

Выполнение индивидуального проекта. Стадия 1.

Отчет

Анна Александровна Глушенок

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение индивидуального проекта	7
4	Выводы	22
	Список литературы	23

Список иллюстраций

3.1	Скачивание hugo sites (1)	7
3.2	Скачивание hugo sites (2)	8
3.3	Распаковка архива, перемещение hugo	8
3.4	Переход к шаблону	9
3.5	Создание репозитория по шаблону	9
3.6	Ввод имени репозитория	10
3.7	Клонирование репозитория	10
3.8	Скачивание go	11
3.9	Сборка сайта	12
3.10	Создание 2-го нового репозитория	13
3.11	Именованное репозитория	14
3.12	Клонирование репозитория, переход на ветку main	15
3.13	Выполнение коммита	15
3.14	Добавление submodule, удаление public	16
3.15	Повторная сборка; Проверка удаления public	17
3.16	Выполнение коммита	17
3.17	Отправка файлов	18
3.18	Проверка наличия файлов	19
3.19	Копирование имени репозитория	20
3.20	Открытие сайта по ссылке	21

Список таблиц

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 Выполнение индивидуального проекта

Скачаем hugo sites с github:

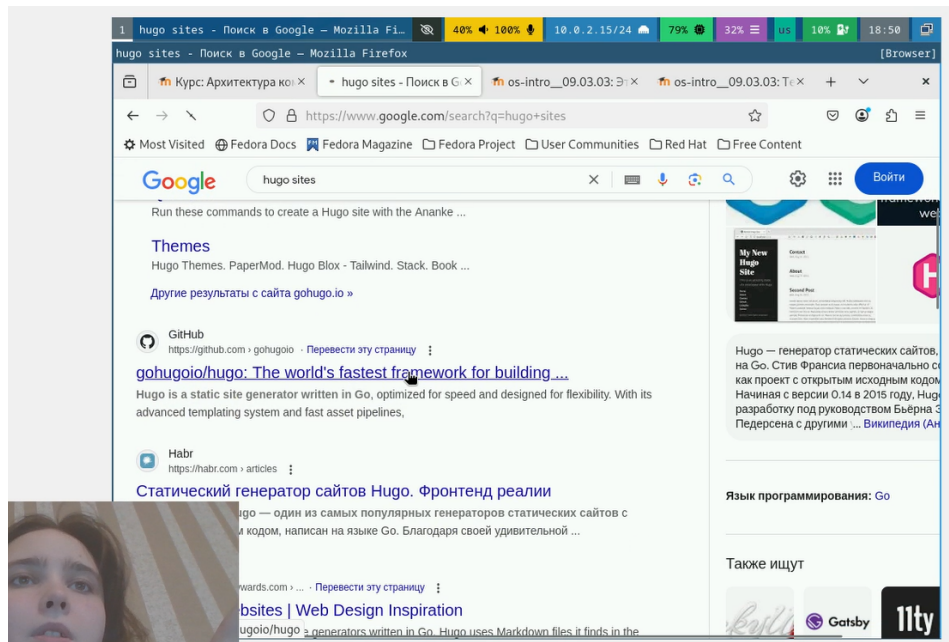


Рис. 3.1: Скачивание hugo sites (1)

