

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ  
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

**институт информационных технологий и технологического образования  
кафедра информационных технологий и электронного обучения**

Основная профессиональная образовательная программа  
Направление подготовки 09.03.01 Информатика и вычислительная техника  
Направленность (профиль) «Технологии разработки программного  
обеспечения»  
форма обучения – очная

**Курсовая работа**

по дисциплине «Организация электронной образовательной среды»  
Тема: «Разработка веб-портфолио студента с помощью генератора Pelican»

Обучающегося 4 курса ИВТ  
Афанасьева Андрея Дмитриевича

---

Руководитель:  
Старший преподаватель  
Аксютин Павел Александрович

« \_\_\_\_\_ » \_\_\_\_\_ 2022 г.

Санкт-Петербург  
2022

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
ГЛАВА 1	4
1.1 Понятие электронного портфолио	4
1.2 Pelican	5
1.3 Вывод	6
ГЛАВА 2	7
2.1 Установка Pelican и создание скелета портфолио	7
ЗАКЛЮЧЕНИЕ	12
ЛИТЕРАТУРА	13

## **ВВЕДЕНИЕ**

В современном мире очень много талантливых людей. Для того чтобы тебя заметили и оценили, одного умения, к сожалению, не достаточно. Нужно уметь показать себя, и свои работы в правильном свете.

В этом может помочь портфолио (итал. portfolio - портфель, папка для документов) - собрание образцов работ, фотографий, дающих представление о предлагаемых услугах специалиста.

Для студента портфолио ещё более важно, так как именно через него студент может показать, чему он научился за время обучения, и каких успехов достиг при этом. Кроме того, портфолио играет огромную роль при дальнейшей жизни, будь то трудоустройство, или получение ещё одного образования.

Актуальность данной курсовой работы обусловлена тем, что студенту необходимо уметь, хранить и структурировать информацию, а также, собирать её в удобное и понятное к просмотру веб-портфолио.

Цель курсовой работы: разработать веб-портфолио студента-программиста.

Задачи курсовой работы:

Изучить различную литературу и иные источники информации по теме исследования.

Раскрыть основные понятия и термины темы.

Воспользоваться полученными знаниями для реализации практической задачи.

Объект исследования: разработка веб-портфолио студента

## ГЛАВА 1

### 1.1 Понятие электронного портфолио

Портфолио — это собрание или комплект документов, образцов работ, которые студент собрал за время обучения. Содержание портфолио зависит от вида деятельности, программисты выкладывают в портфолио свои наработки в виде программного кода, фотографы загружают сделанные фотографии и так далее.

Главная задача портфолио — показать навыки, наработанный опыт и выполненные работы человека.

Портфолио дополняет резюме и показывает на наглядном примере, что умеет специалист. Ценность портфолио состоит в том, что оно наполняется годами.

#### **Преимущества портфолио:**

- позволяет наглядно продемонстрировать умения человека в самых разных областях;
- это весь комплекс знаний, навыков, умений и опыта, целостно и системно отражающих всю трудовую, учебную и иную деятельность;
- очень удобная форма хранения информации;
- позволяет работодателю оценить уровень профессиональных навыков.

Веб-портфолио — это веб-ресурс на котором находятся материалы, которые были получены в процессе обучения: результаты лабораторных работ, проектов, совместной деятельности. Веб-портфолио — буквально то же самое, что и обычное портфолио, но выполненное с использованием различных информационных технологий.

#### **Веб портфолио удобно тем, что:**

- его легче дополнить и усовершенствовать
- можно переслать в сети интернета в любую точку одним щелчком
- всегда можно взять отдельную часть и распечатать

- можно хранить на электронных носителях

### **Основные функции портфолио:**

- аналитическая - анализирует и обобщает работу специалиста;
- накопительная - отражает достижения работника (грамоты, дипломы, удостоверения, сертификаты и т. д.) и раскрывает спектр выполненных работ (проекты, исследования, и т. п.);
- модельная - отражает динамику профессионального развития, демонстрирует стиль работы, помогает спланировать деятельность;

### **1.2 Pelican**

Pelican - широко используемый генератор статических сайтов, написанный на Python. Pelican напрямую поддерживает ReStructuredText и может поддерживать Pelican, когда установлен требуемый пакет. Все задачи выполняются с помощью инструментов интерфейса командной строки (CLI), что делает его простым для всех, кто с ним знаком. Плюс его простой инструмент быстрого запуска делает создание веб-сайта чрезвычайно простым.

