

# **Индивидуальный проект 2 этап**

**Операционные системы**

Кочарян Никита Робертович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
<b>5</b>	<b>Выводы</b>	<b>13</b>
	<b>Список литературы</b>	<b>14</b>

## Список иллюстраций

4.1	замена фото и краткого описания . . . . .	8
4.2	добавляю информацию об интересах и образовании . . . . .	9
4.3	пост о прошедшей неделе . . . . .	10
4.4	пост об управлении версиями Git . . . . .	11
4.5	пост об управлении версиями Git(2) . . . . .	12

# Список таблиц

3.1	Описание некоторых каталогов файловой системы GNU Linux . .	7
-----	---	---

# 1 Цель работы

Добавить к сайту данные о себе.

## 2 Задание

Разместить фотографию владельца сайта. Разместить краткое описание владельца сайта (Biography). Добавить информацию об интересах (Interests). Добавить информацию об образовании (Education). Сделать пост по прошедшей неделе. Добавить пост на тему по выбору: Управление версиями. Git. Непрерывная интеграция и непрерывное развертывание (CI/CD).

### 3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы. Например, в табл. 3.1 приведено краткое описание стандартных каталогов Unix.

Таблица 3.1: Описание некоторых каталогов файловой системы GNU Linux

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя
/media	Точки монтирования для сменных носителей
/root	Домашняя директория пользователя root
/tmp	Временные файлы
/usr	Вторичная иерархия для данных пользователя

Более подробно об Unix см. в [1–6].

## 4 Выполнение лабораторной работы

1. С помощью редактирования файла `avatar.jpg`, находящимся в директории `~/work/study/blog/content/authors/admin` меняю фото на свое; редактирую файл `index` в той же директории меняю краткое описание владельца сайта



Рис. 4.1: замена фото и краткого описания

2. Редактируя тот же файл `index` добавляю информацию об интересах и образовании



```
# Interests to show in About widget
interests:
  - python/c++ programming
  - learning foreign languages
  - travelling

# Education to show in About widget
education:
  courses:
    - course: business infromatics
      institution: People's Friendship University
      year: 2022
```

Рис. 4.2: добавляю информацию об интересах и образовании

3. Редактируя файл `index` в директории `~/work/study/blog/content/post/getting-started` создаю пост о своей прошедшей неделе

# Всем привет я Никита и я расскажу о моей прошлой неделе

Nikita Kocharyan

Mar 17, 2023 · 1 min read · Demo



## Кратко о моей неделе

1. В среду я прошел ennagram test и его результаты меня разочаровали! Там было написано что я добрячок и не токсик, а это неправда.
2. В четверг я провел в вузе 11 часов, но вообще не устал, а ещё сходил на шоу голос и мне очень понарвилось.
3. Сегодня я пойду вкусно кушать 😊

fishing

PFUR

Hunting



Рис. 4.3: пост о прошедшей неделе

4. Редактируя файл index в директории ~/work/study/blog/content/post/writing technical content создаю пост об управлении версиями Git.

# Управление версиями Git

Mar 17, 2023 · 3 min read



На этой странице вы найдете полезную информацию о VCS

Система контроля версий (Version Control System, VCS) представляет собой программное обеспечение, которое позволяет отслеживать изменения в документах, при необходимости производить их откат, определять, кто и когда внес исправления и т.п. В статье рассмотрены виды VCS, принципы их работы, а также приведены примеры программных продуктов.

## Локальные системы контроля версий

Многие люди в качестве метода контроля версий применяют копирование файлов в отдельный каталог (возможно даже, каталог с отметкой по времени, если они достаточно сообразительны). Данный подход очень распространён из-за его простоты, однако он невероятно сильно подвержен появлению ошибок. Можно легко забыть в каком каталоге вы находитесь и случайно изменить не тот файл или скопировать не те файлы, которые вы хотели. Для того, чтобы решить эту проблему, программисты давным-давно разработали локальные VCS с простой базой данных, которая хранит записи о всех изменениях в файлах, осуществляя тем самым контроль ревизий.

## Централизованные системы контроля версий

Следующая серьёзная проблема, с которой сталкиваются люди, — это

Рис. 4.4: пост об управлении версиями Git

## Распределённые системы контроля версий

Здесь в игру вступают распределённые системы контроля версий (Distributed Version Control System, далее DVCS). В DVCS (таких как Git, Mercurial, Bazaar или Darcs) клиенты не просто скачивают снимок всех файлов (состояние файлов на определённый момент времени) — они полностью копируют репозиторий. В этом случае, если один из серверов, через который разработчики обменивались данными, умрёт, любой клиентский репозиторий может быть скопирован на другой сервер для продолжения работы. Каждая копия репозитория является полным бэкапом всех данных. Более того, многие DVCS могут одновременно взаимодействовать с несколькими удалёнными репозиториями, благодаря этому вы можете работать с различными группами людей, применяя различные подходы одновременно в рамках одного проекта. Это позволяет применять сразу несколько подходов в разработке, например, иерархические модели, что совершенно невозможно в централизованных системах.

### Git

Git – распределенная система контроля версий, разработанная Линусом Торвальдсем для работы над ядром операционной системы Linux. Среди крупных проектов, в рамках которых используется git, можно выделить ядро Linux, Qt, Android. Git свободен и распространяется под лицензией GNU GPL 2 и, также как Mercurial, доступен практически на всех операционных системах. По своим базовым возможностям git схож с Mercurial (и другими DVCS), но благодаря ряду достоинств (высокая скорость работы, возможность интеграции с другими VCS, удобный интерфейс) и очень активному сообществу, сформировавшемуся вокруг этой системы, git вышел в лидеры рынка распределенных систем контроля версий.

**Did you find this page helpful? Consider sharing it**



Рис. 4.5: пост об управлении версиями Git(2)

## 5 Выводы

Благодаря данной лабораторной работе я научился редактировать сайт, добавлять посты и вставлять в них картинки.

## Список литературы

1. GNU Bash Manual [Электронный ресурс]. Free Software Foundation, 2016.  
URL: <https://www.gnu.org/software/bash/manual/>.
2. Newham C. Learning the bash Shell: Unix Shell Programming. O'Reilly Media, 2005. 354 с.
3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
4. Robbins A. Bash Pocket Reference. O'Reilly Media, 2016. 156 с.
5. Таненбаум Э. Архитектура компьютера. 6-е изд. СПб.: Питер, 2013. 874 с.
6. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.