше. Тогда устанавливаем Hugo Extended в ручную. Следует скачать архив с репозитория: https://github.com/gohugoio/hugo/releases. Выбрал самую последнюю версию. Скопировал ссылку на загрузку. (рис. [2](#fig:002)) Далее я перешел в каталог tmp, создал новый в tmp и назвал 01. В tmp/01 я скачал данный архив.(рис. [3](#fig:003)) Потом я разархивировал скаченную папку и файл Hugo переместил в usr/local/bin. (рис. [4](#fig:004)), (рис. [5](#fig:005))

Figure 2: Hugo Extended

Figure 2: Hugo Extended

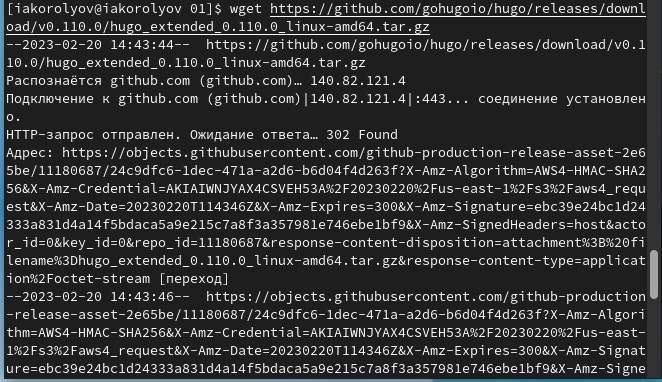


Figure 3: Hugo Extended

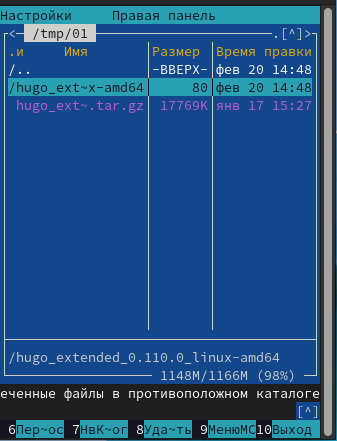


Figure 4: Hugo Extended

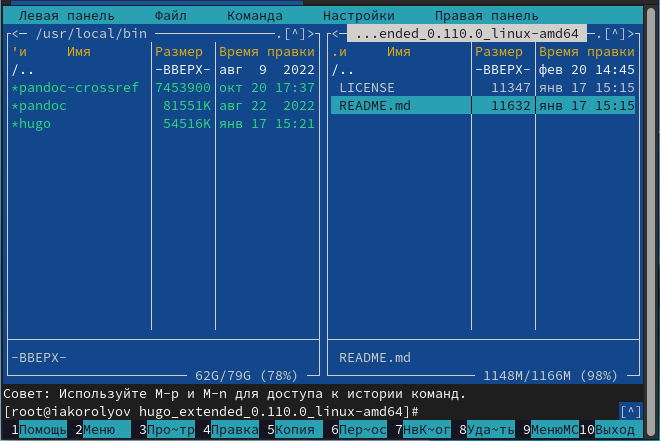


Figure 5: Hugo Extended

1. С помощью команды “hugo version” убедился, какая установлена версия. (рис. [6](#fig:006))

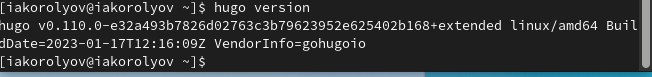


Figure 6: Hugo version

## 4.2 Скачать шаблон темы сайта.

1. На нашем курсе открывали техническая реализация. Открывали репозиторий Hugo Academic Theme. Создаем новый репозиторий нажимая на use this tempplate. Называем его blog и создаем. (рис. [7](#fig:007))

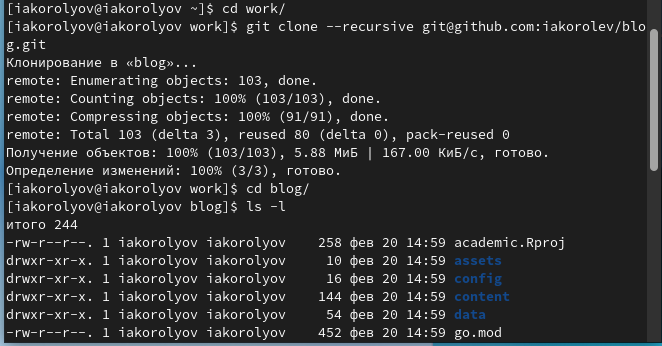


Figure 7: Blog

1. Переходим в каталог blog и прописываем команду hugo. Видим, что у нас есть папка public, нам необходимо её удалить. (рис. [8](#fig:008)), (рис. [9](#fig:009))

