

НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«МОСКОВСКИЙ УНИВЕРСИТЕТ «СИНЕРГИЯ»  
Факультет Информационных технологий  
Кафедра Цифровой экономики

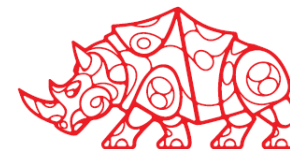
**ОТЧЕТ  
о прохождении учебной практики**

**по профессиональному модулю  
ПМ.02 Осуществление интеграции программных модулей**

**в период с «25» мая 2025 г. по «07» июня 2025 г.**

**Специальность 09.02.07 Информационные системы и  
программирование**

ФИО обучающегося: Ощепков Даниил Алексеевич  
Группа: Vдкп\_111  
ФИО Руководителя: Сибирев Иван Валерьевич



# Содержание

1. Инструктаж по соблюдению правил противопожарной безопасности, правил охраны труда, техники безопасности, санитарно-эпидемиологических правил и гигиенических нормативов
2. Ознакомление с инструментальными средствами
3. Сбор информации об объекте практики и анализ содержания источников
4. Экспериментально-практическая работа. Приобретение необходимых умений и первоначального практического опыта работы по специальности в рамках освоения вида деятельности ВД 2. Осуществление интеграции программных модулей
5. Обработка и систематизация полученного фактического материала



# Организационный этап

Правила внутреннего распорядка, правила и нормы охраны труда, техники безопасности при работе с вычислительной техникой

Я, **Ощепков Даниил Алексеевич** проходил учебную практику в лабораторных условиях на базе Университета «Синергия».

При выполнении индивидуального задания по практике решал кейс № 7 по интеграции git bash

Перед началом практики:

- Принял участие в организационном собрании по практике.
- Ознакомился с комплектом шаблонов отчетной документации по практике.
- Уточнил контакты руководителя практики от Образовательной организации, а также правила в отношении субординации, внешнего вида, графика работы, техники безопасности:

Требования к внешнему виду: Деловой

График работы: 26 мая-7 июня

Круг обязанностей: студент

Доступ к данным: студент

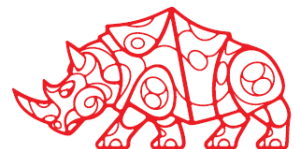
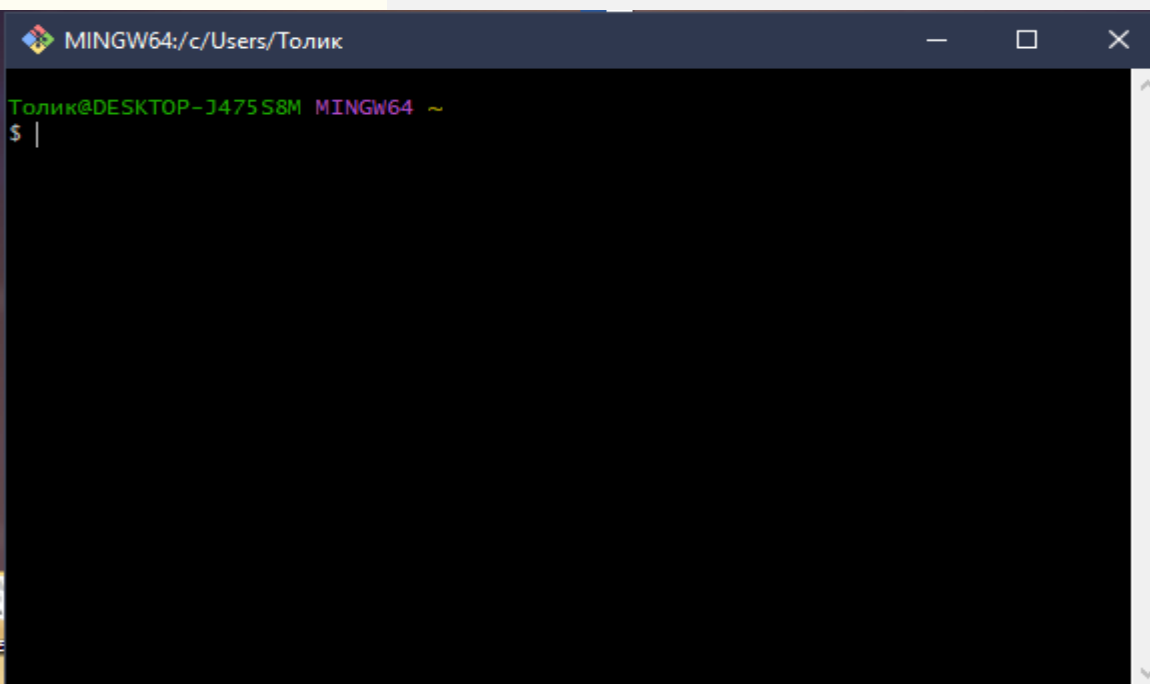


