

Отчёт по лабораторной работе 3

Архитектура компьютера

Исмаил Хамза НКАбд-03-24

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
4.1	Знакомство с Markdown	8
4.2	Выполнение заданий для самостоятельной работы.	14
5	Выводы	17
	Список литературы	18

Список иллюстраций

4.1	Компиляция файлов	9
4.2	Просмотр dosx файла	10
4.3	Просмотр pdf файла	11
4.4	Удаление файлов dosx и pdf	12
4.5	Изучаю шаблон отчета	13
4.6	Заполняю свой отчет	14
4.7	Заполняю отчет по лабораторной №2	15
4.8	Компилирую отчет по лабораторной №2	16

Список таблиц

1 Цель работы

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

2 Задание

1. В соответствующем каталоге сделайте отчёт по лабораторной работе № 3 в формате Markdown. В качестве отчёта необходимо предоставить отчёты в 3 форматах: pdf, docx и md.
2. Загрузите файлы на github.

3 Теоретическое введение

Markdown - это простой язык разметки, который позволяет быстро и легко форматировать текст. Он используется для создания структурированных документов, включая веб-страницы, блоги, форумы и документацию.

Markdown предназначен быть простым в использовании и чтении, и включает в себя несколько простых синтаксических правил. Он позволяет добавлять форматирование и структуру к тексту, не требуя сложных или громоздких тегов.

4 Выполнение лабораторной работы

4.1 Знакомство с Markdown

Установили программы pandoc и TexLive по указаниям в лабораторной работе.

Открываю терминал, перехожу в каталог курса сформированный при выполнении лабораторной работы №3: Обновляю локальный репозиторий, скачав изменения из удаленного репозитория.

Перехожу в каталог с шаблоном отчета по лабораторной работе № 3

Провожу компиляцию шаблона с использованием Makefile. Для этого ввожу команду make. При успешной компиляции должны сгенерироваться файлы report.pdf и report.docx. Открою их и проверю корректность полученных файлов. (рис. 4.1, 4.2, 4.3)