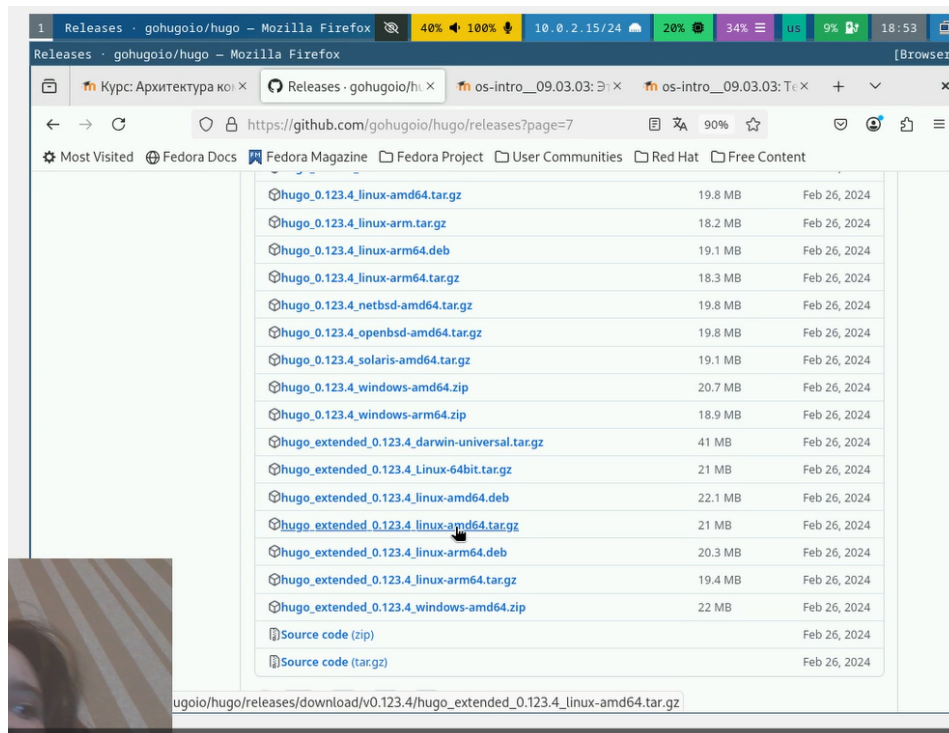


Рис. 3.2: Скачивание hugo sites (2)

Распакуем установленный архив, переместим файл hugo в каталог /usr/local/bin:

```
[aaglushenok@aaglushenok Загрузки]$ ls
hugo_extended_0.123.4_linux-amd64.tar.gz  pandoc-crossref.1  pandoc-crossref-Linux.tar.xz
[aaglushenok@aaglushenok Загрузки]$ tar -xvf hugo_extended_0.123.4_linux-amd64.tar.gz
hugo
README.md
LICENSE
[aaglushenok@aaglushenok Загрузки]$ ls
hugo  LICENSE  pandoc-crossref-Linux.tar.xz
hugo_extended_0.123.4_linux-amd64.tar.gz  pandoc-crossref.1  README.md
[aaglushenok@aaglushenok Загрузки]$ sudo mv hugo /usr/local/bin
[sudo] пароль для aaglushenok:
[aaglushenok@aaglushenok Загрузки]$ ls
hugo_extended_0.123.4_linux-amd64.tar.gz  pandoc-crossref.1  README.md
LICENSE  pandoc-crossref-Linux.tar.xz
[aaglushenok@aaglushenok Загрузки]$
```

Рис. 3.3: Распаковка архива, перемещение hugo

На ТУИСе перейдем в раздел “Техническая реализация проекта”, перейдем по ссылке на репозиторий с необходимыми файлами. Нажмем “use this template” -> “create new repository”, создав новый репозиторий по заданному шаблону:

Техническая реализация проекта

- Для реализации сайта используется генератор статических сайтов *Hugo*.
- Общие файлы для тем *Wowchemy*:
 - Репозиторий: <https://github.com/wowchemy/wowchemy-hugo-themes>
- В качестве шаблона индивидуального сайта используется шаблон *Hugo Academic Theme*.
 - Демо-сайт: <https://academic-demo.netlify.app/>
 - Репозиторий: <https://github.com/wowchemy/starter-hugo-academic>

Время: Воскресенье, 17 апреля 2022, 20:05

Рис. 3.4: Переход к шаблону

The screenshot shows the GitHub interface for the repository 'Hugoblox / theme-academic-cv'. The repository is a public template with 34 watches, 4.2k stars, and 6.4k forks. A dropdown menu is open under the 'Use this template' button, showing three options: 'Create a new repository' (highlighted), 'Open in a codespace', and 'Use this template'. Below the repository name, there is a table of files and folders, including '.github', 'assets/media', 'config/_default', and 'content'. On the right side, there is a description in Chinese and English about creating an academic website using Hugo and GitHub, along with a link to 'hugoblox.com/templates/' and a list of other templates like 'resume-template', 'hugo-theme', 'hugo', 'rstudio', 'personal-website', 'hugo-site', 'portfolio-website', 'academic', 'blog-engine', 'rmarkdown', 'website-generation', 'netlify', 'obsidian', 'portfolio-template', and 'academic-website'.

Рис. 3.5: Создание репозитория по шаблону

Зададим имя нового репозитория website (далее изменен на “blog”):

Owner * aaglushenok / Repository name * website

website is available.

Great repository names are short and memorable. Need inspiration? How about [animated-fiesta](#) ?

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

ⓘ You are creating a public repository in your personal account.

Create repository

Рис. 3.6: Ввод имени репозитория

Клонируем созданный репозиторий в папку work:

```
[aaglushenok@aaglushenok ~]$ cd ~/work
[aaglushenok@aaglushenok work]$ git clone --recursive git@github.com:aaglushenok/website.git
Клонирование в «website»...
remote: Enumerating objects: 98, done.
remote: Counting objects: 100% (98/98), done.
remote: Compressing objects: 100% (84/84), done.
remote: Total 98 (delta 5), reused 75 (delta 1), pack-reused 0 (from 0)
Получение объектов: 100% (98/98), 4.12 МБ | 199.00 КиБ/с, готово.
Определение изменений: 100% (5/5), готово.
[aaglushenok@aaglushenok work]$
```