# Исследовательский этап

Сбор информации об объекте практики и анализ содержания источников



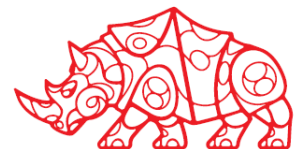
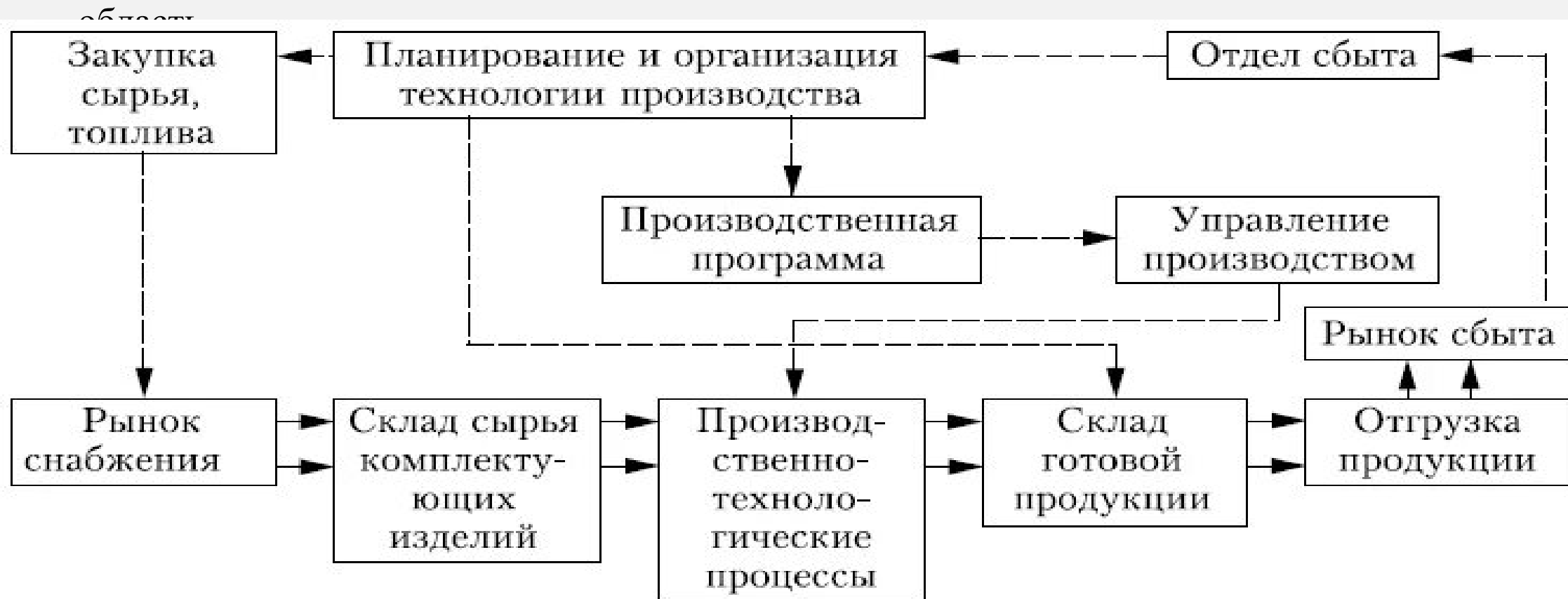
Знакомимся с интринсическими средствами



# Исследовательский этап

Сбор информации об объекте практики и анализ содержания источников

Спроектировать организационную структуру и описать выбранную предметную



# Подготовительный этап

## Описание предметной области

### «Производство: планирование производства, контроль качества, управление запасами»

#### 1. Планирование производства:

- \* Планирование ресурсов: Определение необходимых ресурсов (сырье, материалы, оборудование, персонал) для производства заданного количества продукции в заданные сроки. Это включает в себя прогнозирование спроса, определение производственных мощностей и составление производственного графика.
- \* Управление производственным процессом: Оптимизация последовательности операций, распределение задач между рабочими центрами, мониторинг выполнения плана и корректировка в случае отклонений.
- \* Управление заказами: Обработка заказов клиентов, планирование выпуска продукции в соответствии с заказами, отслеживание выполнения заказов.

#### 2. Контроль качества:

- \* Входной контроль: Проверка качества поступающего сырья и материалов.
- \* Контроль процесса: Мониторинг качества продукции на всех этапах производства.
- \* Выходной контроль: Проверка качества готовой продукции перед отгрузкой клиенту.
- \* Управление несоответствиями: Выявление и анализ дефектов, разработка и внедрение корректирующих действий.
- \* Метрики качества: Сбор и анализ данных о качестве продукции, определение ключевых показателей качества (KPI).

#### 3. Управление запасами:

- \* Управление запасами сырья и материалов: Оптимизация уровня запасов, минимизация затрат на хранение, предотвращение дефицита материалов.
- \* Управление запасами готовой продукции: Оптимизация уровня запасов готовой продукции, минимизация затрат на хранение, обеспечение своевременной поставки клиентам.
- \* Прогнозирование запасов: Предсказание будущих потребностей в сырье, материалах и готовой продукции.
- \* Оптимизация цепочки поставок: Управление взаимоотношениями с поставщиками, обеспечение бесперебойных поставок.

