

# **Отчёт по первому этапу итогового проекта**

**Специальность: архитектура компьютеров**

Кудинец Максим Антонович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>4</b>	<b>Выводы</b>	<b>11</b>
	<b>Список литературы</b>	<b>12</b>

# Список иллюстраций

3.1	Установка go hugo . . . . .	7
3.2	Проверка версии . . . . .	7
3.3	Создание репозитория . . . . .	7
3.4	Загрузка пакетов данных . . . . .	8
3.5	Удаление каталога . . . . .	8
3.6	Создание нового репозитория . . . . .	9
3.7	Клонирование репозитория, проверка ветки . . . . .	9
3.8	Первый коммит . . . . .	9
3.9	Добавление подраздела . . . . .	10
3.10	Коммит подраздела . . . . .	10
3.11	Выгруженные файлы на github . . . . .	10

## **Список таблиц**

# 1 Цель работы

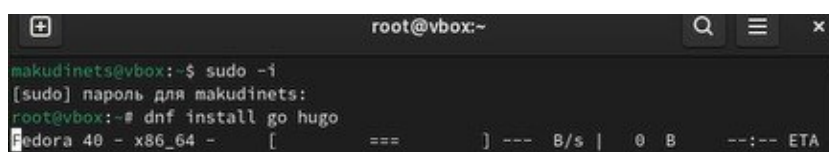
Размещение на Github pages заготовки для персонального сайта.

## 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 лабораторной работы

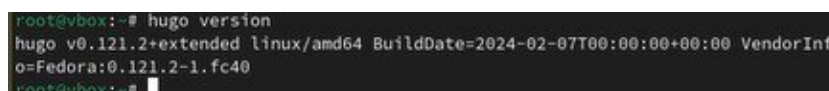
1. Переходим в режим суперпользователя и устанавливаем go hugo. (рис. 3.1).



```
root@vbox:~  
makudinets@vbox:~$ sudo -i  
[sudo] пароль для makudinets:  
root@vbox:~# dnf install go hugo  
Fedora 40 - x86_64 - [ === ] --- B/s | 0 B --:-- ETA
```

Рис. 3.1: Установка go hugo

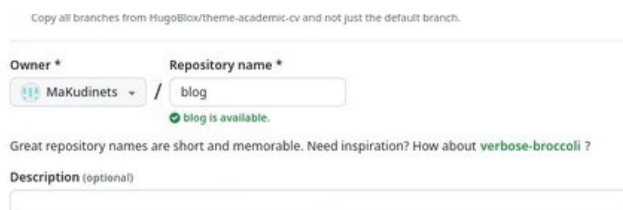
2. Проверяем версию hugo, если версия слишком низкая, устанавливаем новейшую. (рис. 3.2).



```
root@vbox:~# hugo version  
hugo v0.121.2+extended linux/amd64 BuildDate=2024-02-07T00:00:00+00:00 VendorInfo=Fedora:0.121.2-1.fc40  
root@vbox:~#
```

Рис. 3.2: Проверка версии

3. Создаем репозиторий на github. (рис. 3.3).



Copy all branches from HugoBlox/theme-academic-cv and not just the default branch.

Owner \*  / Repository name \*

✔ blog is available.

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

Description (optional)

Рис. 3.3: Создание репозитория

4. Скачиваем пакеты данных. (рис. 3.4).

```

makudinets@vbox:~$ cd ~/bin/blog
makudinets@vbox:~/bin/blog$ ~/bin/hugo
hugo: downloading modules ...

```

Рис. 3.4: Загрузка пакетов данных

5. Через `mc` находим папку `public` в необходимом каталоге и удаляем ее. (рис. 3.5).

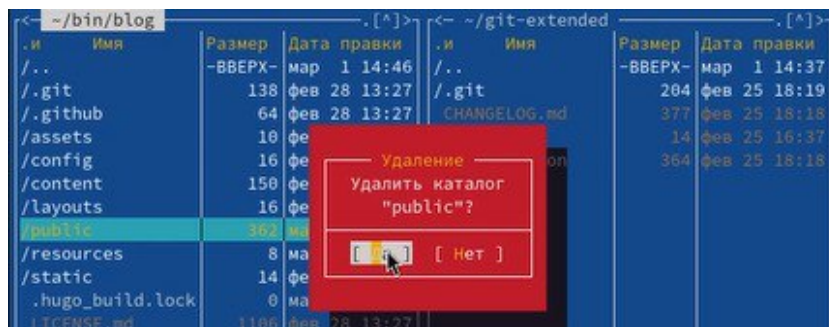


Рис. 3.5: Удаление каталога

6. Создаем новый репозиторий, (рис. 3.6). клонируем его и проверяем, на какой ветке мы сейчас находимся. (рис. 3.7).