Рис. 3.7: Клонирование репозитория

Скачаем go через `sudo dnf install go`:

```

foot
[aaglushmanok@aaglushmanok work]$ cd website
[aaglushmanok@aaglushmanok website]$ sudo dnf install go
[sudo] пароль для aaglushmanok:
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет      Арх.      Версия      Репозиторий      Размер
Установка:
golang      x86_64     1.23.6-1.fc41      updates          8.9 MiB
Установка зависимостей:
go-filesystem      x86_64     3.6.0-5.fc41      updates          0.0 B
golang-bin         x86_64     1.23.6-1.fc41      updates          113.9 MiB
golang-src         noarch     1.23.6-1.fc41      updates          76.3 MiB
Установка слабых зависимостей:
mercurial         x86_64     6.8.2-1.fc41      updates          30.8 MiB

Сводка транзакции:
Установка:      5 пакетов

Общий размер входящих пакетов составляет 47 MiB. Необходимо загрузить 47 MiB.
После этой операции будут использоваться дополнительные 230 MiB (установка 230 MiB, удаление 0 B).
Is this ok [y/N]: y
[1/5] golang-0:1.23.6-1.fc41.x86_64      100% | 1.3 MiB/s | 668.9 KiB | 00m00s
[2/5] go-filesystem-0:3.6.0-5.fc41.x86_64 100% | 61.4 KiB/s | 9.0 KiB | 00m00s
[3/5] mercurial-0:6.8.2-1.fc41.x86_64    100% | 866.8 KiB/s | 6.5 MiB | 00m08s
[4/5] golang-src-0:1.23.6-1.fc41.noarch  100% | 1.1 MiB/s | 12.8 MiB | 00m12s
[5/5] golang-bin-0:1.23.6-1.fc41.x86_64  100% | 1.4 MiB/s | 27.5 MiB | 00m20s
-----
[5/5] Total                             100% | 2.2 MiB/s | 47.4 MiB | 00m21s
Выполнение транзакции
[1/7] Проверить файлы пакета              100% | 14.0 B/s | 5.0 B | 00m00s00s
[2/7] Подготовить транзакцию              100% | 3.0 B/s | 5.0 B | 00m01s00s
[3/7] Установка go-filesystem-0:3.6.0-5.fc41.x86_64 100% | 8.3 KiB/s | 392.0 B | 00m00s
[4/7] Установка golang-src-0:1.23.6-1.fc41.x86_64 59% [===== ] | 2.0 MiB/s | 45.9 MiB | 00m15s

```

Рис. 3.8: Скачивание go

Перейдем в каталог website и введем hugo server, после чего сайт “соберется”, и в терминале мы получим ссылку и номер порта для открытия сайта. После перехода по ссылке видим открывшийся сайт:

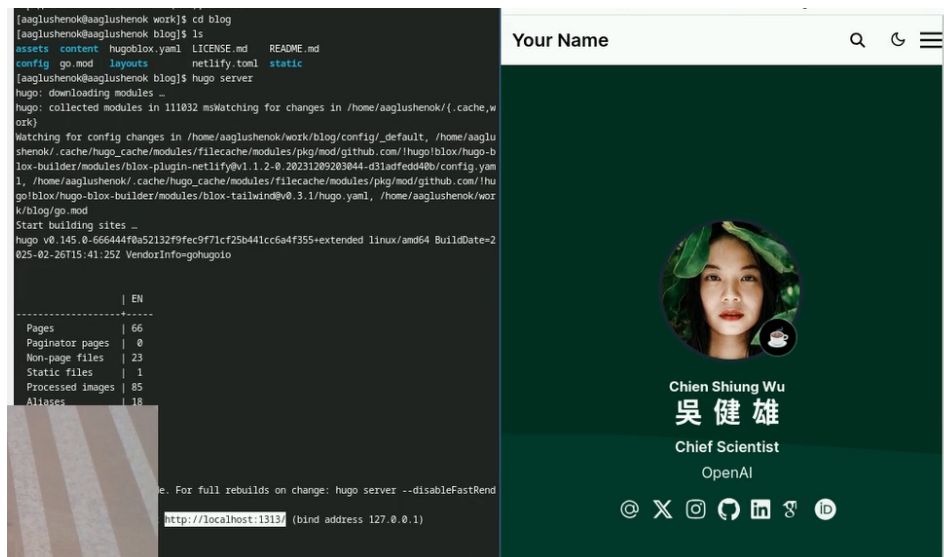


Рис. 3.9: Сборка сайта

Создадим еще один новый репозиторий, зададим его имя как “aaglushmanok.github.io”:

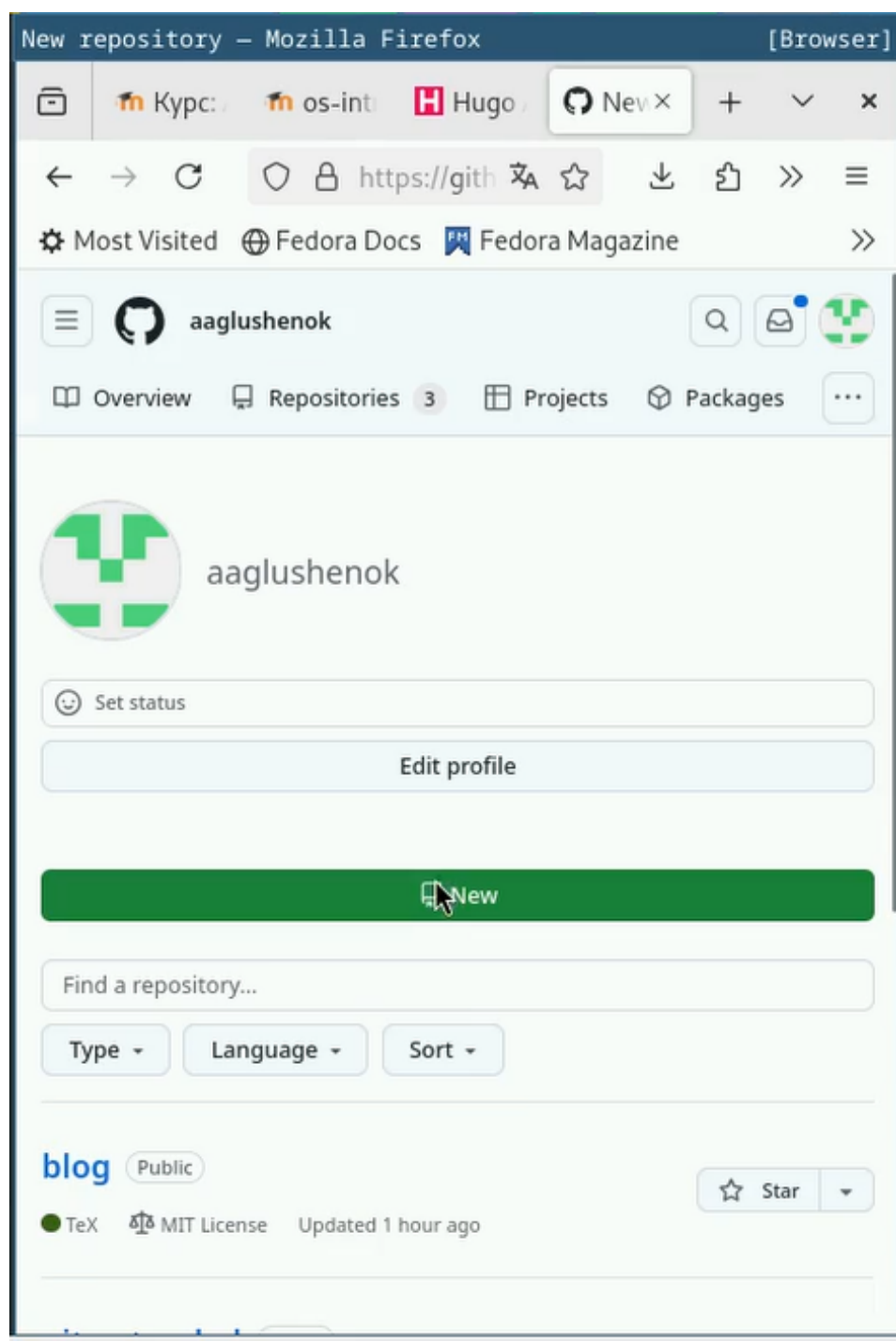


Рис. 3.10: Создание 2-го нового репозитория

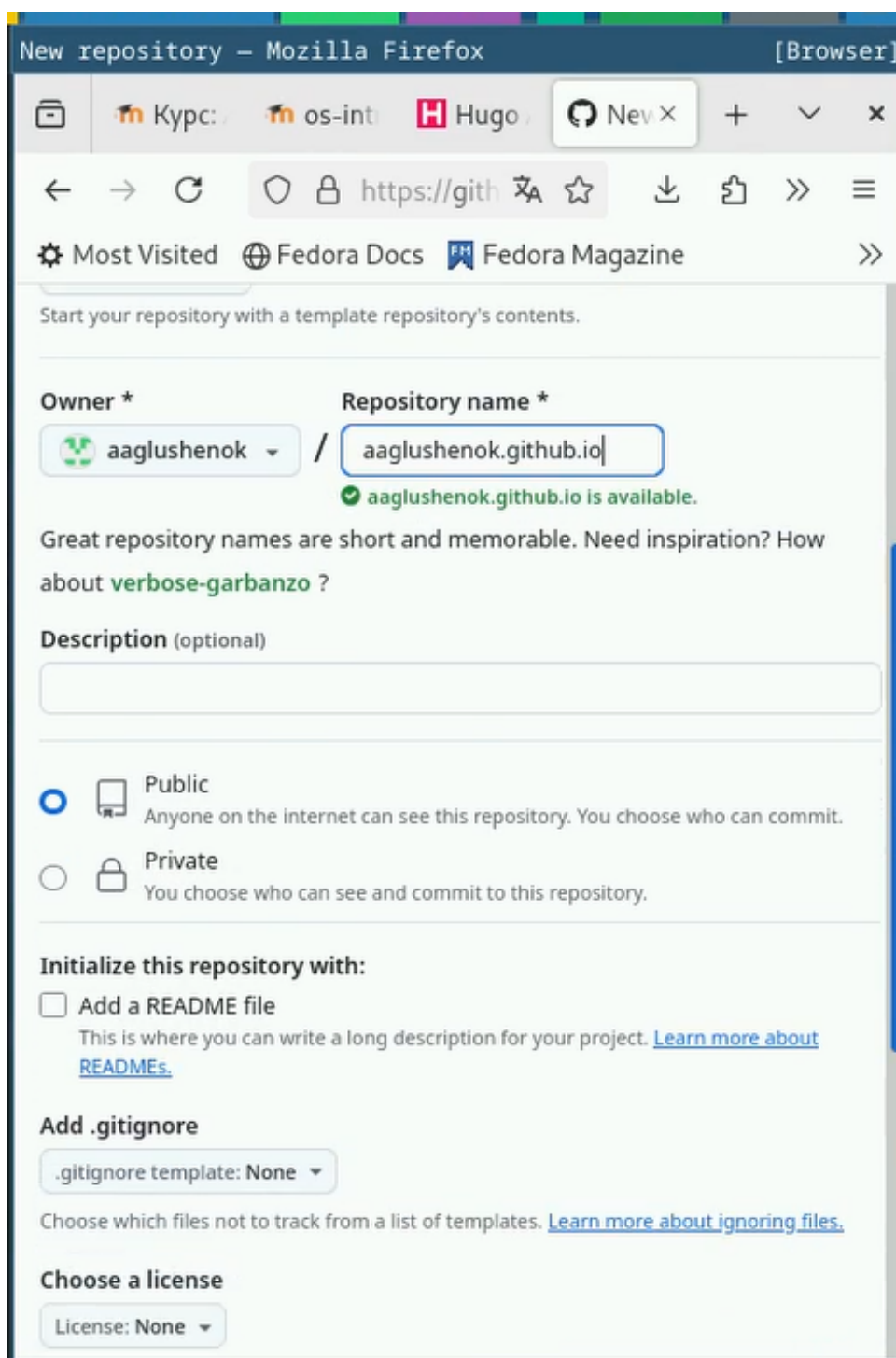


Рис. 3.11: Именованное репозитория

Клонируем 2-ой созданный репозиторий в папку work. Перейдем в папку с материалами репозитория. Используя "git checkout -b main" переключимся на

новую ветку main:

```
^C[aaglushmanok@aaglushmanok blog]$ cd
[aaglushmanok@aaglushmanok ~]$ cd ~/work
[aaglushmanok@aaglushmanok work]$ git clone --recursive git@github.com:aaglushmanok/aaglushmanok.github.io.git
Клонирование в «aaglushmanok.github.io»...
warning: Похоже, что вы клонировали пустой репозиторий.
[aaglushmanok@aaglushmanok work]$ ls
aaglushmanok.github.io  blog  study  website
[aaglushmanok@aaglushmanok work]$ cd aaglushmanok.github.io
[aaglushmanok@aaglushmanok aaglushmanok.github.io]$ git checkout -b main
Переключились на новую ветку «main»
```

Рис. 3.12: Клонирование репозитория, переход на ветку main

Для того чтобы не работать с пустым репозиторием, создадим фиктивный файл README.md. Пропишем команду `git add .`, выполним коммит.

```
[aaglushmanok@aaglushmanok aaglushmanok.github.io]$ touch README.md
[aaglushmanok@aaglushmanok aaglushmanok.github.io]$ git add .
[aaglushmanok@aaglushmanok aaglushmanok.github.io]$ git commit -am "Created README.md"
[main (корневой коммит) b5638da] Created README.md
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
[aaglushmanok@aaglushmanok aaglushmanok.github.io]$ git push
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 871 байт | 435.00 КиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:aaglushmanok/aaglushmanok.github.io.git
 * [new branch]      main -> main
[aaglushmanok@aaglushmanok aaglushmanok.github.io]$ cd
```

