

# **Отчёт по лабораторной работе № 3**

**Операционные системы**

Паулу Антонью Жоау

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
3.1	Структурная составляющая отчета . . . . .	6
3.2	Техническая составляющая отчета . . . . .	8
<b>4</b>	<b>Выводы</b>	<b>13</b>

## Список иллюстраций

3.1	Титульный лист в md . . . . .	6
3.2	Цель работы . . . . .	6
3.3	Задачи работы . . . . .	7
3.4	Выполнение лабораторной работы . . . . .	7
3.5	Вывод . . . . .	7
3.6	Ответы на контрольные вопросы . . . . .	8
3.7	Pandoc . . . . .	8
3.8	Pandoc . . . . .	9
3.9	Заголовок первого уровня . . . . .	9
3.10	Заголовки второго уровня . . . . .	10
3.11	Прикрепление изображения . . . . .	10
3.12	Папка image . . . . .	11
3.13	Нумерованные изображения . . . . .	11
3.14	Каталог отчета . . . . .	11
3.15	make . . . . .	11
3.16	Отчёт в формате pdf, docx . . . . .	12

# 1 Цель работы

Научиться оформлять отчёты с помощью легковесного языка разметки Markdown.

## 2 Задание

– Сделать отчёт по предыдущей лабораторной работе в формате Markdown. –  
В качестве отчёта предоставить отчёты в 3 форматах: pdf, docx и md (в архиве, поскольку он должен содержать скриншоты, Makefile и т.д.)

## 3 Выполнение лабораторной работы

### 3.1 Структурная составляющая отчета

Оформили титульный лист:(рис. [3.1])

```
---  
## Front matter  
title: "Отчёт по лабораторной работе № 2"  
subtitle: "Операционные системы"  
author: "Паулу Антонию Жоау"
```

Рис. 3.1: Титульный лист в md

Обозначили цель лабораторной работы: (рис. [3.2])

```
---  
  
# Цель работы  
- Изучить идеологию и применение средств контроля версий.  
- Освоить умения по работе с git.
```

Рис. 3.2: Цель работы

Поставили задачи, которые необходимо выполнить в ходе лабораторной работы.  
(рис. [3.3])

### # Задание

– Установить и настроить ПО для работы с `git`.

Рис. 3.3: Задачи работы

В разделе “Выполнение лабораторной работы” подробно описали операции, реализуемые в ходе описываемой работы. (рис. [3.4])

```
# Выполнение лабораторной работы

## Установка программного обеспечения

Установили git: (рис. [-@fig:001])

{ #fig:001 width=70%}

Установили gh: (рис. [-@fig:002])

{ #fig:002 width=70%}

## Базовая настройка git

Задали имя и email владельца репозитория: (рис. [-@fig:003])

{ #fig:003 width=70%}

Настроили utf-8 в выводе сообщений git: (рис. [-@fig:004])

{ #fig:004 width=70%}

Настроили верификацию и подписание коммитов git.
Задали имя начальной ветки (будем называть её master). (рис. [-@fig:005])

{ #fig:005 width=70%}

Параметр autocrlf: (рис. [-@fig:006])

{ #fig:006 width=70%}
```

Рис. 3.4: Выполнение лабораторной работы

Подвели итоги выполненной лабораторной работы. (рис. [3.5])

### # Выводы

В ходе выполнения данной лабораторной работы была изучена идеология и применение средств контроля версий и освоены умения по работе с `git`.

Рис. 3.5: Вывод

В конце лаболатонной работы ответили на контрольные вопросы. (рис. [3.6])

**# Ответы на контрольные вопросы**

**1.** Что такое системы контроля версий (VCS) и для решения каких задач они предназначаются? Система управления версиями (также используется определение «система контроля версий», от англ. *Version Control System*, VCS или *Revision Control System*) – программное обеспечение для облегчения работы с изменяющейся информацией. Система управления версиями позволяет хранить несколько версий одного и того же документа, при необходимости возвращаться к более ранним версиям, определять, кто и когда сделал то или иное изменение, и многое другое.

**2.** Объясните следующие понятия VCS и их отношения: хранилище, *commit*, история, рабочая копия.  
 Хранилище (*repository*), или *депозитарий*, – место хранения файлов и их версий, служебной информации.  
 Версия (*revision*), или ревизия, – состояние всего хранилища или отдельных файлов в момент времени («пункт истории»)  
*Commit* («трудовой вклад», не переводится) – процесс создания новой версии; иногда синоним версии.  
 Рабочая копия (*working copy*) – текущее состояние файлов проекта (любой версии), полученных из хранилища и, возможно, измененных.