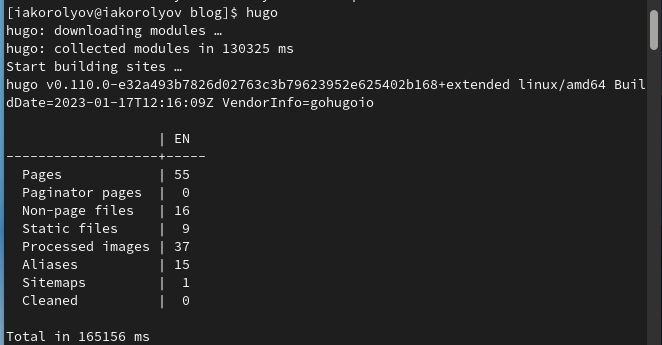


Figure 8: hugo

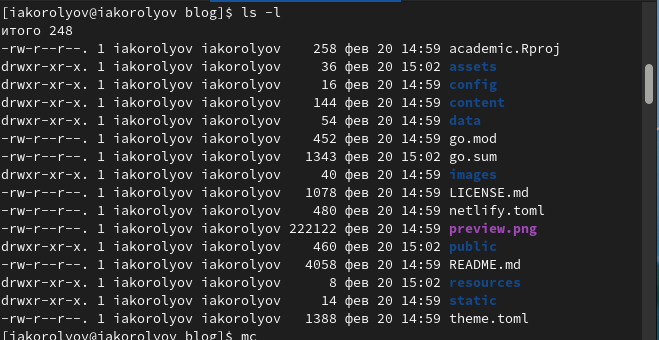


Figure 9: Папка public

1. Далее мы пишем команду hugo server и если все скомпилируется правильно, то не будет никаких ошибок. В самом низу будет написано, что ctrl + c - это выход из программы. И чуть выше мы ввидим, что у нас есть ссылка http://localhost:1313/ Перейдя по ней мы увидим, что всё работает, но сайт работает только на одном компьютере, а нам необходимо чтобы он запускался на любом. Нужно, чтобы любой мог получать доступ к моему сайту. Для этого надо разместить его на github (рис. [10](#fig:0010))

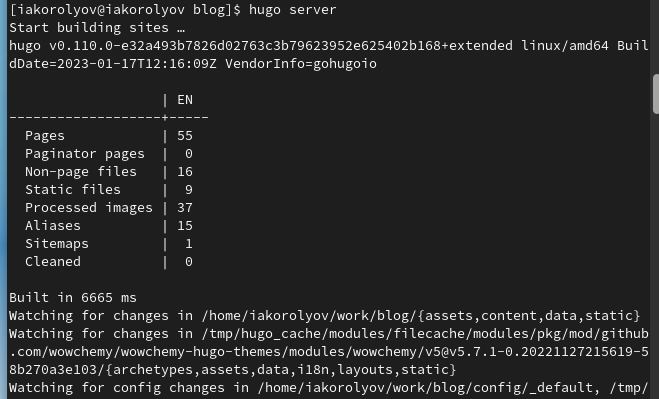


Figure 10: server

## 4.3 Размещение его на хостинге git. Установка параметра для URLs сайта. Размещение заготовки сайта на Github pages.

1. Создаем новый репозиторий на git. ВАЖНО, чтобы при создании имя совпадало с тем, которое указано в самом git, а после добавляеми github.io ( должно получится вот так: iakorolev.github.io ) (рис. [11](#fig:0011))