Рис. 3.13: Выполнение коммита

Перейдем в каталог `blog` (бывш. `website`), удалим файл `public`. Добавим submodule:


```

[aaglushmanok@aaglushmanok ~]$ cd ~/work/blog
[aaglushmanok@aaglushmanok blog]$ ls
assets content go.sum hugo_stats.json LICENSE.md public resources
config go.mod hugoblox.yaml layouts netlify.toml README.md static
[aaglushmanok@aaglushmanok blog]$ git submodule add -b main git@github.com:aaglushmanok/aaglushmanok.github.io.git public
fatal: 'public' already exists and is not a valid git repo
[aaglushmanok@aaglushmanok blog]$ nano .gitignore
[aaglushmanok@aaglushmanok blog]$ ls
assets content go.sum hugo_stats.json LICENSE.md public resources
config go.mod hugoblox.yaml layouts netlify.toml README.md static
[aaglushmanok@aaglushmanok blog]$ rm -R public/
[aaglushmanok@aaglushmanok blog]$ ls
assets content go.sum hugo_stats.json LICENSE.md README.md static
config go.mod hugoblox.yaml layouts netlify.toml resources
[aaglushmanok@aaglushmanok blog]$ git submodule add -b main git@github.com:aaglushmanok/aaglushmanok.github.io.git public
Клонирование в «/home/aaglushmanok/work/blog/public»...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0 (from 0)
Получение объектов: 100% (3/3), готово.
[aaglushmanok@aaglushmanok blog]$

```

Рис. 3.14: Добавление submodule, удаление public

Повторно пропишем hugo, после “сборки” сайта убедимся, что `public` удален, затем введем “`git remote -v`” :


```
[aaglushenok@aaglushenok blog]$ hugo
Start building sites ...
hugo v0.145.0-666444f0a52132f9fec9f71cf25b441cc6a4f355+extended linux/amd64 BuildDate=2025-02-26T15:41:25Z VendorInfo=gohugoio

| EN
-----+-----
Pages | 66
Paginator pages | 0
Non-page files | 23
Static files | 1
Processed images | 85
Aliases | 18
Cleaned | 0

Total in 3075 ms
[aaglushenok@aaglushenok blog]$ cd public/
[aaglushenok@aaglushenok public]$ ls
404.html      dist          index.html    post          publication_types  sitemap.xml
author        event         index.xml     project       README.md         tags
backlinks.json  experience   js           projects      _redirects        teaching
css           _headers     media        publication  robots.txt        uploads
[aaglushenok@aaglushenok public]$ git remote -v
origin  git@github.com:aaglushenok/aaglushenok.github.io.git (fetch)
origin  git@github.com:aaglushenok/aaglushenok.github.io.git (push)
```

Рис. 3.15: Повторная сборка; Проверка удаления public

Введем команду `git add .`, выполним коммит. Пропишем команду `git push` для отправки файлов, с указанием ветки `origin main`:

```
[aaglushenok@aaglushenok public]$ git add .
[aaglushenok@aaglushenok public]$ git commit -am "added site"
```

Рис. 3.16: Выполнение коммита

```
create mode 100644 teaching/python/featured_hu_4f6890dcc47514ed.webp
create mode 100644 teaching/python/featured_hu_d9d57e069036e13e.webp
create mode 100644 teaching/python/index.html
create mode 100644 uploads/resume.pdf
[aaglushenok@aaglushenok public]$ git push origin main
Перечисление объектов: 307, готово.
Подсчет объектов: 100% (307/307), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (261/261), готово.
Запись объектов: 100% (306/306), 8.10 МиБ | 1.81 МиБ/с, готово.
Total 306 (delta 75), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (75/75), done.
To github.com:aaglushenok/aaglushenok.github.io.git
   b5638da..e6f3340  main -> main
[aaglushenok@aaglushenok public]$
```

Рис. 3.17: Отправка файлов

Откроем репозиторий на github, убедимся, что он заполнен всеми необходимыми файлами:

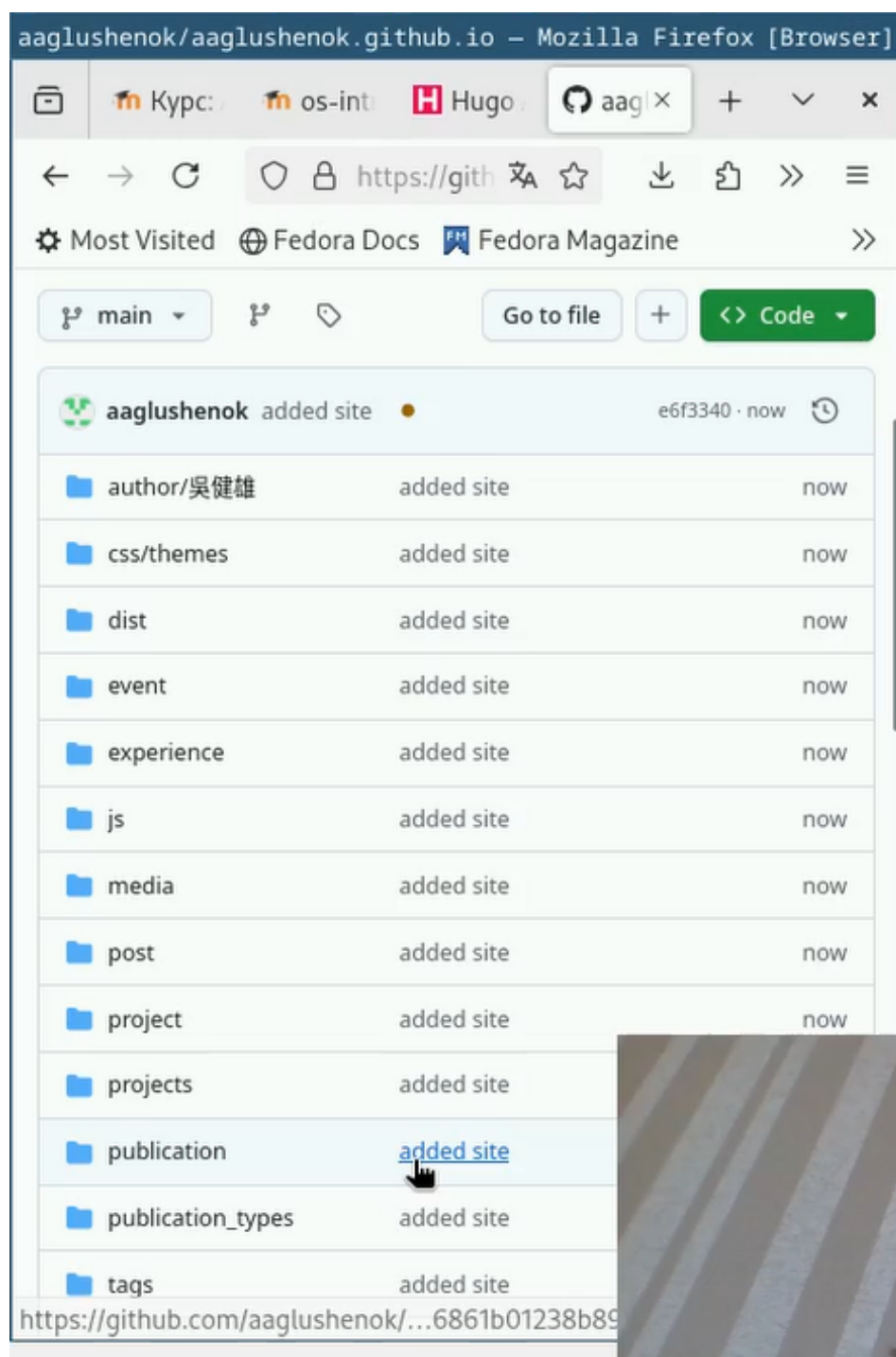


Рис. 3.18: Проверка наличия файлов

Копируем имя репозитория, введем его в виде ссылки в FireFox. Убедимся, что по введенной ссылке открывается наш сайт:

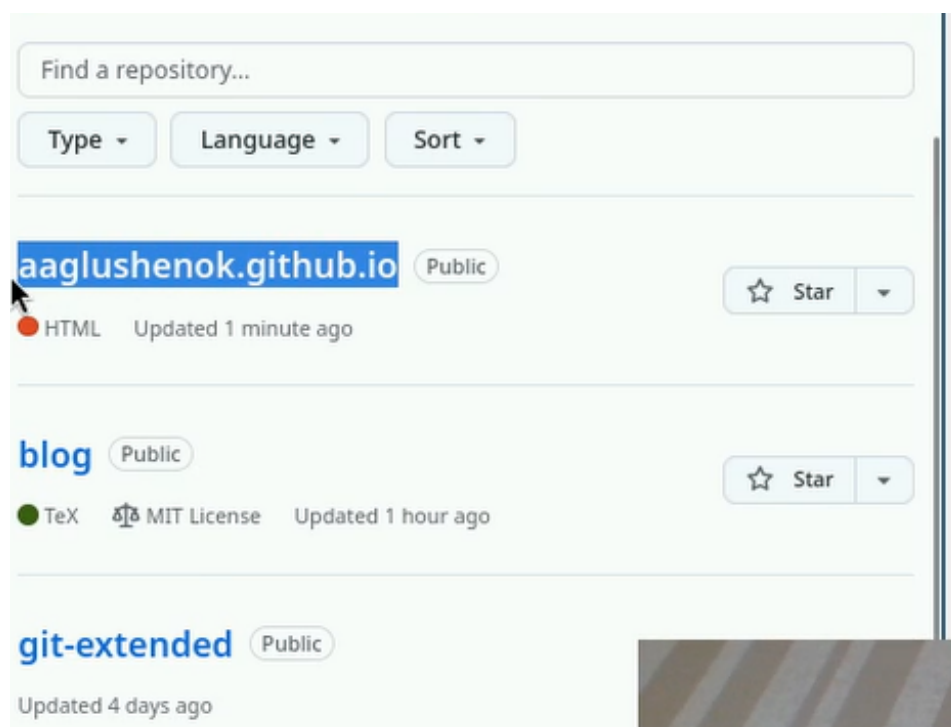


Рис. 3.19: Копирование имени репозитория

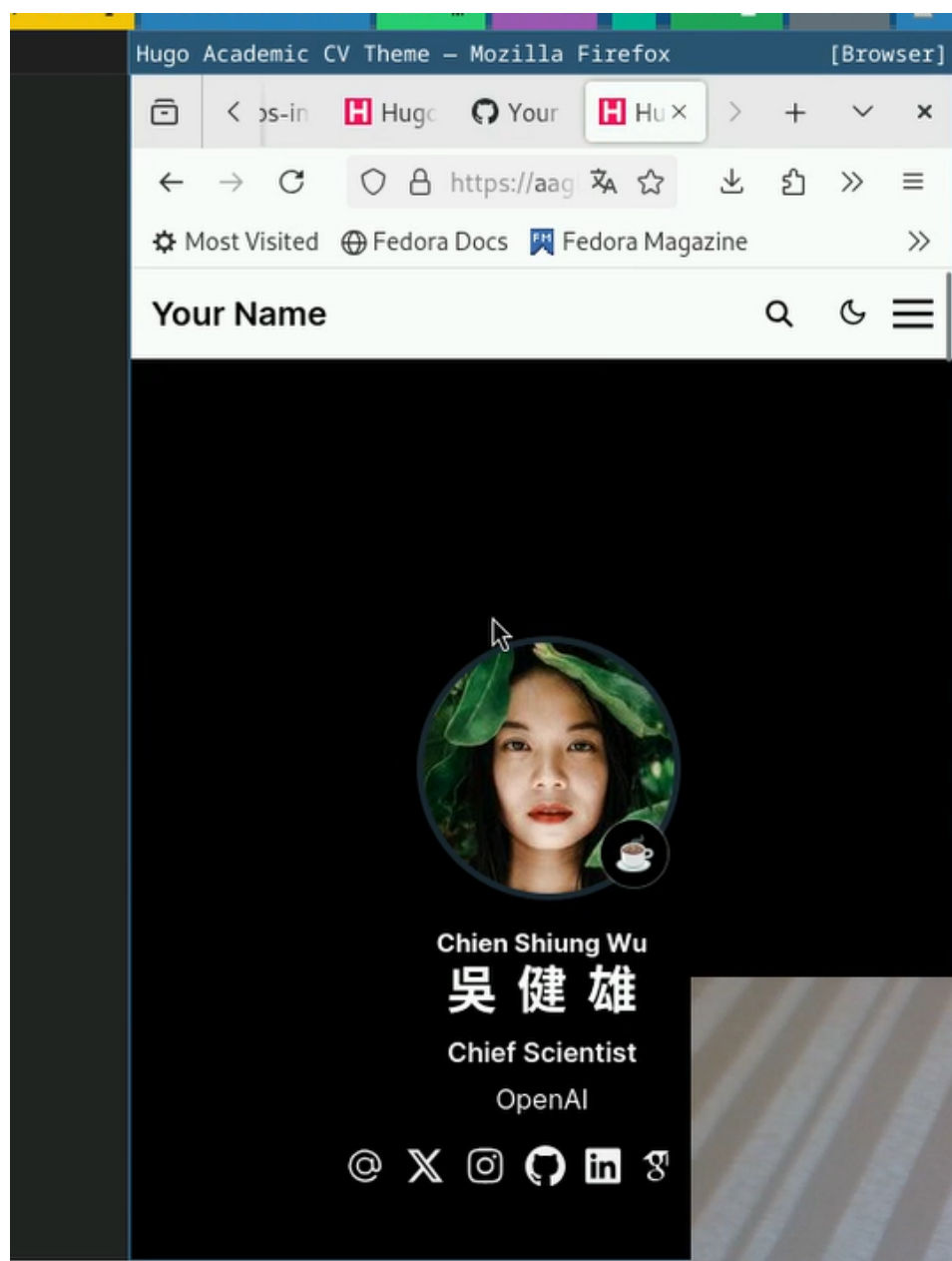


Рис. 3.20: Открытие сайта по ссылке

4 Выводы

В ходе выполнения первой стадии индивидуального проекта мне удалось разместить на Github pages заготовки для персонального сайта.

Список литературы