**Create a new repository**

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (\*).

**Repository template**

No template

Start your repository with a template repository's contents.

**Owner \*** MaKudinets / **Repository name \*** github.io

github.io is available.

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

**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.

**Initialize this repository with:**

☒ Add a README file  
This is where you can write a long description for your project. [Learn more about READMEs](#).

**Add .gitignore**

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

**Choose a license**

Рис. 3.6: Создание нового репозитория

```
makudinets@vbox:~$ git clone --recursive git@github.com:MaKudinets/github.io.git
Клонирование в «github.io»...
warning: Похоже, что вы клонировали пустой репозиторий.
makudinets@vbox:~$ cd github.io
makudinets@vbox:~/github.io$ git checkout -b main
Переключились на новую ветку «main»
makudinets@vbox:~/github.io$
```

Рис. 3.7: Клонирование репозитория, проверка ветки

7. Делаем коммит в этот репозиторий. (рис. 3.8).

```
makudinets@vbox:~/github.io$ touch README.md
makudinets@vbox:~/github.io$ git add .
makudinets@vbox:~/github.io$ git commit "feat(main): add files"
error: pathspec 'feat(main): add files' did not match any file(s) known to git
makudinets@vbox:~/github.io$ git commit -am "feat(main): add files"
[main (корневой коммит) 1d2d47f] feat(main): add files
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
makudinets@vbox:~/github.io$ gti push
bash: gti: команда не найдена...
Аналогичная команда: 'git'
makudinets@vbox:~/github.io$ git push
```

Рис. 3.8: Первый коммит

8. Добавляем подраздел в ветку main. (рис. 3.9).

```

makudinets@vbox:~/bin/blog$ git submodule add -t main git@github.com:MaKudinets/
github.io.git public
использование: git submodule [--quiet] [--cached]
or: git submodule [--quiet] add [-b <branch>] [-f|--force] [--name <name>] [-
reference <repository>] [--] <repository> [<path>]
or: git submodule [--quiet] status [--cached] [--recursive] [--] [<path>...]
or: git submodule [--quiet] init [--] [<path>...]
or: git submodule [--quiet] deinit [-f|--force] [--all] [--] [<path>...]
or: git submodule [--quiet] update [--init [--filter=<filter-spec>]] [--remot
e] [-N|--no-fetch] [-f|--force] [--checkout|--merge|--rebase] [--[no-]recommen
d-shallow] [--reference <repository>] [--recursive] [--[no-]single-branch] [--] [<
path>...]
or: git submodule [--quiet] set-branch (--default|--branch <branch>) [--] <pa
th>
or: git submodule [--quiet] set-url [--] <path> <newurl>
or: git submodule [--quiet] summary [--cached|--files] [--summary-limit <n>]
[commit] [--] [<path>...]
or: git submodule [--quiet] foreach [--recursive] <command>
or: git submodule [--quiet] sync [--recursive] [--] [<path>...]
or: git submodule [--quiet] absorbgitdirs [--] [<path>...]
makudinets@vbox:~/bin/blog$

```

Рис. 3.9: Добавление подраздела

9. Делаем коммит подраздела на репозиторий. (рис. 3.10).

```

makudinets@vbox:~/bin/blog/public$ git remote
origin
makudinets@vbox:~/bin/blog/public$ git add .
makudinets@vbox:~/bin/blog/public$ git commit -am "feat(main): add files"

```

Рис. 3.10: Коммит подраздела

10. Проверяем наличие выложенных файлов на репозитории. (рис. 3.11).

File	Commit Message	Time
.github	Initial commit	yesterday
assets/media	Initial commit	yesterday
config/_default	Initial commit	yesterday
content	Initial commit	yesterday
layouts/partials/hooks/head-end	Initial commit	yesterday
public	feat(main): add files	now
static/uploads	Initial commit	yesterday
LICENSE.md	Initial commit	yesterday
README.md	Initial commit	yesterday
go.mod	Initial commit	yesterday
hugoblox.yaml	Initial commit	yesterday
netlify.toml	Initial commit	yesterday

Рис. 3.11: Выгруженные файлы на github

## 4 Выводы

Разместил на Github pages заготовки для персонального сайта.

## **Список литературы**