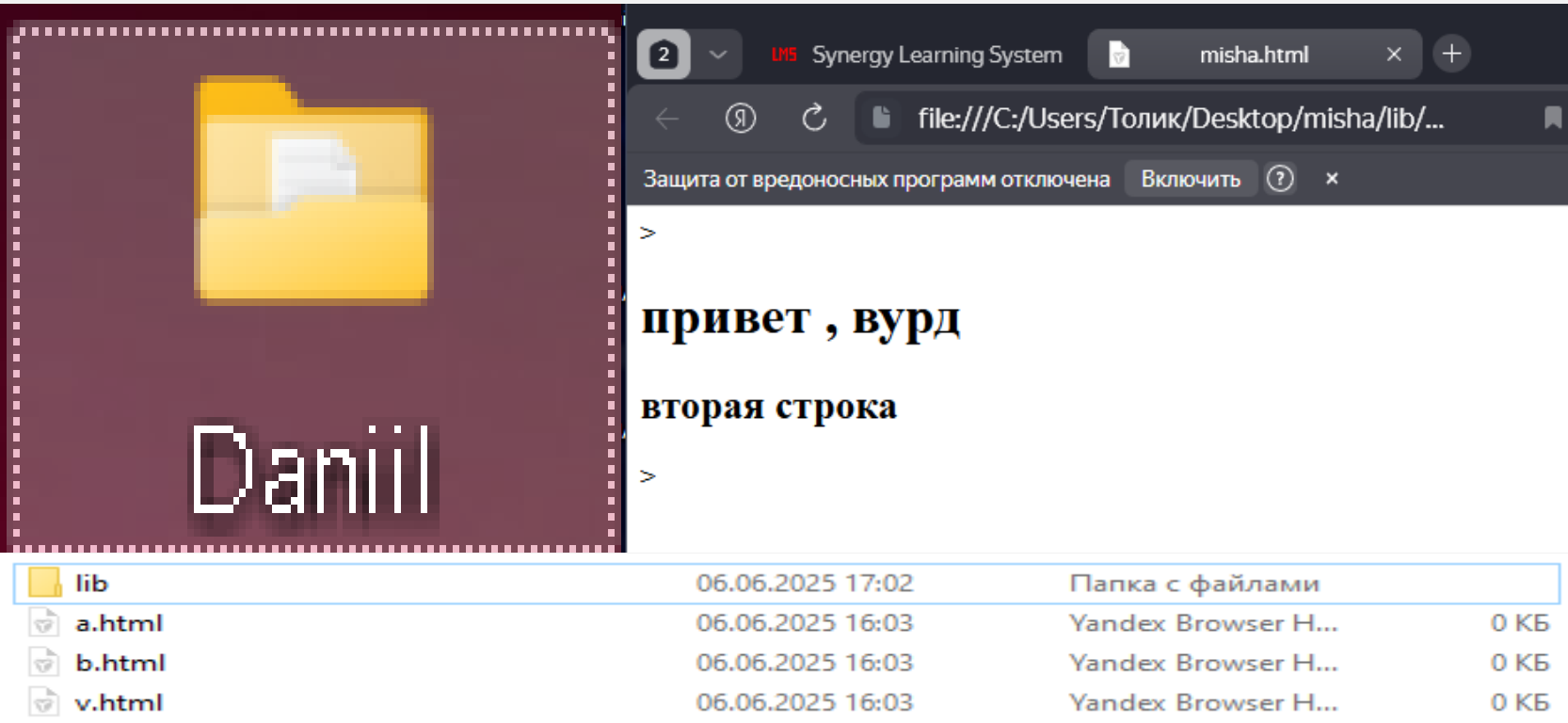
Все три компонента тесно связаны и влияют друг на друга. Например, планирование производства учитывает данные о качестве продукции и наличие запасов. Контроль качества влияет на уровень брака и, соответственно, на объемы производства и запасы. Управление запасами обеспечивает бесперебойное снабжение производства материалами и своевременную отгрузку готовой продукции. Эффективное взаимодействие этих трех компонентов критически важно для оптимизации производственного процесса и достижения высокой производительности и качества продукции.



# Этап проектирования

Использование системы контроля версий и методов для получения кода с заданной функциональностью и степенью качества

## Создание проекта



The screenshot displays a web browser window and a file explorer. The browser window shows a file named `misha.html` with the content:

```
>  
привет , вурд  
вторая строка  
>
```

The file explorer shows a folder named `lib` containing three files:

File Name	Modified Date	Modified Time	Source	Size
lib	06.06.2025	17:02	Папка с файлами	
a.html	06.06.2025	16:03	Yandex Browser H...	0 КБ
b.html	06.06.2025	16:03	Yandex Browser H...	0 КБ
v.html	06.06.2025	16:03	Yandex Browser H...	0 КБ



# Этап проектирования

Использование системы контроля версий и методов для получения кода с заданной функциональностью и степенью качества

Индексация изменений и 1 коммит

a.html – Блокнот

Файл Правка Формат Вид Справка

vnosim  
1comm

b.html – Блокнот




Файл Правка Формат Вид Справка

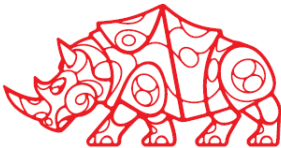
vnosim  
2comm

v.html – Блокнот

Файл Правка Формат Вид Справка

vnosim  
3comm

 a.html	07.06.2025 14:47	Yandex Browser H...	1 КБ
 b.html	07.06.2025 14:47	Yandex Browser H...	1 КБ
 v.html	07.06.2025 14:48	Yandex Browser H...	1 КБ