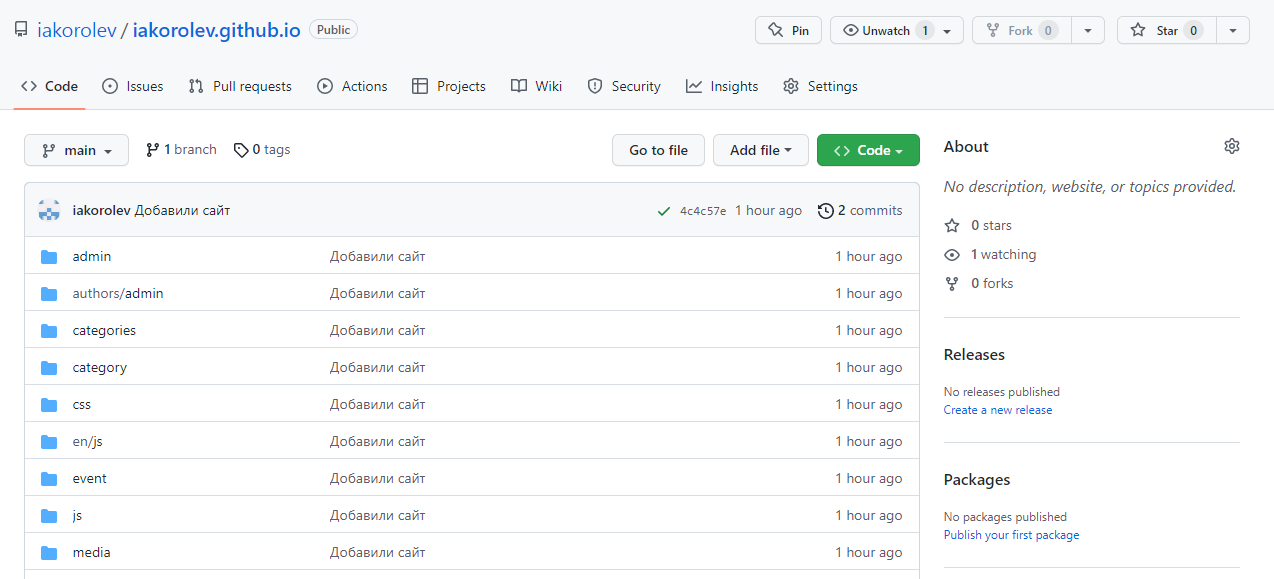


Figure 11: iakorolev.github.io

1. Копируем ссылку и клонируем его перейдя в каталог work. (рис. [12](#fig:0012))

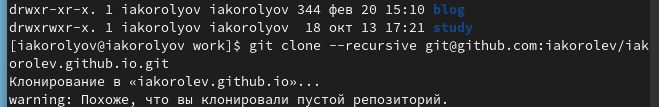


Figure 12: work

1. Т.к. репозиторий у нас пустой необходимо создать ветку. Ветка должна обязательно называться main ( git checkout -b main ) (рис. [13](#fig:0013))

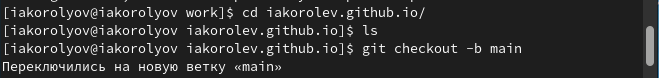


Figure 13: git checkout -b main

1. После этого мы создаем пустой readme.md ( touch README.md ) и отправляем это все на git (рис. [14](#fig:0014))

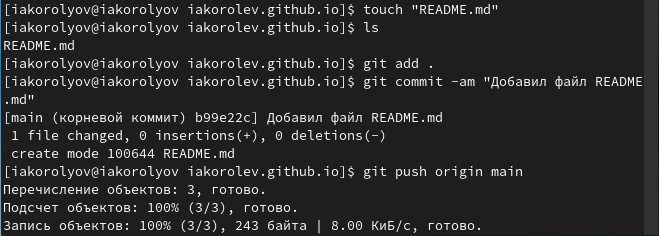


Figure 14: git push, readme.md

1. Переходим обратно в blog. Необходимо к папке public подключить наш созданный репозиторий iakorolev.github.io. (рис. [15](#fig:0015))

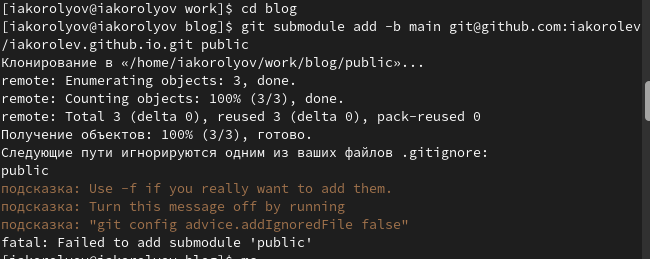


Figure 15: git submodule add -b main .. public

1. Public проигнорировался. Исправим это, перейдя в mc, там открываем файл gitignore. И комментируем значение public. (рис. [16](#fig:0016))

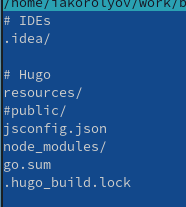


Figure 16: public

1. Повторяем команду git submodule add -b main .. public ещё раз (рис. [17](#fig:0017))

Figure 17: git submodule add -b main .. public

Figure 17: git submodule add -b main .. public

1. Пока наша папка public пуста. Но если мы пропишем hugo, то она заполнится необходимыми файлами. (рис. [18](#fig:0018))

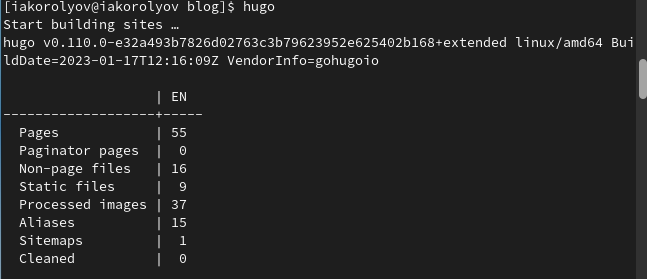


Figure 18: hugo and public

1. Проверяем, перейдя в каталог. Так же, с помощью команды git remote -v ( видим в строке, что наш репозиторий подключен и всё работает ) (рис. [19](#fig:0019))

