

Отчёт по лабораторной работе 3

Архитектура компьютеров

Шамес Эддин Хамза НКА-06-24

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Знакомство с Markdown	6
2.2	Задание для самостоятельной работы	11
3	Выводы	14

Список иллюстраций

2.1	Компиляция шаблона с использованием Makefile	6
2.2	Файл report.docx	7
2.3	Файл report.pdf	8
2.4	Удаление файлов report.docx и report.pdf	9
2.5	Открытие шаблона отчета	10
2.6	Заполнение шаблона отчета	11
2.7	Заполнение шаблона для второй лабораторной работы	12
2.8	Экспорт отчетных файлов	13

Список таблиц

1 Цель работы

Целью работы является освоение процедуры оформления отчетов с помощью легковесного языка разметки Markdown.

2 Выполнение лабораторной работы

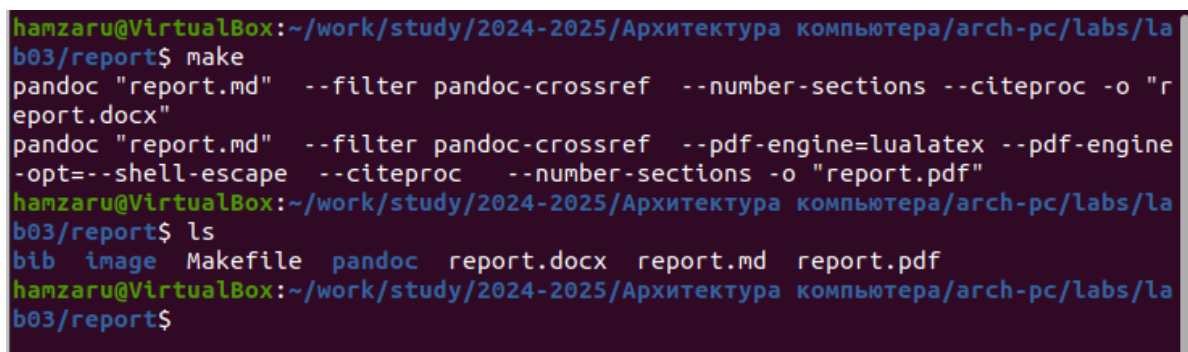
2.1 Знакомство с Markdown

В процессе выполнения лабораторной работы установил необходимые программы: pandoc и TexLive, как было указано в инструкциях.

Перешел в каталог курса, созданный в ходе выполнения лабораторной работы №3, затем в директорию с шаблоном отчета по лабораторной работе №3.

Для компиляции шаблона использовал Makefile, выполнив команду make. (рис. 2.1)

Создание PDF-файла с первого раза не удалось, так как потребовалась установка дополнительных шрифтов.



```
hamzaru@VirtualBox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab03/report$ make
pandoc "report.md" --filter pandoc-crossref --number-sections --citeproc -o "report.docx"
pandoc "report.md" --filter pandoc-crossref --pdf-engine=lualatex --pdf-engine-opt=--shell-escape --citeproc --number-sections -o "report.pdf"
hamzaru@VirtualBox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab03/report$ ls
bib image Makefile pandoc report.docx report.md report.pdf
hamzaru@VirtualBox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab03/report$
```

Рис. 2.1: Компиляция шаблона с использованием Makefile

После успешной компиляции были сгенерированы файлы report.pdf (рис. 2.3) и report.docx (рис. 2.2), что я проверил.