# Этап проектирования

Использование системы контроля версий и методов для получения кода с заданной функциональностью и степенью качества

Скриншоты индексации и коммита

```
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   store.html

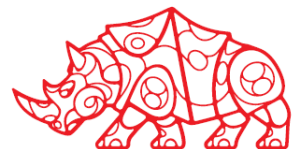
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    sell.txt
    sell_inf.txt
    ../../Текстовый документ.txt

Temyp@INBOOK_Y1_PLUS_MSYS /c/store managment/Desktop/store (master)
$ git status
On branch master

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    deleted:    store.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    ../
    ../../Test.txt

Temyp@INBOOK_Y1_PLUS_MSYS /c/store managment/Desktop/store (master)
$ touch sell.html
```



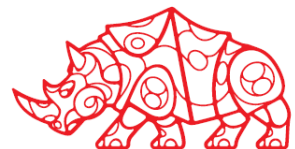
# Этап проектирования

Использование системы контроля версий и методов для получения кода с заданной функциональностью и степенью качества

нужно указать параметры установки окончаний строк и установка отображения Unicode

```
Толик@DESKTOP-J475S8M MINGW64 ~  
$ git config --global core.autocrlf true  
  
Толик@DESKTOP-J475S8M MINGW64 ~  
$ git config --global core.safecrlf true  
  
Толик@DESKTOP-J475S8M MINGW64 ~  
$ |
```

```
Толик@DESKTOP-J475S8M MINGW64 ~  
$ git config --global core.quotepath off
```



# Этап проектирования

Использование системы контроля версий и методов для получения кода с заданной функциональностью и степенью качества

Просмотр историй изменений

```
Толик@DESKTOP-J475S8M MINGW64 ~/desktop/sergey (master)
$ git log
commit fa6ca8ceb4dd631875ce213872cfcc38cfc5b3b3 (HEAD -> master)
Author: sergey <danik.aesen@yandex.ru>
Date: Sat Jun 7 12:49:06 2025 +0300

    add standard HTML tags
```

Однострочный формат вывода

```
Толик@DESKTOP-J475S8M MINGW64 ~/desktop/sergey (master)
$ git log --pretty=oneline
fa6ca8ceb4dd631875ce213872cfcc38cfc5b3b3 (HEAD -> master) add standard HTML tag
39e6ebf0458c930110c1bd67ddec92873d893081 changes for a and b
0c54321a676c04588747f4192987b61892e6cdb8 first commit
```

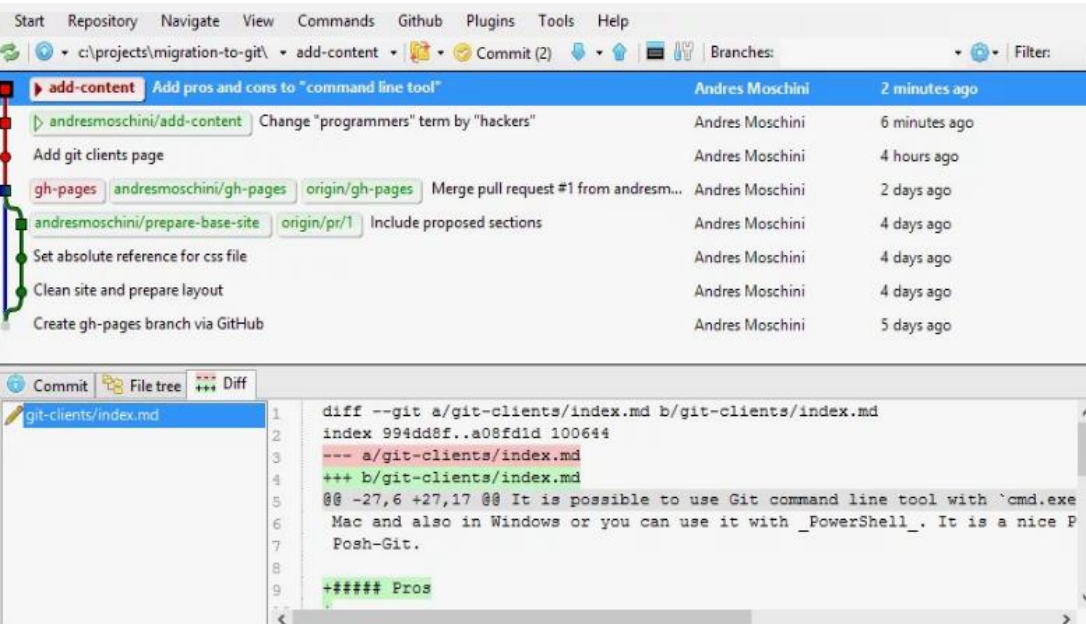
```
Толик@DESKTOP-J475S8M MINGW64 ~/desktop/sergey (master)
$ git log --pretty=oneline --max-count 2
fa6ca8ceb4dd631875ce213872cfcc38cfc5b3b3 (HEAD -> master) add standard HTML tag
39e6ebf0458c930110c1bd67ddec92873d893081 changes for a and b
```