Кратко об устройстве:

-Контент создается в виде Markdown файлов. Файлы находятся в папке content, при этом, страницы будущего сайта хранятся во встроенной папке pages. Картинки для страниц лежат в папке images.

-Далее происходит обработка файлов в соответствии с выбранным шаблоном.

Плюсы данного генератора:

- Простота редактирования и обновления контента на сайте.
- Быстрая и удобная загрузка готовых страниц.
- Лёгкость создания новых тем.
- Возможность использовать бесплатные хостинги.
- Богатый набор готовых плагинов.

Минусы:

-Нет панели администрирования. Все действия выполняются посредством ввода команд в командной строке.

-Нет WYSIWYG-редактора так что для создания постов придется освоить Markdown.

### **1.3 Вывод**

Были рассмотрены виды портфолио, а так же определение данного термина. Так же был выбран основной инструмент для разработки своего портфолио, что и будет продемонстрировано в следующей части работы.

## ГЛАВА 2

### 2.1 Установка Pelican и создание скелета портфолио.

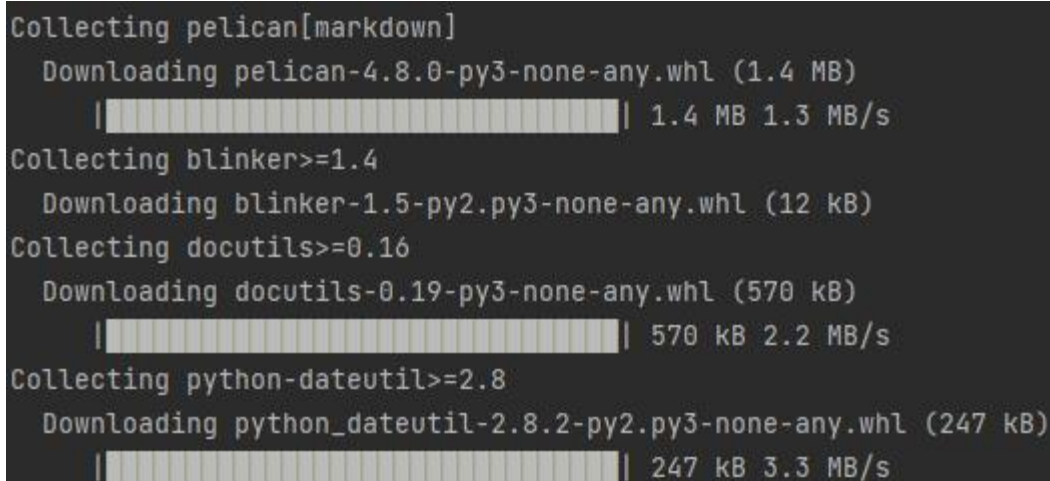
Установка Pelican довольно проста, и может быть выполнена одной командой(`python -m pip install pelican`), введенной в консоли PyCharm.



```
PS D:\KURSACH> python -m pip install pelican
```

Рис. 1

После ввода команды начнется установка, которую мы можем отслеживать в консоли.



```
Collecting pelican[markdown]
  Downloading pelican-4.8.0-py3-none-any.whl (1.4 MB)
    |████████████████████████████████████████| 1.4 MB 1.3 MB/s
Collecting blinker>=1.4
  Downloading blinker-1.5-py2.py3-none-any.whl (12 kB)
Collecting docutils>=0.16
  Downloading docutils-0.19-py3-none-any.whl (570 kB)
    |████████████████████████████████████████| 570 kB 2.2 MB/s
Collecting python-dateutil>=2.8
  Downloading python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
    |████████████████████████████████████████| 247 kB 3.3 MB/s
```

Рис 2.

После установки необходимо создать проект, в котором мы будем работать далее, и как только это будет сделано, можно приступать. Как упоминалось выше, в Pelican есть функция быстрого запуска, которая сильно облегчит нашу работу. Для её запуска, так же необходимо ввести команду.(`pelican-quickstart`)

```
PS D:\KURSACH> pelican-quickstart
Welcome to pelican-quickstart v4.8.0.

This script will help you create a new Pelican-based website.

Please answer the following questions so this script can generate the files
needed by Pelican.

> Where do you want to create your new web site? [.]
```