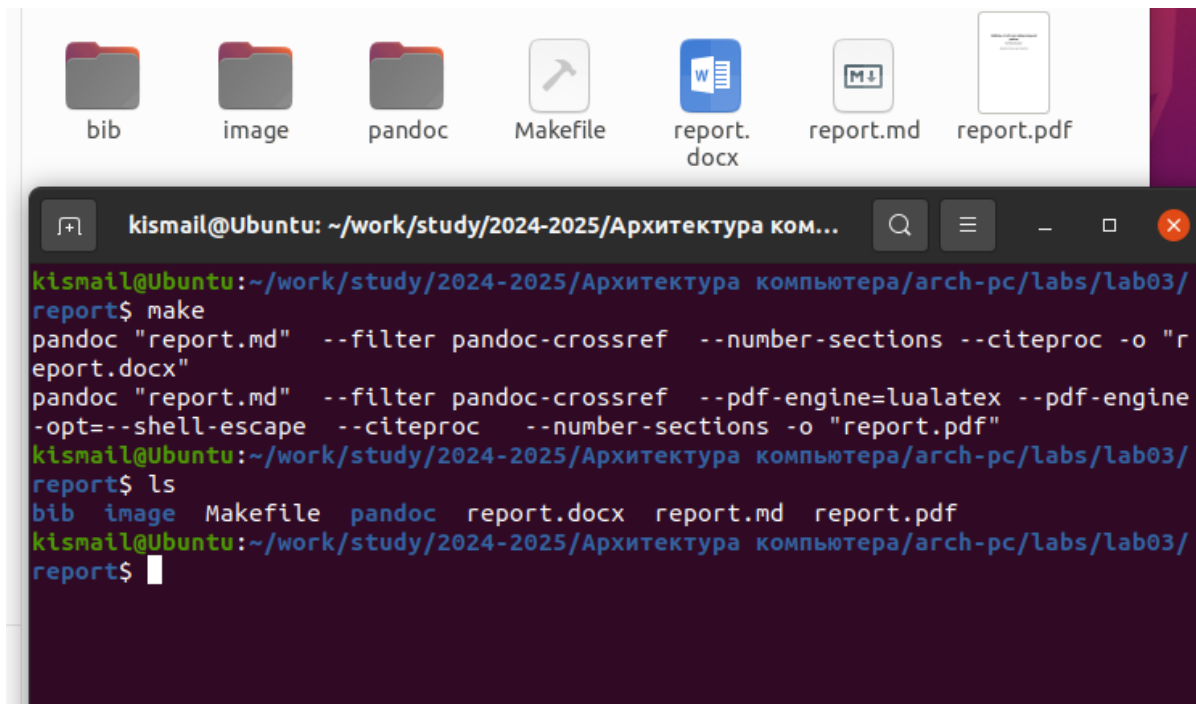


Рис. 4.1: Компиляция файлов

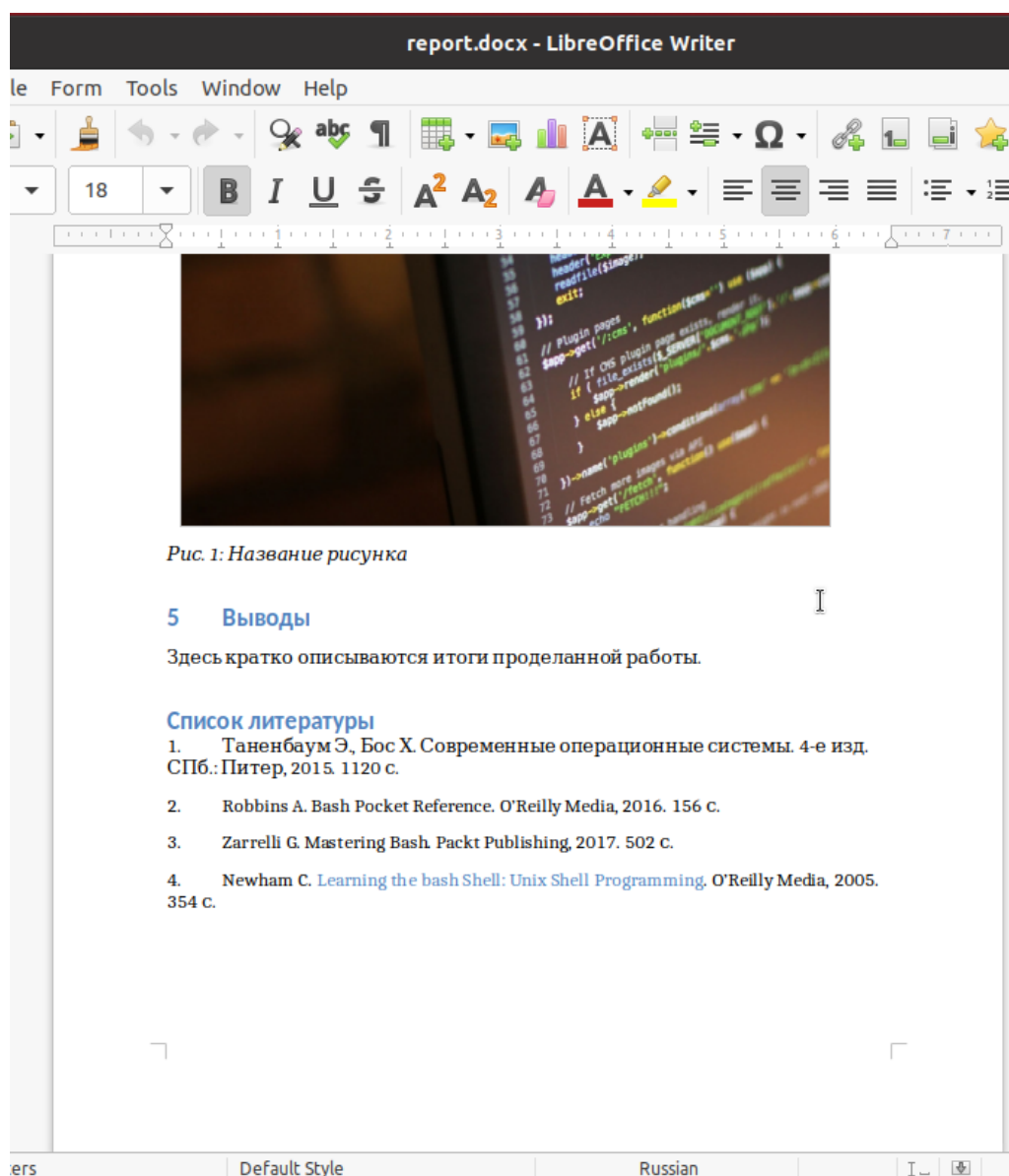


Рис. 4.2: Просмотр docx файла

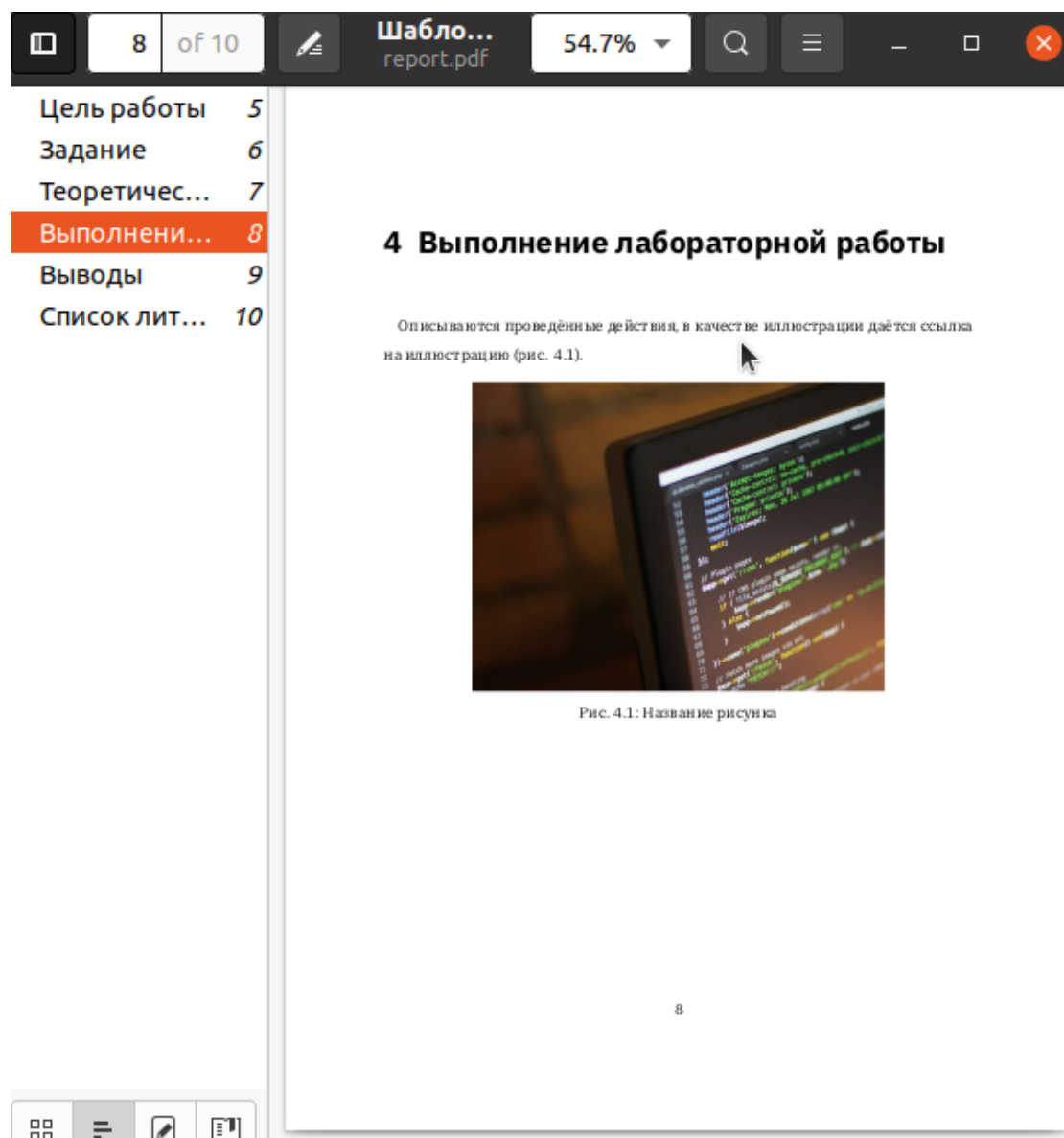


Рис. 4.3: Просмотр pdf файла

Удаляю полученные файлы с использованием Makefile. Для этого ввожу команду `make clean`. Проверю, что после этой команды файлы `report.pdf` и `report.docx` были удалены. (рис. 4.4)