Вывод последних двух изменений

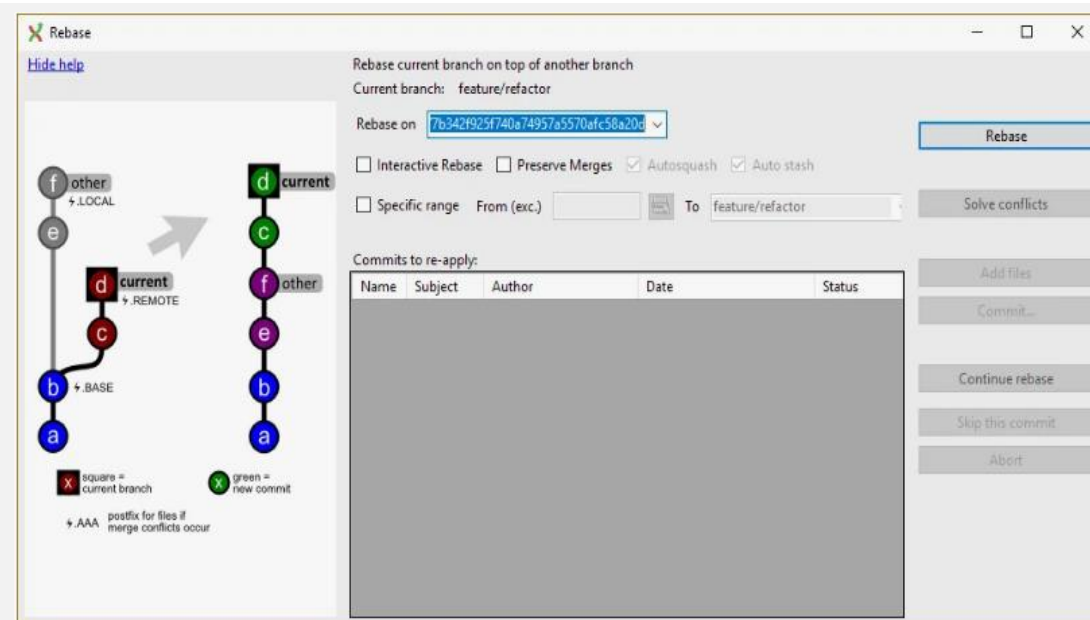


# Этап проектирования

Использование системы контроля версий и методов для получения кода с заданной функциональностью и степенью качества

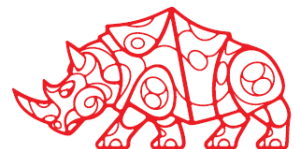


The screenshot shows the GitHub Desktop application. The top bar includes menus: Start, Repository, Navigate, View, Commands, Github, Plugins, Tools, Help. The main area displays a commit history for the repository 'c:\projects\migration-to-git'. The history shows several commits by 'Andres Moschini' with descriptions like 'Add pros and cons to "command line tool"', 'Change "programmers" term by "hackers"', and 'Merge pull request #1 from andresm...'. The bottom panel shows a diff view for the file 'git-clients/index.md', highlighting changes in the content.



The screenshot shows the 'Rebase' dialog in GitHub Desktop. The dialog title is 'Rebase'. It contains a 'Rebase on' dropdown menu with the selected value '7b342f925f740a74957a5570afc58a20d'. Below the dropdown are checkboxes for 'Interactive Rebase', 'Preserve Merges', 'Autosquash', and 'Auto stash'. There is also a 'Specific range' section with 'From (exc.)' and 'To' fields. A 'Rebase' button is on the right. Below the dialog, there is a table for 'Commits to re-apply' with columns: Name, Subject, Author, Date, Status. The table is currently empty. On the left side of the dialog, there is a diagram showing the rebase process with branches 'other', 'current', and 'BASE'.