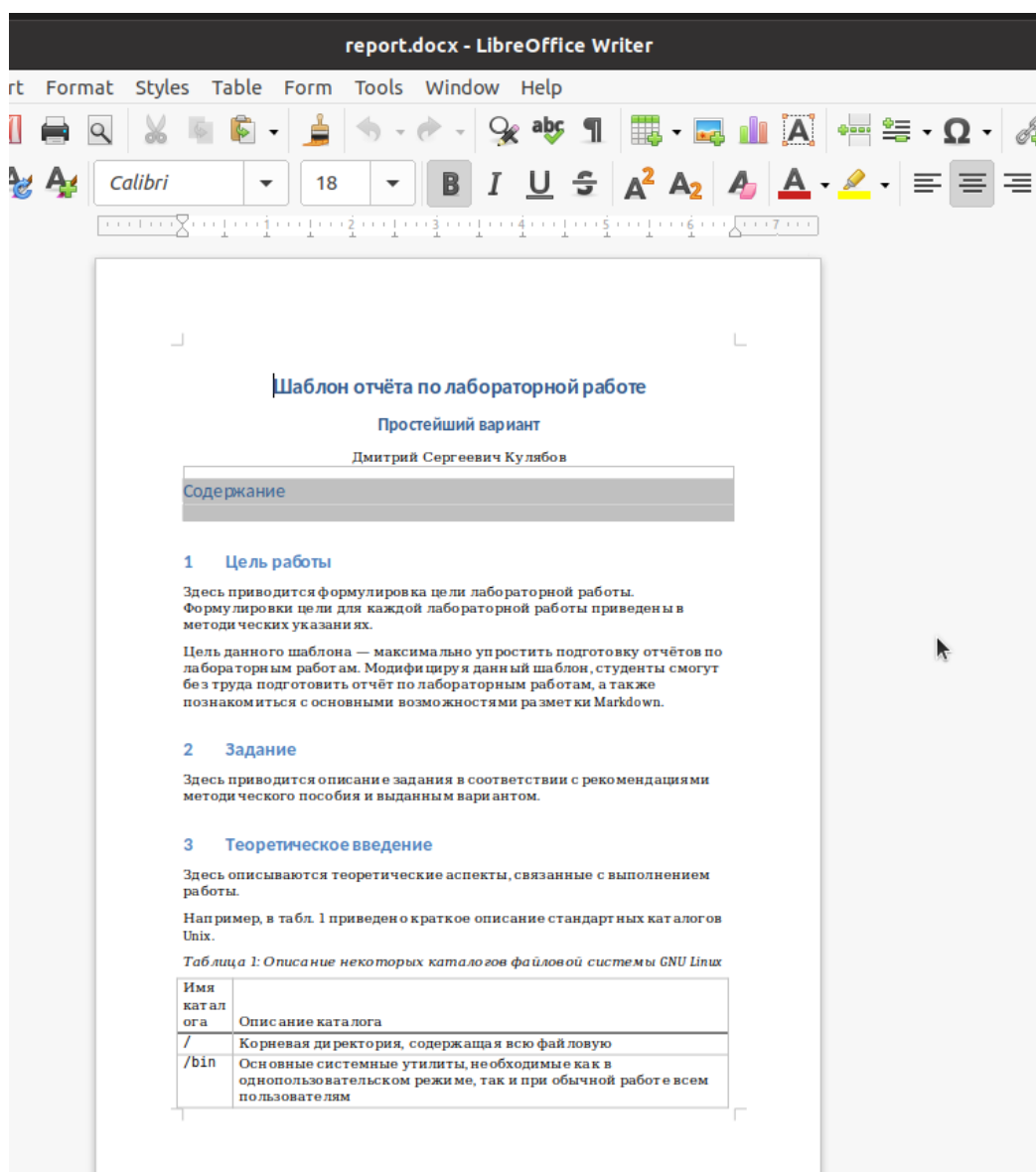


Рис. 2.2: Файл report.docx

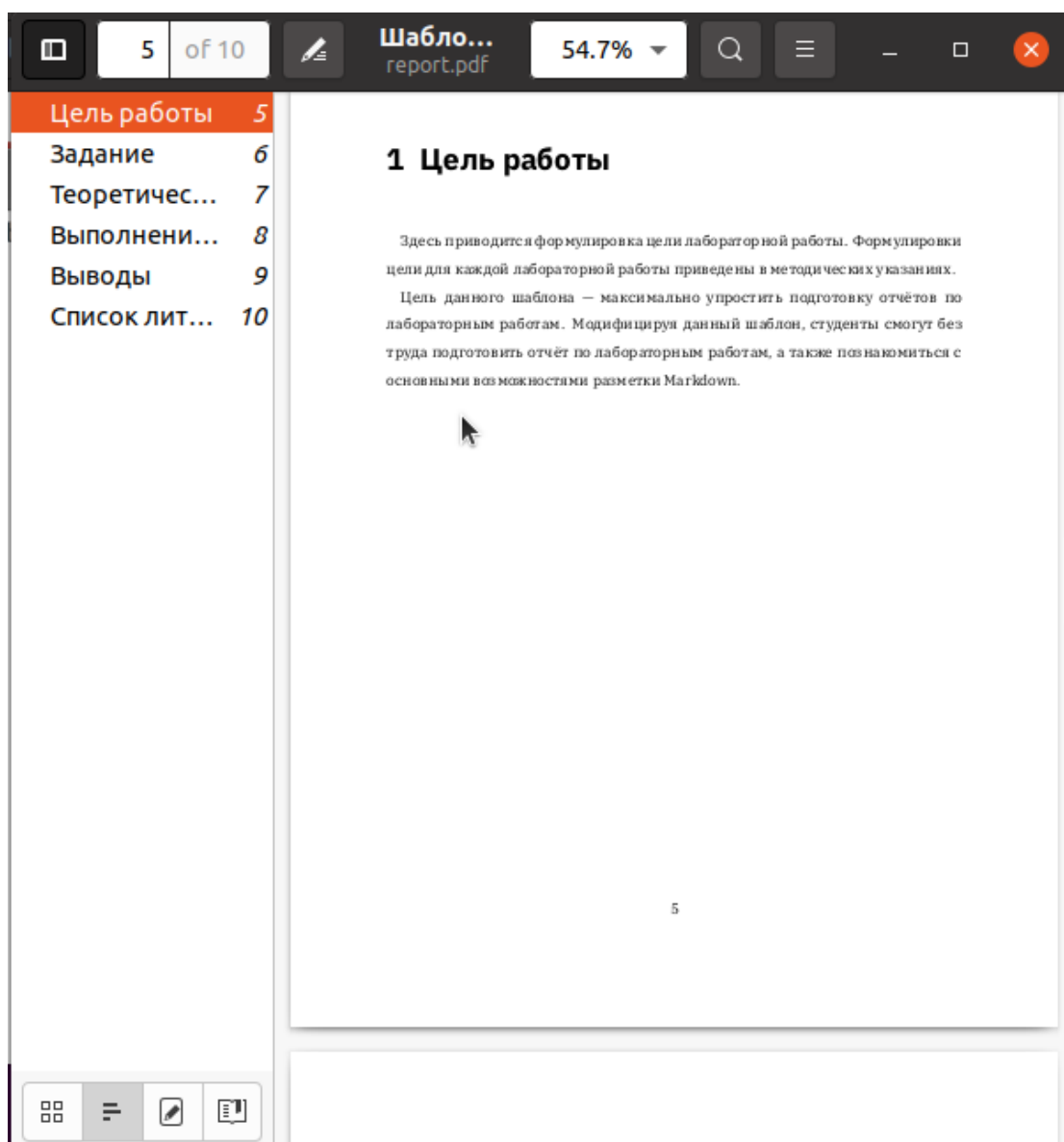


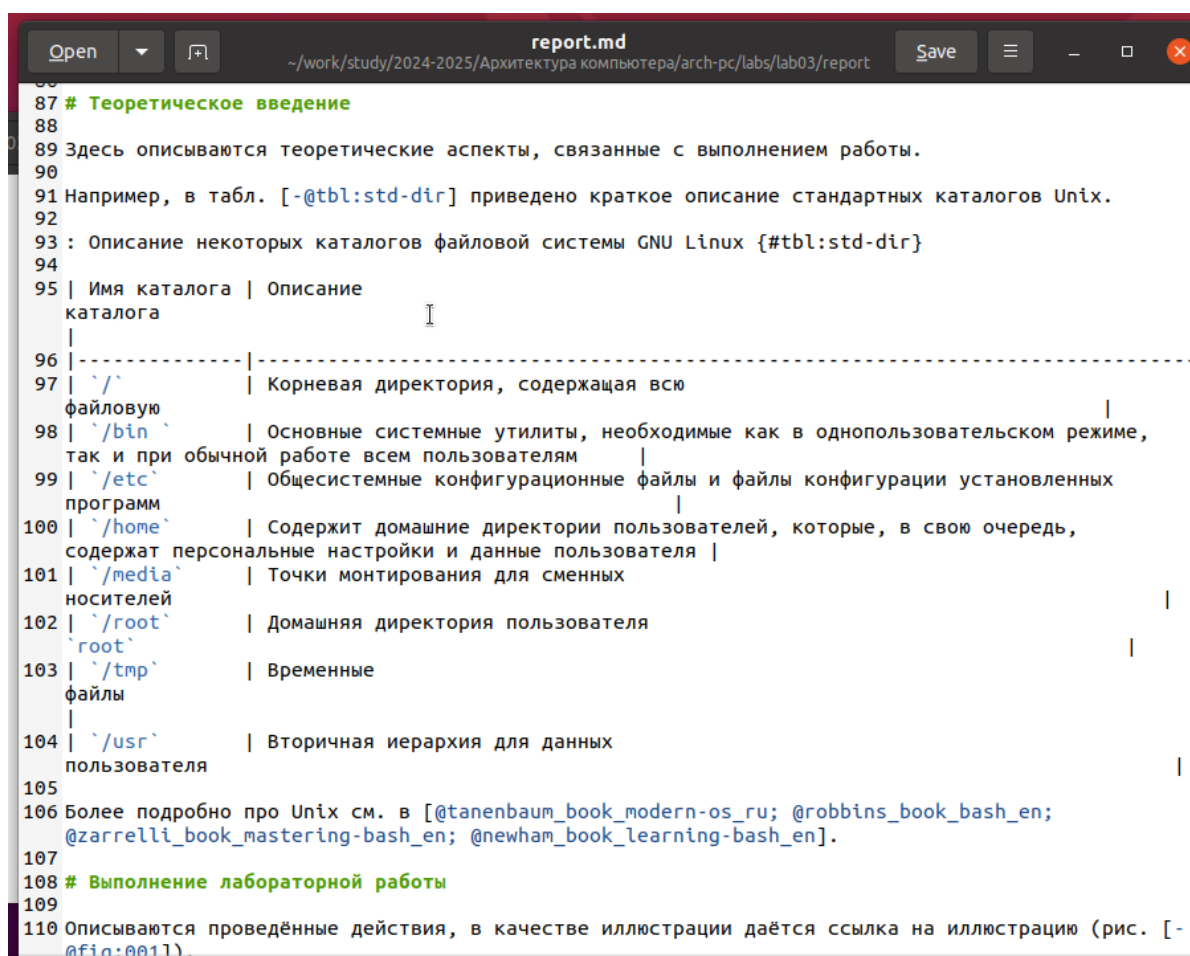
Рис. 2.3: Файл report.pdf

Затем с помощью команды `make clean` удалил сгенерированные файлы. После выполнения команды убедился, что файлы `report.pdf` и `report.docx` были удалены. (рис. 2.4)


```
hamzaru@VirtualBox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab03/report$ make
pandoc "report.md" --filter pandoc-crossref --number-sections --citeproc -o "report.docx"
pandoc "report.md" --filter pandoc-crossref --pdf-engine=lualatex --pdf-engine-opt=--shell-escape --citeproc --number-sections -o "report.pdf"
hamzaru@VirtualBox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab03/report$ ls
bib image Makefile pandoc report.docx report.md report.pdf
hamzaru@VirtualBox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab03/report$
hamzaru@VirtualBox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab03/report$ make clean
rm report.docx report.pdf *~
rm: cannot remove '*~': No such file or directory
make: [Makefile:35: clean] Error 1 (ignored)
hamzaru@VirtualBox:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab03/report$
```