**3.** Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.  
 Децентрализованные VCS:  
 У каждого пользователя свой вариант (возможно не

Рис. 3.6: Ответы на контрольные вопросы

## 3.2 Техническая составляющая отчета

Для обработки файлов в формате Markdown использовали Pandoc. (рис. [3.7], рис. [3.8])

```

XXXXXXXXXXXXX XXXXXXXXXXXXX
sansfontoptions: Ligatures=TeX,Scale=MatchLowercase
monofontoptions: Scale=MatchLowercase,Scale=0.9
## Biblatex
biblatex: true
biblio-style: "gost-numeric"
biblatexoptions:
- parenttracker=true
- backend=biber
- hyperref=auto
- language=auto
- autolang=other*
- citestyle=gost-numeric
## Pandoc-crossref LaTeX customization
figureTitle: "Рис."
tableTitle: "Таблица"
listingTitle: "Листинг"
lofTitle: "Список иллюстраций"
lolTitle: "Листинги"
## Misc options
indent: true
header-includes:
- \usepackage[indentfirst]
- \usepackage{float} # keep figures where there are in the text
- \floatplacement{figure}{H} # keep figures where there are in the text
---
```

Рис. 3.7: Pandoc



```

---
## Front matter
title: "Отчёт по лабораторной работе № 2"
subtitle: "Операционные системы"
author: "Паулу Антонию Жоау"

## Generic options
lang: ru-RU
toc-title: "Содержание"

## Bibliography
bibliography: bib/cite.bib
csl: pandoc/csl/gost-r-7-0-5-2008-numeric.csl

## Pdf output format
toc: true # Table of contents
toc-depth: 2
lof: true # List of figures
fontsize: 12pt
linestretch: 1.5
papersize: a4
documentclass: scrreprt
## I18n polyglossia
polyglossia-lang:
  name: russian
  options:

```

Рис. 3.8: Pandoc

Разделы “Цель работы”, “Задание”, “Выполнение лабораторной работы”, “Выводы”, “Ответы на контрольные вопросы” были отмечены как заголовки первого уровня (#) (рис. [3.9]), а подразделы Выполнения лабораторной работы - как заголовки второго уровня (##).(рис. [3.10])

```

# Задание
- Установить и настроить ПО для работы с git.

```

Рис. 3.9: Заголовок первого уровня

```

# Выполнение лабораторной работы

## Установка программного обеспечения

Установили git: (рис. [-@fig:001])

 { #fig:001 width=70%}

Установили gh: (рис. [-@fig:002])

 { #fig:002 width=70%}

## Базовая настройка git

Задали имя и email владельца репозитория: (рис. [-@fig:003])

 { #fig:003 width=70%}

Настроили utf-8 в выводе сообщений git: (рис. [-@fig:004])

 { #fig:004 width=70%}

Настроили верификацию и подписание коммитов git.
Задали имя начальной ветки (будем называть её master). (рис. [-@fig:005])

 { #fig:005 width=70%}

Параметр autocrlf: (рис. [-@fig:006])

 { #fig:006 width=70%}

```

Рис. 3.10: Заголовки второго уровня

Ссылка на изображение и его подпись. (рис. [3.11])

```

Установили gh: (рис. [-@fig:002])

 { #fig:002 width=70%}

```

Рис. 3.11: Прикрепление изображения

Изображения располагаются в папке image, каталога с отчетом лабораторной № 2. (рис [3.12], [3.13])

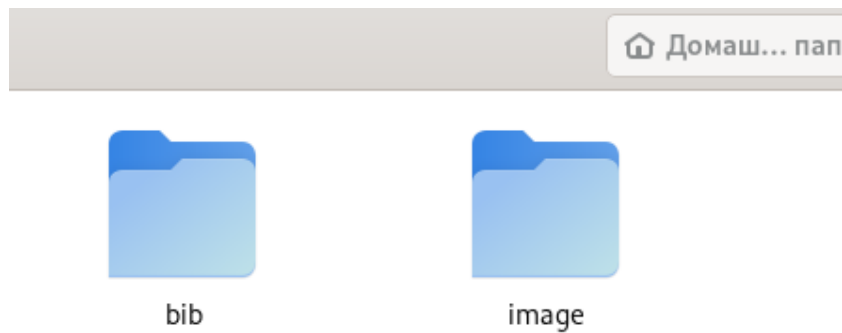


Рис. 3.12: Папка image



Рис. 3.13: Нумерованные изображения

Перешли в каталог отчета лабораторной работы № 2. (рис. [3.14])

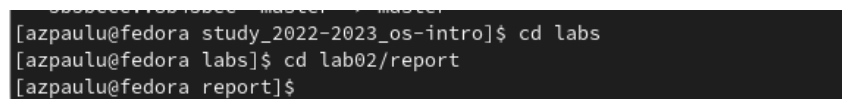


Рис. 3.14: Каталог отчета

С помощью команды make создали отчёт в формате pdf, docx. (рис. [3.15], [3.16])

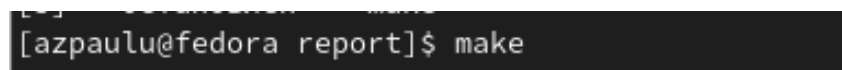


Рис. 3.15: make

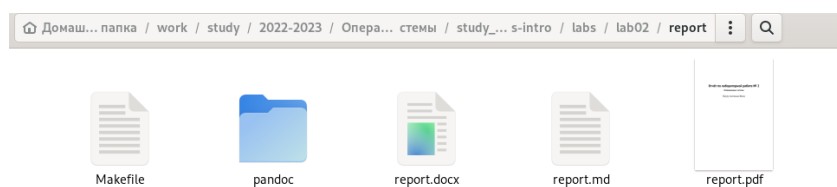


Рис. 3.16: Отчёт в формате pdf, docx

## 4 Выводы

В ходе выполнения лабораторной работы были изучены способы оформления отчётов с помощью легковесного языка разметки Markdown.