Здесь мы работаем в git extensions



# Этап проектирования

## Просмотр историй изменений проекта и команды для вывода изменений

```
$ git logo
git: 'logo' is not a git command. See 'git --help'.

The most similar command is
  log

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git log
bash: git: command not found

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git log
commit 555312de4bc1126329f4a0607c2b8603d4137c74 (HEAD -> master)
Author: Jorik <csgo133799@gmail.com>
Date:   Wed May 28 23:19:26 2025 +0300

    add html

commit 2ce7aadd669b5d3e18f51ecbba6b9a186438a719
Author: Jorik <csgo133799@gmail.com>
Date:   Wed May 28 23:18:24 2025 +0300

    add standart html

commit 25e28bb790e9a7424bd2cf2e7130cbdd4a0fb79b
Author: Jorik <csgo133799@gmail.com>
Date:   Wed May 28 23:08:56 2025 +0300

    added h2 tag

commit ee8df9bc86d1ac6cea68735242834e2d0c836770
Author: Jorik <csgo133799@gmail.com>
Date:   Wed May 28 22:45:43 2025 +0300

    Changes 2

commit 13b3c81397ee65968ddb6baf5bf939af4d76f4a5
Author: Jorik <csgo133799@gmail.com>
Date:   Wed May 28 22:44:55 2025 +0300

    Changes

commit a82d66174a64311295a9153b4e566be450fe592f
Author: Jorik <csgo133799@gmail.com>
Date:   Wed May 28 22:42:24 2025 +0300

    FM

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git log --pretty=oneline
555312de4bc1126329f4a0607c2b8603d4137c74 (HEAD -> master) add html
2ce7aadd669b5d3e18f51ecbba6b9a186438a719 add standart html
25e28bb790e9a7424bd2cf2e7130cbdd4a0fb79b added h2 tag
ee8df9bc86d1ac6cea68735242834e2d0c836770 Changes 2
13b3c81397ee65968ddb6baf5bf939af4d76f4a5 Changes
a82d66174a64311295a9153b4e566be450fe592f FM

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git log --pretty=oneline --max-count=2
555312de4bc1126329f4a0607c2b8603d4137c74 (HEAD -> master) add html
2ce7aadd669b5d3e18f51ecbba6b9a186438a719 add standart html

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git log --pretty=oneline --since= '10 days ago'
fatal: ambiguous argument '10 days ago': unknown revision or path not in the working tree
Use '--' to separate paths from revisions, like this:
'git log <revision> -- <path>'
```

```
$ git log --pretty=oneline
555312de4bc1126329f4a0607c2b8603d4137c74 (HEAD -> master) add html
2ce7aadd669b5d3e18f51ecbba6b9a186438a719 add standart html
25e28bb790e9a7424bd2cf2e7130cbdd4a0fb79b added h2 tag
```

```
$ git log --pretty=oneline --max-count=2
555312de4bc1126329f4a0607c2b8603d4137c74 (HEAD -> master) add html
2ce7aadd669b5d3e18f51ecbba6b9a186438a719 add standart html
```

```
git hist
f85fffc6 2025-05-28 | Revert "oops" (HEAD -> master) [Jorik]
7300cc4 2025-05-28 | oops [Jorik]
555312d 2025-05-28 | add html (tag: ver1) [Jorik]
2ce7aad 2025-05-28 | add standart html (tag: ver1-beta) [Jorik]
25e28bb 2025-05-28 | added h2 tag [Jorik]
```



# Этап проектирования

Алиасы (будут приведены команды для работы с ними)

```
C:\store managment\Desktop\store>alias gs='git status '  
"alias" не является внутренней или внешней  
командой, исполняемой программой или пакетным файлом.  
  
C:\store managment\Desktop\store>git checkout 8c62ca6  
Note: switching to '8c62ca6'.  
  
You are in 'detached HEAD' state. You can look around, make experimental  
changes and commit them, and you can discard any commits you make in this  
state without impacting any branches by switching back to a branch.  
  
If you want to create a new branch to retain commits you create, you may  
do so (now or later) by using -c with the switch command. Example:  
  
git switch -c <new-branch-name>  
  
Or undo this operation with:  
  
git switch -  
  
Turn off this advice by setting config variable advice.detachedHead to false  
  
HEAD is now at 8c62ca6 add html  
  
C:\store managment\Desktop\store>cat hello.html  
"cat" не является внутренней или внешней  
командой, исполняемой программой или пакетным файлом.
```

```
C:\store managment\Desktop\store>git config --global alias.co checkout  
C:\store managment\Desktop\store>git config --global alias.ci commit  
C:\store managment\Desktop\store>git config --global alias.st status  
C:\store managment\Desktop\store>git config --global alias.br branch  
C:\store managment\Desktop\store>git config --global alias.hist "log --pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"  
C:\store managment\Desktop\store>git config --global alias.type 'cat-file -t'  
C:\store managment\Desktop\store>git config --global alias.dump 'cat-file -p'  
C:\store managment\Desktop\store>git config --global alias.hist "log --pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"  
C:\store managment\Desktop\store>git hist  
fatal: bad alias.hist string: unclosed quote  
C:\store managment\Desktop\store>git config --global alias.hist "log --pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"  
C:\store managment\Desktop\store>git hist  
fatal: bad alias.hist string: unclosed quote  
C:\store managment\Desktop\store>git config --global alias.hist "log --pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"
```





# Этап проектирования

Скриншоты из приложения 1.3 (работа с коммитом, его индексация и т.д.)

```
Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git add gogo.html

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git commit -m "Add an author"
[master 7425f8c] Add an author
1 file changed, 1 insertion(+)

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git add gogo.html

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git commit --amend -m "Add an author and email"
[master 87742d4] Add an author and email
Date: Sun Jun 1 13:10:14 2025 +0300
1 file changed, 1 insertion(+)

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git hist
* 87742d4 2025-06-01 | Add an author and email (HEAD -> master) [
* 555312d 2025-05-28 | add html (tag: ver1) [Jorik]
* 2ce7aad 2025-05-28 | add standart html (tag: ver1-beta) [Jorik]
* 25e28bb 2025-05-28 | added h2 tag [Jorik]
* ee8df9b 2025-05-28 | Changes 2 [Jorik]
* 13b3c81 2025-05-28 | Changes [Jorik]
* a82d661 2025-05-28 | FM [Jorik]

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ mkdir lib

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git mv gogo.html lib

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ giit status
bash: giit: command not found

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git st
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    renamed:   gogo.html -> lib/gogo.html

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git commit -m "Moved gogo.html to lib"
[master 217287e] Moved gogo.html to lib
1 file changed, 0 insertions(+), 0 deletions(-)
rename gogo.html => lib/gogo.html (100%)

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git add index.html

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ git commit -m "Added index.html"
```

```
Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ ls -C .git/object
ls: cannot access '.git/object': No such file or directory

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ ls -C .git/objects
3/ 25/ 2c/ 34/ 4a/ 57/ 73/ 87/ 96/ a3/ a8/ c8/ e6/ eb/ ef/ f6/ fb/ pack/
1/ 2b/ 2f/ 3b/ 55/ 63/ 74/ 92/ a1/ a4/ ba/ c9/ e7/ ee/ f0/ f8/ info/

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ ls -C .git/Objects/8d
ls: cannot access '.git/Objects/8d': No such file or directory

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ ls -C .git/objects/8d
ls: cannot access '.git/objects/8d': No such file or directory

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ ls -C .git/objects/7d
ls: cannot access '.git/objects/7d': No such file or directory

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ ls -C .git/objects/6d
ls: cannot access '.git/objects/6d': No such file or directory

Alex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
$ ls -C .git/objects/5d
```