Рис. 2.4: Удаление файлов report.docx и report.pdf

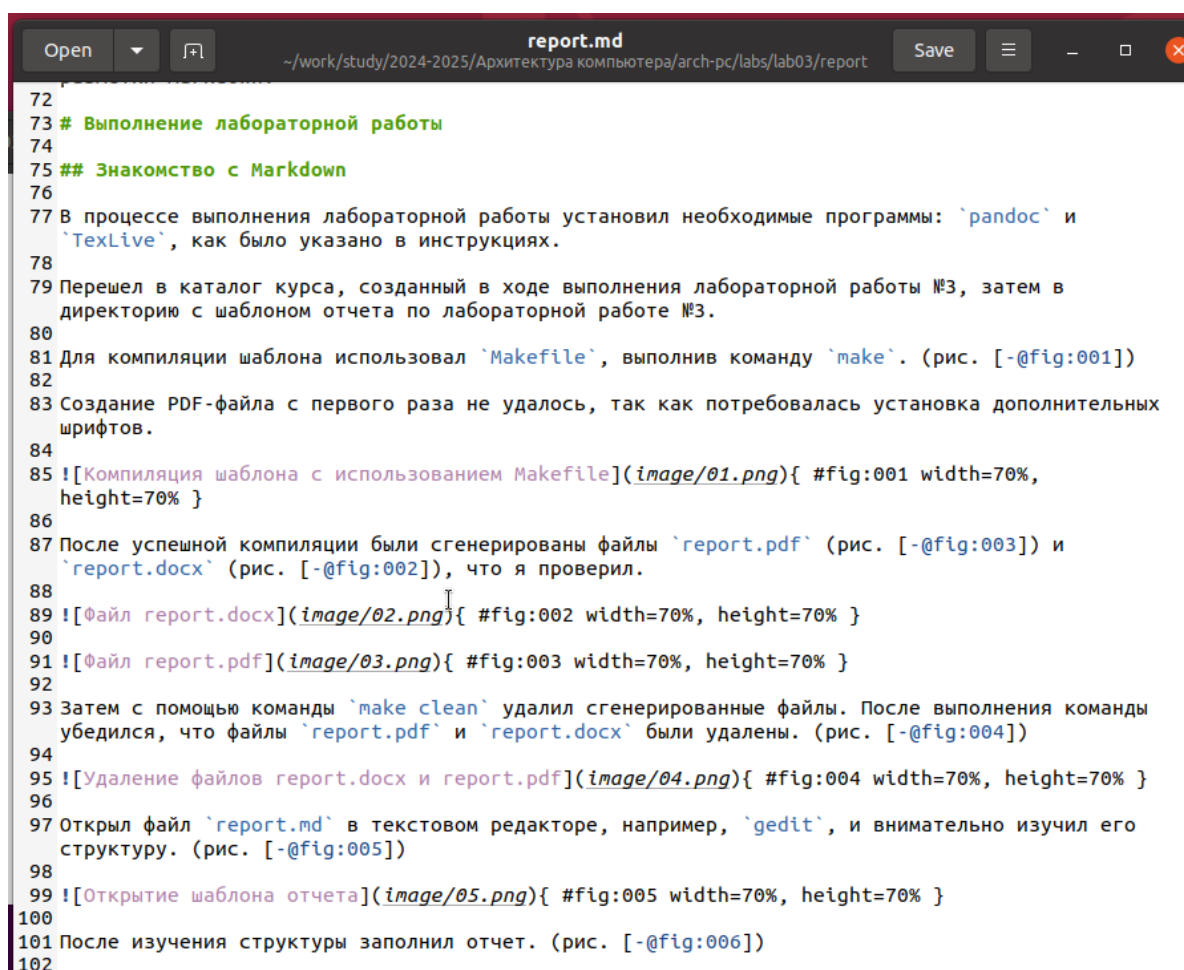
Открыл файл report.md в текстовом редакторе, например, gedit, и внимательно изучил его структуру. (рис. 2.5)



```
87 # Теоретическое введение
88
89 Здесь описываются теоретические аспекты, связанные с выполнением работы.
90
91 Например, в табл. [-@tbl:std-dir] приведено краткое описание стандартных каталогов Unix.
92
93 : Описание некоторых каталогов файловой системы GNU Linux {#tbl:std-dir}
94
95 | Имя каталога | Описание
96 |-----|-----
97 | `/` | Корневая директория, содержащая всю
98 | файловую |
99 | `/bin` | Основные системные утилиты, необходимые как в однопользовательском режиме,
100 | так и при обычной работе всем пользователям |
101 | `/etc` | Общесистемные конфигурационные файлы и файлы конфигурации установленных
102 | программ |
103 | `/home` | Содержит домашние директории пользователей, которые, в свою очередь,
104 | содержат персональные настройки и данные пользователя |
105 | `/media` | Точки монтирования для сменных
106 | носителей |
107 | `/root` | Домашняя директория пользователя
108 |
109 | `/tmp` | Временные
110 | файлы |
111 | `/usr` | Вторичная иерархия для данных
112 | пользователя |
113
114 Более подробно про Unix см. в [@tanenbaum_book_modern-os_ru; @robbsins_book_bash_en;
115 @zarrelli_book_mastering-bash_en; @newham_book_learning-bash_en].
116
117 # Выполнение лабораторной работы
118
119
120 Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. [-
121 @fig:001]).
```