Рис. 3

Для дальнейшей работы, необходимо ответить на вопросы, которые нам задаст Pelican. Что бы ответить, надо напечатать ответ все так же в консоли.

```
> Where do you want to create your new web site? [.]
> What will be the title of this web site? Портфолио Студента
> Who will be the author of this web site? Афанасьев А.Д.
> What will be the default language of this web site? [Russian]
Please enter a valid time zone:
(check [https://en.wikipedia.org/wiki/List\_of\_tz\_database\_time\_zones])
> What is your time zone? [Europe/Rome] Europe/Moscow
> Do you want to generate a tasks.py/Makefile to automate generation and publishing? (Y/n) y
> Do you want to upload your website using FTP? (Y/N) n
> Do you want to upload your website using SSH? (Y/N) n
> Do you want to upload your website using Dropbox? (Y/N) n
> Do you want to upload your website using S3? (Y/N) n
> Do you want to upload your website using Rackspace Cloud Files? (Y/N) n
> Do you want to upload your website using GitHub Pages? (Y/N) y
> Is this your personal page (username.github.io)? (Y/N) y
```

Рис. 4

По завершению всех ответов, нам сообщат об итогах.

```
Done. Your new project is available at D:\KURSACH
```

Рис. 5

Как видно, все удалось, и наш проект был создан успешно. Далее, переходим к папке Content, и создаем там папку Pages. именно в ней будут находиться страницы нашего будущего портфолио. Для проверки, создадим тестовую страницу Home.md, и заполним её минимальным контентом.



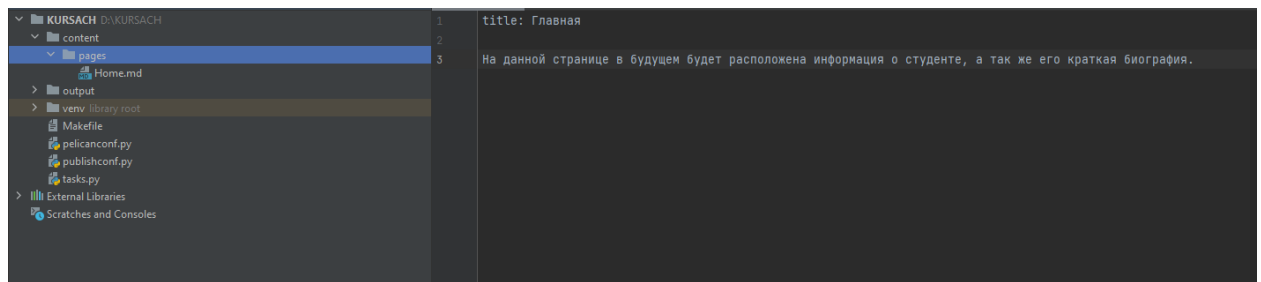


Рис. 6

Осталось подтвердить изменения, и посмотреть как выглядит наш сайт на данный момент.

```
PS D:\KURSACH> pelican content
[15:32:28] WARNING Docutils has no localization for 'russian'. Using 'en' instead.
WARNING Watched path does not exist: D:\KURSACH\content\images
Done: Processed 0 articles, 0 drafts, 0 hidden articles, 1 page, 0 hidden pages and 0 draft pages in 0.16 seconds.
PS D:\KURSACH> pelican --listen
Serving site at: http://127.0.0.1:8000 - Tap CTRL-C to stop
```

Рис. 7

Для сохранения изменений используем команду `pelican content`, а для запуска сайта `pelican --listen`. В итоге, получаем следующее:

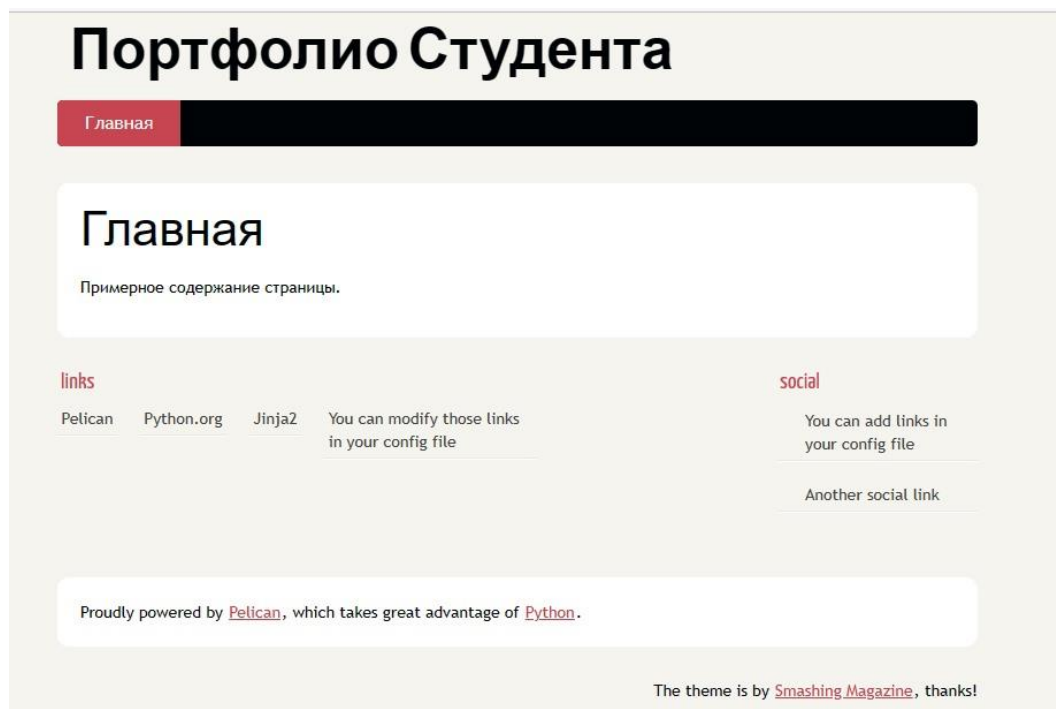


Рис. 8

Как видно, все удалось. Мы смогли создать тестовую страницу, и она успешно отображается в браузере. Осталось наполнить её полноценными работами, а так же связать наш сайт с каким либо хостингом. В данном случае, будет использоваться GitHub Pages.

Для наполнения были взяты работы по дисциплине “Веб-Технологии” и дисциплине “Прикладное IT”. Данный выбор обусловлен тем, что для данных работ уже есть готовые облачные хранилища.

Что бы все это сделать, необходимо создать два новых файла в папке Pages, и назвать их соответственно дисциплинам.

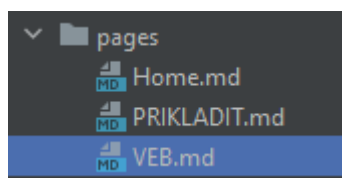


Рис. 9

После чего расположить на них ссылки на выполненные работы, а также сделать разметку путем использования Markdown. И последнее, что необходимо сделать, создать репозиторий на GitHub, куда выгрузить папку Output из нашего проекта. Что бы сделать это, просто идем туда куда сохранили наш проект при создании, и там находим нужную папку.

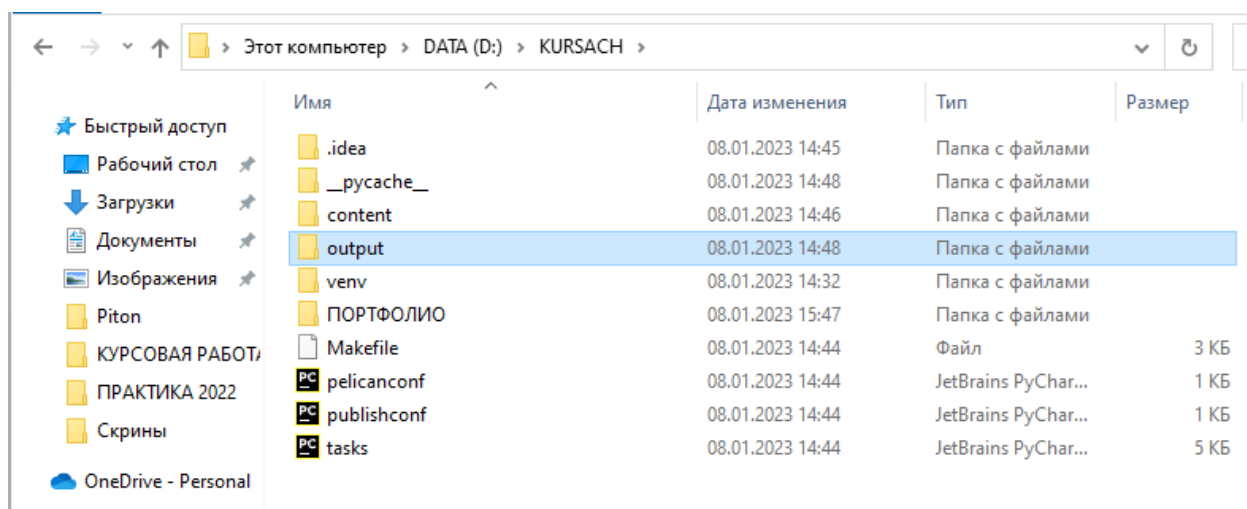


Рис. 10

В итоге, наши сайт обновился, и мы можем посмотреть работы по указанным выше дисциплинам.