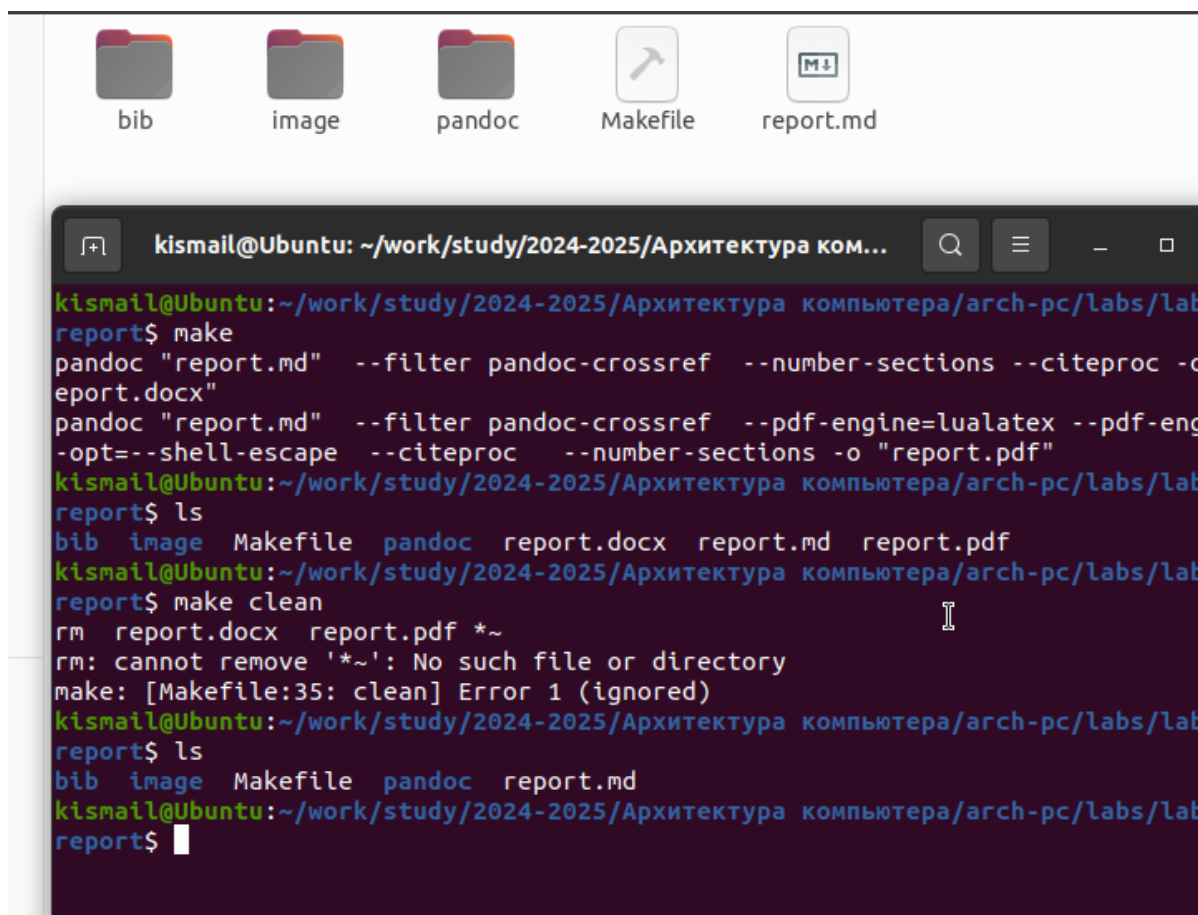


Рис. 4.4: Удаление файлов docx и pdf

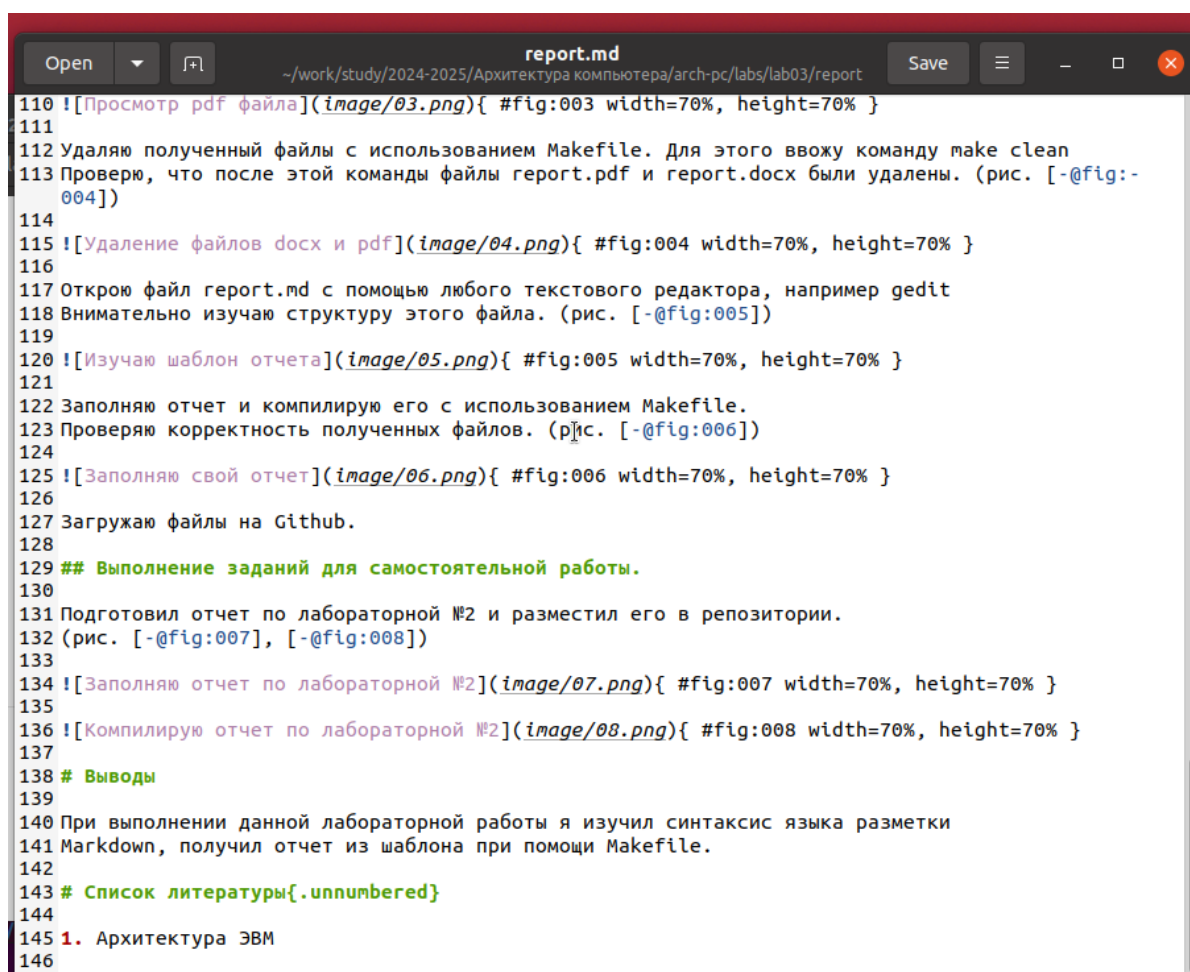
Открою файл report.md с помощью любого текстового редактора, например gedit Внимательно изучаю структуру этого файла. (рис. 4.5)

```
report.md
~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab03/report

196 | -----|-----
197 | `/'      | Корневая директория, содержащая всю
    | файловую |
198 | `/bin`   | Основные системные утилиты, необходимые как в однопользовательском режиме,
    | так и при |
    | обычной |
    | работе |
    | всем   |
    | пользователям |
199 | `/etc`   | Общесистемные конфигурационные файлы и файлы конфигурации установленных
    | программ |
200 | `/home`  | Содержит домашние директории пользователей, которые, в свою очередь,
    | содержат |
    | персональные |
    | настройки и |
    | данные пользователя |
201 | `/media` | Точки монтирования для сменных
    | носителей |
202 | `/root`  | Домашняя директория пользователя
203 | `/tmp`   | Временные
    | файлы |
204 | `/usr`   | Вторичная иерархия для данных
    | пользователя |
205
206 Более подробно про Unix см. в [@tanenbaum_book_modern-os_ru; @robbsins_book_bash_en;
    | @zarrelli_book_mastering-bash_en; @newham_book_learning-bash_en].
207
208 # Выполнение лабораторной работы
209
210 Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. [-
    | @fig:001]).
211
212 ![Название рисунка](image/placeimg_800_600_tech.jpg){#fig:001 width=70%}
213
214 # Выводы
215
216 Здесь кратко описываются итоги проделанной работы.
217
218 # Список литературы{.unnumbered}
219
220 ::: {#refs}
```