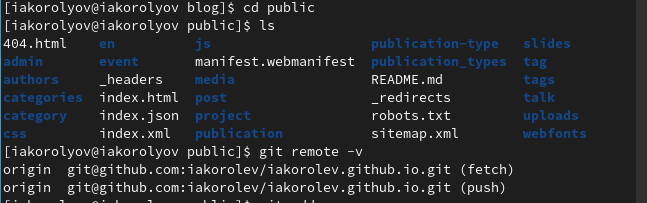


Figure 19: Проверка

1. Отправляем всё на git. (рис. [20](#fig:0020))

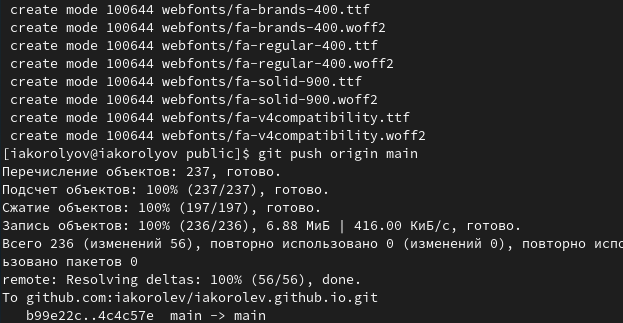


Figure 20: git

## 4.4 Сайт

Сайт работает, демонстрирую на скриншоте. (рис. [21](#fig:0021))

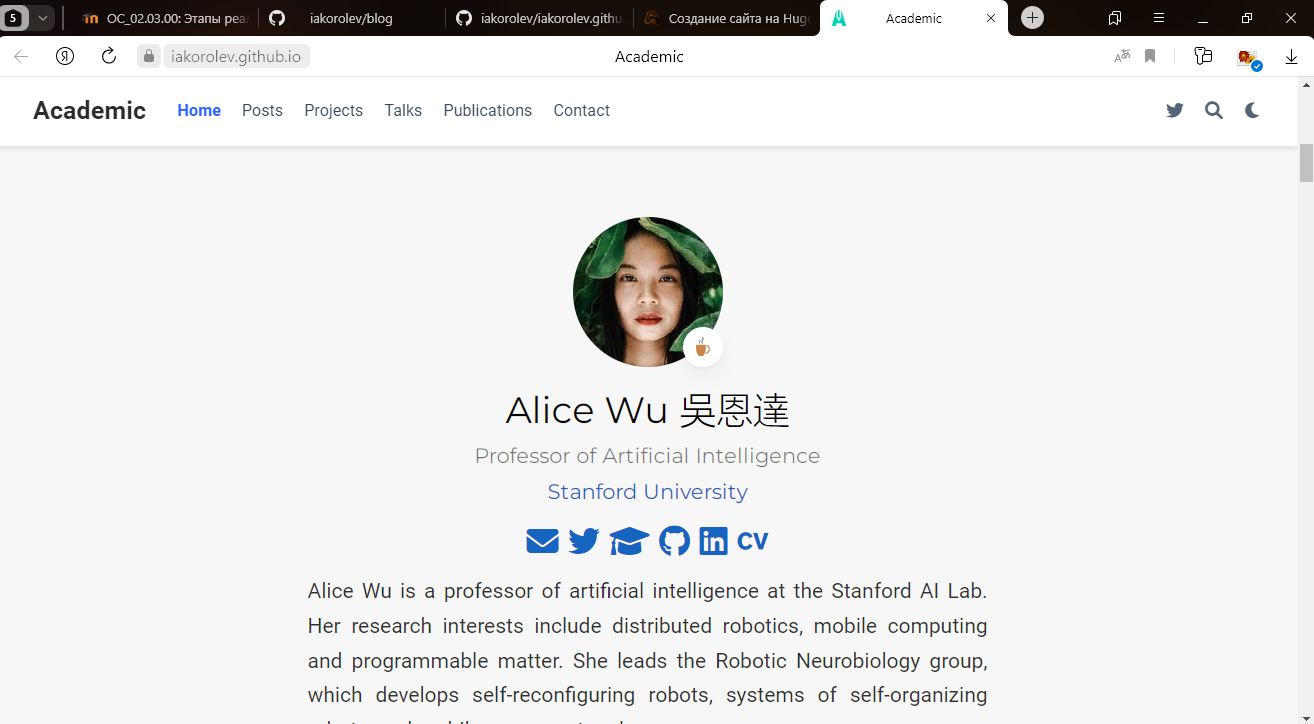


Figure 21: Сайт

# 5 Выводы

Я разместил на Github pages заготовки для персонального сайта.