# Этап проектирования

## УСТАНОВКА Git Exntensions

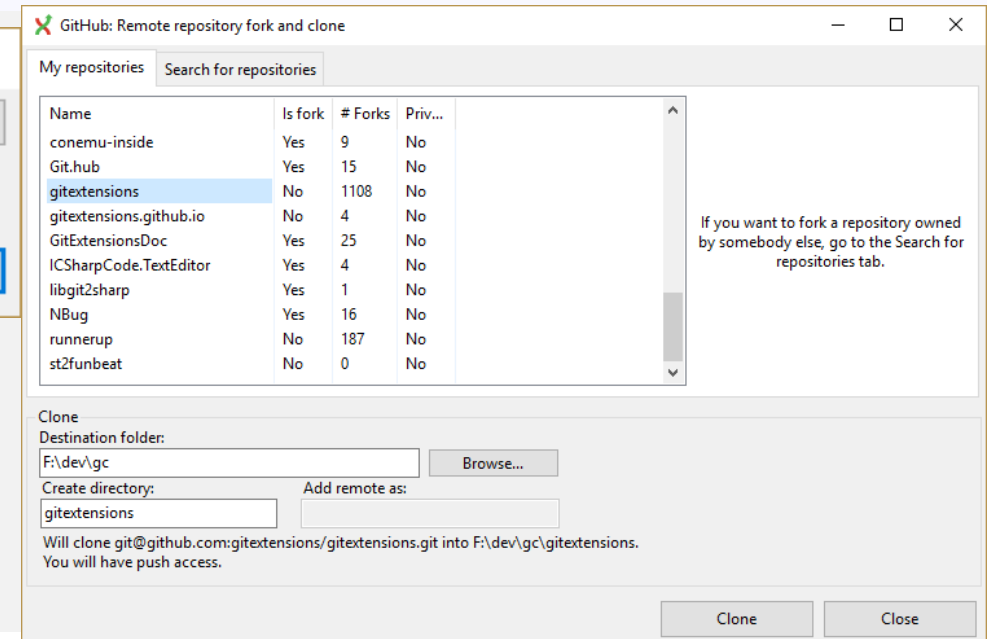
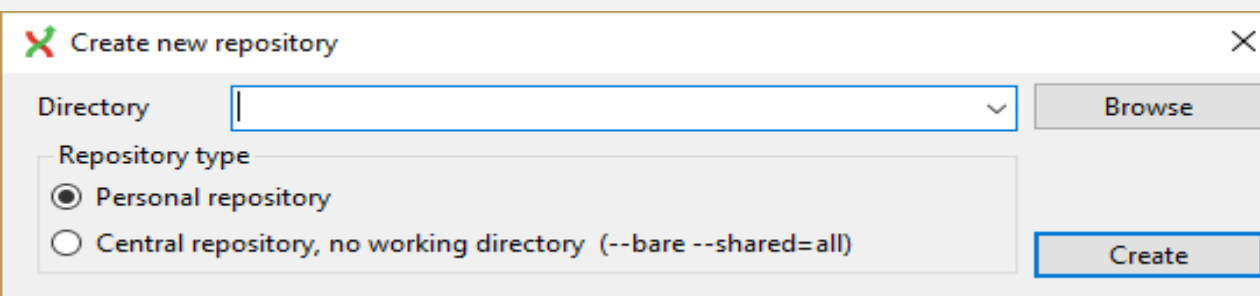
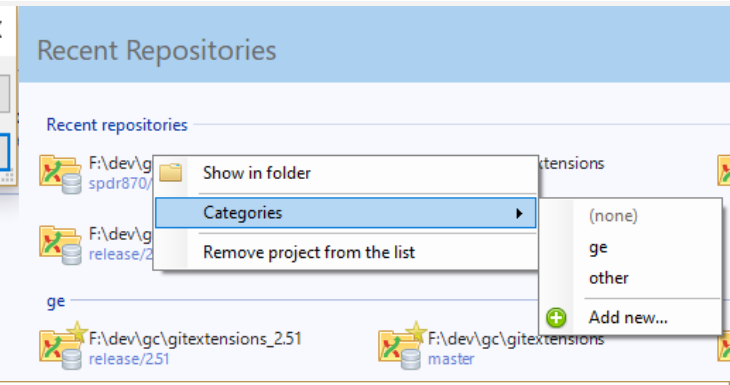
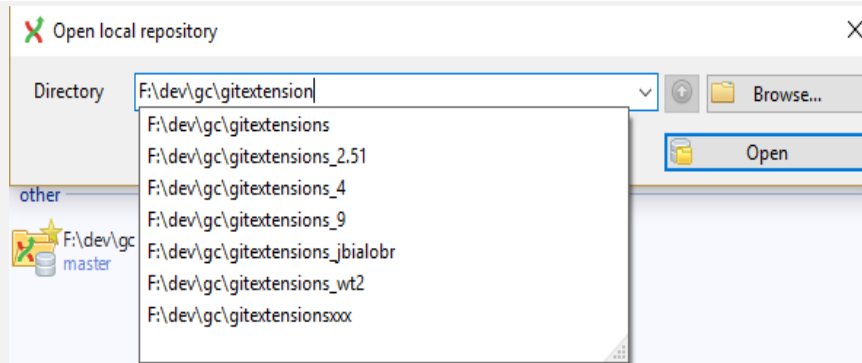
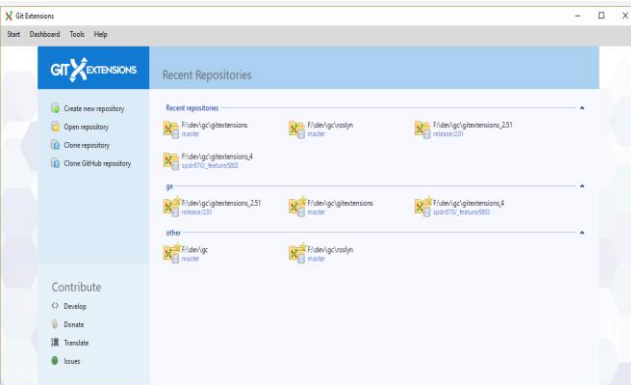
<p><b>Welcome to the Git Extensions 3.00.00.4409 Setup Wizard</b></p> <p>The Setup Wizard will install Git Extensions 3.00.00.4409 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.</p> <p>Back Next Cancel</p>	<p><b>Installation Scope</b></p> <p>Choose the installation scope and folder</p> <p><input type="radio"/> Install just for you (ejgo) Git Extensions 3.00.00.4409 will be installed in a per-user folder and be available just for your user account. You do not need local Administrator privileges.</p> <p><input checked="" type="radio"/> Install for all users of this machine Git Extensions 3.00.00.4409 will be installed in a per-machine folder by default and be available for all users. You can change the default installation folder. You must have local Administrator privileges.</p> <p>Back Next Cancel</p>	<p><b>Destination Folder</b></p> <p>Click Next to install to the default folder or click Change to choos...</p> <p>Install Git Extensions 3.00.00.4409 to:</p> <p>C:\Program Files (x86)\GitExtensions\</p> <p>Change...</p> <p>Back Next Cancel</p>
<p><b>Custom Setup</b></p> <p>Select the way you want features to be installed.</p> <p>Click the icons in the tree below to change the way features will be installed.</p> <p>Git Extensions</p> <ul style="list-style-type: none"><li>Plugins</li><li>Desktop shortcut</li><li>Custom merge scripts</li><li>Spelling dictionaries</li><li>Translations</li><li>Windows Explorer integration</li><li>Add installation directory to PATH</li><li>Visual Studio extension</li></ul> <p>Browse...