Рис. 4.5: Изучаю шаблон отчета

Заполняю отчет и компилирую его с использованием Makefile. Проверяю корректность полученных файлов. (рис. 4.6)



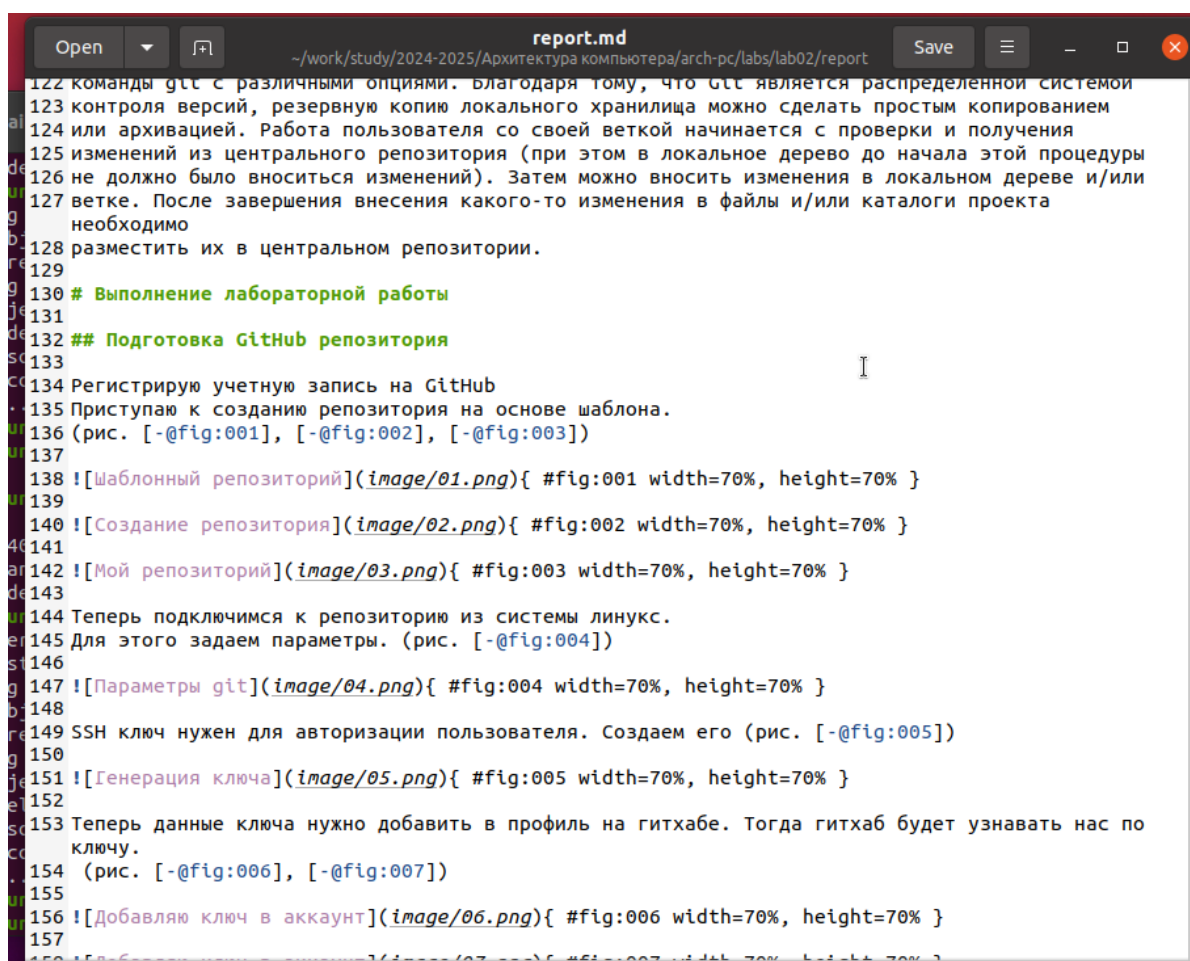
```
110 ![Просмотр pdf файла](image/03.png){ #fig:003 width=70%, height=70% }
111
112 Удаляю полученные файлы с использованием Makefile. Для этого ввожу команду make clean
113 Проверю, что после этой команды файлы report.pdf и report.docx были удалены. (рис. [-@fig:-
004])
114
115 ![Удаление файлов docx и pdf](image/04.png){ #fig:004 width=70%, height=70% }
116
117 Открою файл report.md с помощью любого текстового редактора, например gedit
118 Внимательно изучаю структуру этого файла. (рис. [-@fig:005])
119
120 ![Изучаю шаблон отчета](image/05.png){ #fig:005 width=70%, height=70% }
121
122 Заполняю отчет и компилирую его с использованием Makefile.
123 Проверяю корректность полученных файлов. (рис. [-@fig:006])
124
125 ![Заполняю свой отчет](image/06.png){ #fig:006 width=70%, height=70% }
126
127 Загружаю файлы на Github.
128
129 ## Выполнение заданий для самостоятельной работы.
130
131 Подготовил отчет по лабораторной №2 и разместил его в репозитории.
132 (рис. [-@fig:007], [-@fig:008])
133
134 ![Заполняю отчет по лабораторной №2](image/07.png){ #fig:007 width=70%, height=70% }
135
136 ![Компилирую отчет по лабораторной №2](image/08.png){ #fig:008 width=70%, height=70% }
137
138 # Выводы
139
140 При выполнении данной лабораторной работы я изучил синтаксис языка разметки
141 Markdown, получил отчет из шаблона при помощи Makefile.
142
143 # Список литературы{.unnumbered}
144
145 1. Архитектура ЭВМ
146
```