Рис. 2.5: Открытие шаблона отчета

После изучения структуры заполнил отчет. (рис. 2.6)

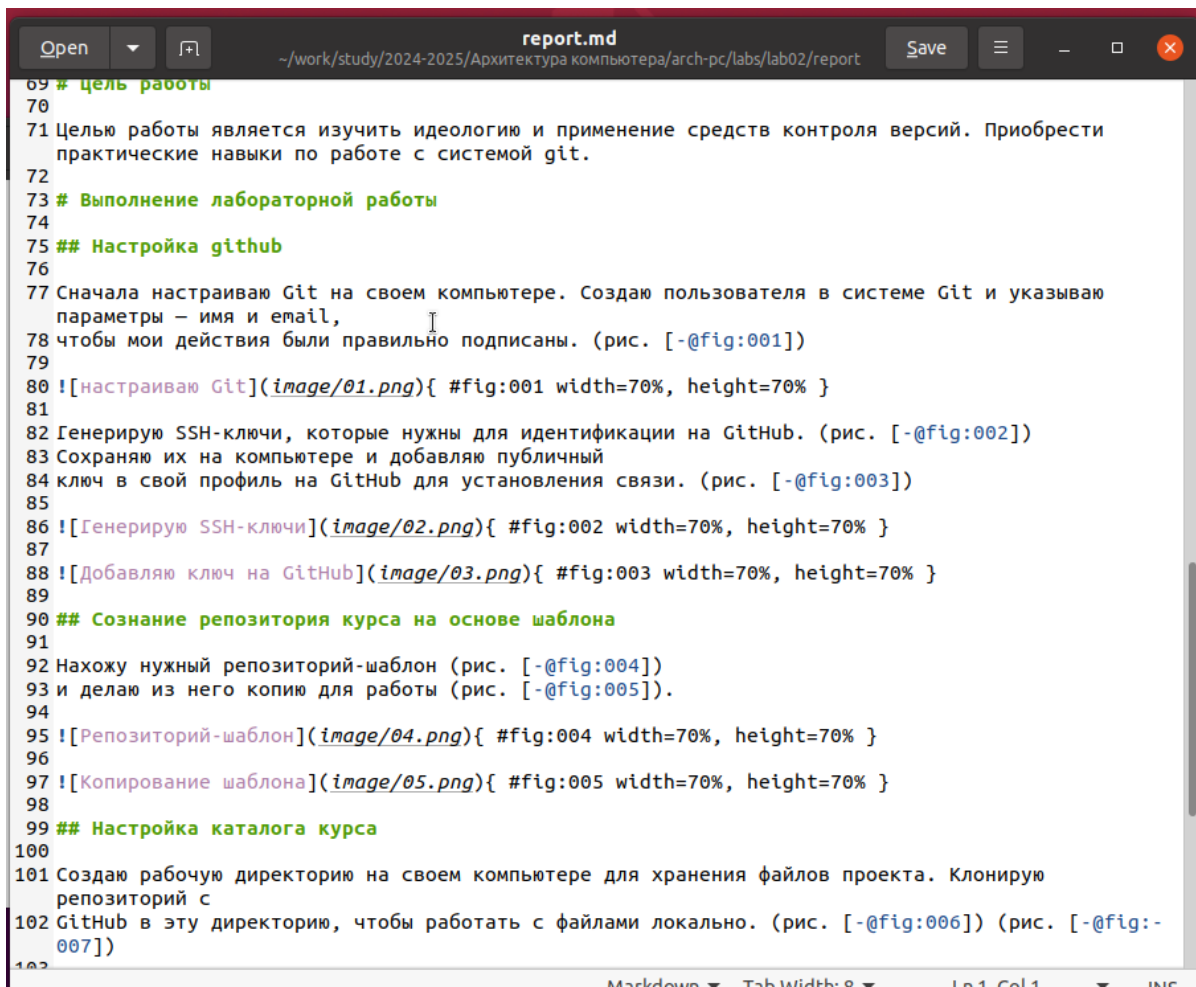


```
72
73 # Выполнение лабораторной работы
74
75 ## Знакомство с Markdown
76
77 В процессе выполнения лабораторной работы установил необходимые программы: `pandoc` и
78 `TexLive`, как было указано в инструкциях.
79 Перешел в каталог курса, созданный в ходе выполнения лабораторной работы №3, затем в
80 директорию с шаблоном отчета по лабораторной работе №3.
81 Для компиляции шаблона использовал `Makefile`, выполнив команду `make`. (рис. [-@fig:001])
82
83 Создание PDF-файла с первого раза не удалось, так как потребовалась установка дополнительных
84 шрифтов.
85 ![Компиляция шаблона с использованием Makefile](image/01.png){ #fig:001 width=70%,
86 height=70% }
87 После успешной компиляции были сгенерированы файлы `report.pdf` (рис. [-@fig:003]) и
88 `report.docx` (рис. [-@fig:002]), что я проверил.
89 ![Файл report.docx](image/02.png){ #fig:002 width=70%, height=70% }
90
91 ![Файл report.pdf](image/03.png){ #fig:003 width=70%, height=70% }
92
93 Затем с помощью команды `make clean` удалил сгенерированные файлы. После выполнения команды
94 убедился, что файлы `report.pdf` и `report.docx` были удалены. (рис. [-@fig:004])
95 ![Удаление файлов report.docx и report.pdf](image/04.png){ #fig:004 width=70%, height=70% }
96
97 Открыл файл `report.md` в текстовом редакторе, например, `gedit`, и внимательно изучил его
98 структуру. (рис. [-@fig:005])
99 ![Открытие шаблона отчета](image/05.png){ #fig:005 width=70%, height=70% }
100
101 После изучения структуры заполнил отчет. (рис. [-@fig:006])
102
```