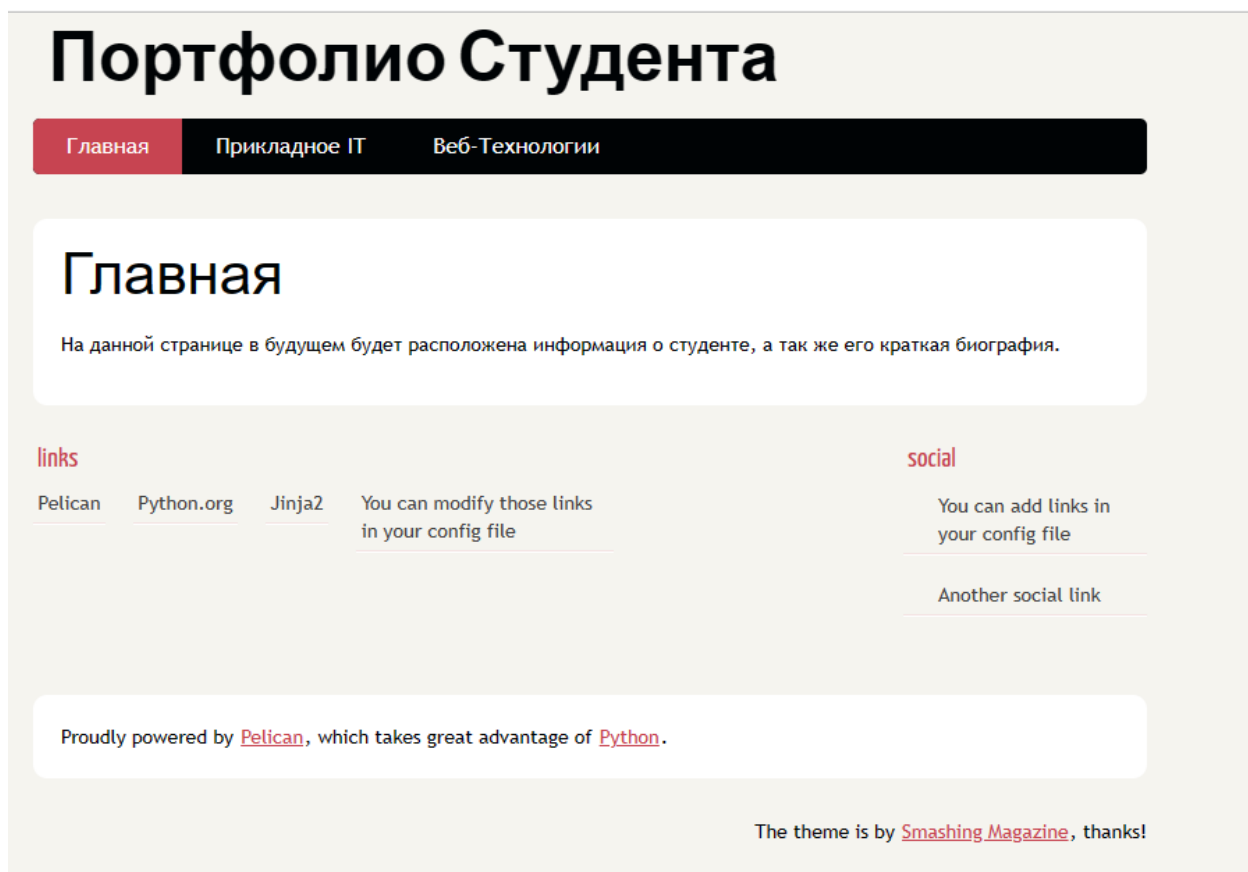


Рис. 11

На основе полученных знаний портфолио, была создана электронная версия портфолио студента.

Его создание включало в себя следующие этапы: сбор подготовка инструмента; создание структуры портфолио, размещение в сети.

## **ЗАКЛЮЧЕНИЕ**

В ходе курсовой работы было разработано веб-портфолио студента, сама разработка была продемонстрирована шаг за шагом, при помощи текста и скриншотами выполненной работы. Также был рассмотрен теоретический материал по теме, были рассмотрены основные функции портфолио, его преимущества и удобства.

Так же было проведено знакомство с генератором статических сайтов «Pelican». Благодаря данному генератору, человек, который далек от программирования, может даже сам, при небольших исследованиях создать свое собственное веб-портфолио.

## **ЛИТЕРАТУРА**

- 1)Шакон С. Pro Git: учебное пособие —2012.
- 2)Чакон С., Штрауб Б. Git для профессионального программиста. —  
Питер, 2017
- 3)Локк Д. Основы управления проектами: учебное пособие. —2004
- 4)Иванова Г.С. Технология программирования: учебное пособие. —2016