Рис. 4.6: Заполняю свой отчет

Загружаю файлы на Github.

4.2 Выполнение заданий для самостоятельной работы.

Подготовил отчет по лабораторной №2 и разместил его в репозитории. (рис. 4.7, 4.8)



```
122 команды git с различными опциями. Благодаря тому, что git является распределенной системой
123 контроля версий, резервную копию локального хранилища можно сделать простым копированием
124 или архивацией. Работа пользователя со своей веткой начинается с проверки и получения
125 изменений из центрального репозитория (при этом в локальное дерево до начала этой процедуры
126 не должно было вноситься изменений). Затем можно вносить изменения в локальном дереве и/или
127 ветке. После завершения внесения какого-то изменения в файлы и/или каталоги проекта
    необходимо
128 разместить их в центральном репозитории.
129
130 # Выполнение лабораторной работы
131
132 ## Подготовка GitHub репозитория
133
134 Регистрирую учетную запись на GitHub
135 Приступаю к созданию репозитория на основе шаблона.
136 (рис. [-@fig:001], [-@fig:002], [-@fig:003])
137
138 ![Шаблонный репозиторий](image/01.png){ #fig:001 width=70%, height=70% }
139
140 ![Создание репозитория](image/02.png){ #fig:002 width=70%, height=70% }
141
142 ![Мой репозиторий](image/03.png){ #fig:003 width=70%, height=70% }
143
144 Теперь подключимся к репозиторию из системы линукс.
145 Для этого задаем параметры. (рис. [-@fig:004])
146
147 ![Параметры git](image/04.png){ #fig:004 width=70%, height=70% }
148
149 SSH ключ нужен для авторизации пользователя. Создаем его (рис. [-@fig:005])
150
151 ![Генерация ключа](image/05.png){ #fig:005 width=70%, height=70% }
152
153 Теперь данные ключа нужно добавить в профиль на гитхабе. Тогда гитхаб будет узнавать нас по
    ключу.
154 (рис. [-@fig:006], [-@fig:007])
155
156 ![Добавляю ключ в аккаунт](image/06.png){ #fig:006 width=70%, height=70% }
157
158 ![Безопасность репозитория](image/07.png){ #fig:007 width=70%, height=70% }
```