Рис. 2.6: Заполнение шаблона отчета

2.2 Задание для самостоятельной работы

В рамках самостоятельной работы заполнил отчет для лабораторной работы №2, как это было указано в задании. (рис. 2.7)



```
69 # цель работы
70
71 Целью работы является изучить идеологию и применение средств контроля версий. Приобрести
  практические навыки по работе с системой git.
72
73 # Выполнение лабораторной работы
74
75 ## Настройка github
76
77 Сначала настраиваю Git на своем компьютере. Создаю пользователя в системе Git и указываю
  параметры – имя и email,
78 чтобы мои действия были правильно подписаны. (рис. [-@fig:001])
79
80 ![настраиваю Git](image/01.png){ #fig:001 width=70%, height=70% }
81
82 Генерирую SSH-ключи, которые нужны для идентификации на GitHub. (рис. [-@fig:002])
83 Сохраняю их на компьютере и добавляю публичный
84 ключ в свой профиль на GitHub для установления связи. (рис. [-@fig:003])
85
86 ![Генерирую SSH-ключи](image/02.png){ #fig:002 width=70%, height=70% }
87
88 ![Добавляю ключ на GitHub](image/03.png){ #fig:003 width=70%, height=70% }
89
90 ## Сознание репозитория курса на основе шаблона
91
92 Нахожу нужный репозиторий-шаблон (рис. [-@fig:004])
93 и делаю из него копию для работы (рис. [-@fig:005]).
94
95 ![Репозиторий-шаблон](image/04.png){ #fig:004 width=70%, height=70% }
96
97 ![Копирование шаблона](image/05.png){ #fig:005 width=70%, height=70% }
98
99 ## Настройка каталога курса
100
101 Создаю рабочую директорию на своем компьютере для хранения файлов проекта. Клониров
  репозиторий с
102 GitHub в эту директорию, чтобы работать с файлами локально. (рис. [-@fig:006]) (рис. [-@fig:-
  007])
103
```

Рис. 2.7: Заполнение шаблона для второй лабораторной работы

Экспортировал файлы отчета в форматы для загрузки. (рис. 2.8)

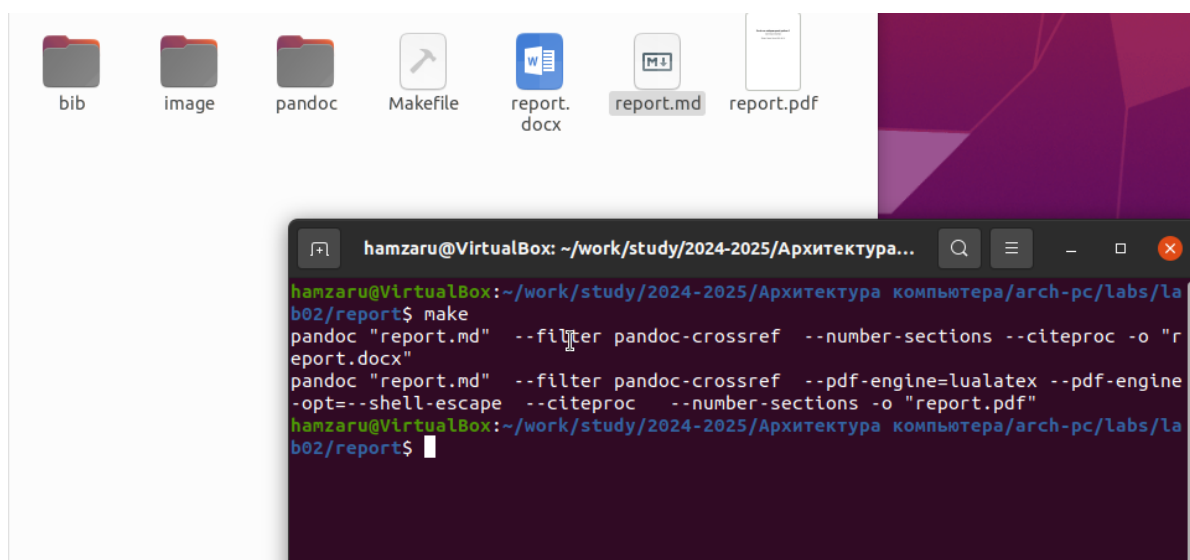


Рис. 2.8: Экспорт отчетных файлов

3 Выводы

Изучили синтаксис языка разметки Markdown, получили отчет из шаблона при помощи Makefile.