Рис. 4.7: Заполняю отчет по лабораторной №2

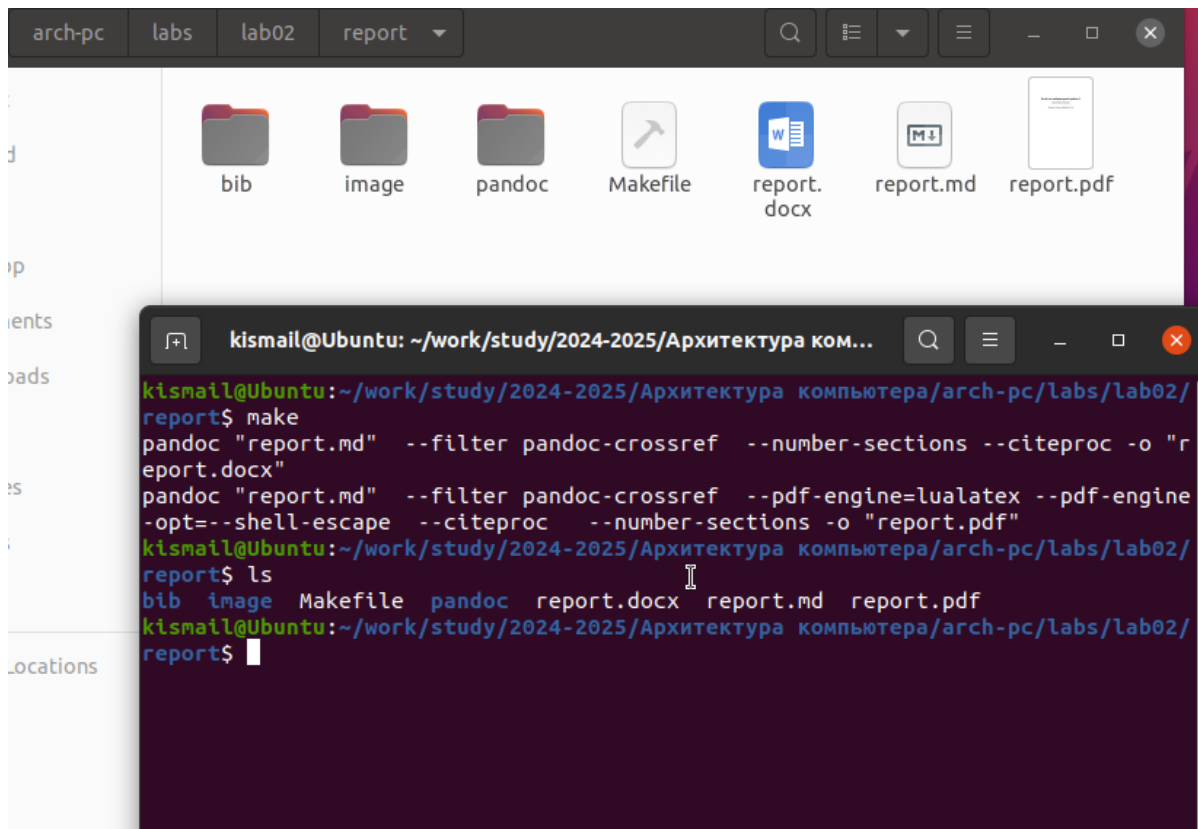


Рис. 4.8: Компилирую отчет по лабораторной №2

5 Выводы

При выполнении данной лабораторной работы я изучил синтаксис языка разметки Markdown, получил отчет из шаблона при помощи Makefile.

Список литературы

1. Архитектура ЭВМ
2. Markdown Документация