</p> <p>Back Next Cancel</p>	<p><b>Select SSH Client</b></p> <p>Select the SSH client that will be used by Git Extensions</p> <p><input type="radio"/> OpenSSH OpenSSH is the Git default.</p> <p><input checked="" type="radio"/> PuTTY (plink.exe) PuTTY has better integration with Windows.</p> <p>Back Next Cancel</p>	<p><b>Ready to install Git Extensions 3.00.00.4409</b></p> <p>Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.</p> <p>Back Install Cancel</p>





# Этап проектирования

Репозитории, коммиты, их удаление и т.д.



# Этап проектирования в git extensions

## Способы создания веток и переключения между ними

Чтобы в Git добавить ветку мы используем:

```
$ git branch <name of new branch>
```

После данной операции ветка уже была создана, но вы по-прежнему находитесь в прежней ветке. Если вы планируете переместиться на другую ветку, в том числе только что созданную, необходимо написать **checkout**:

```
$ git checkout <name of branch>
```

Для того чтобы определить, где сейчас находится разработчик, Git использует специальный указатель **HEAD**, ссылающийся на текущую локальную ветку. В результате checkout **HEAD** переместится на иную ветку.

## Как с помощью git branch создать ветку и перейти в нее

Чаще всего при создании новой ветки git пользователю необходимо сразу же переключиться на нее. В таком случае стоит использовать:

```
$ git checkout branch <name of new branch>
```

Это будет равносильно:

```
$ git branch <name of new branch>
```

```
$ git checkout <name of new branch>
```

И также мы получим тот же результат при использовании git checkout с ключом -b:

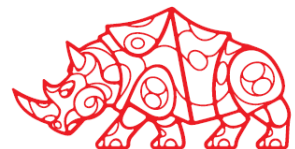
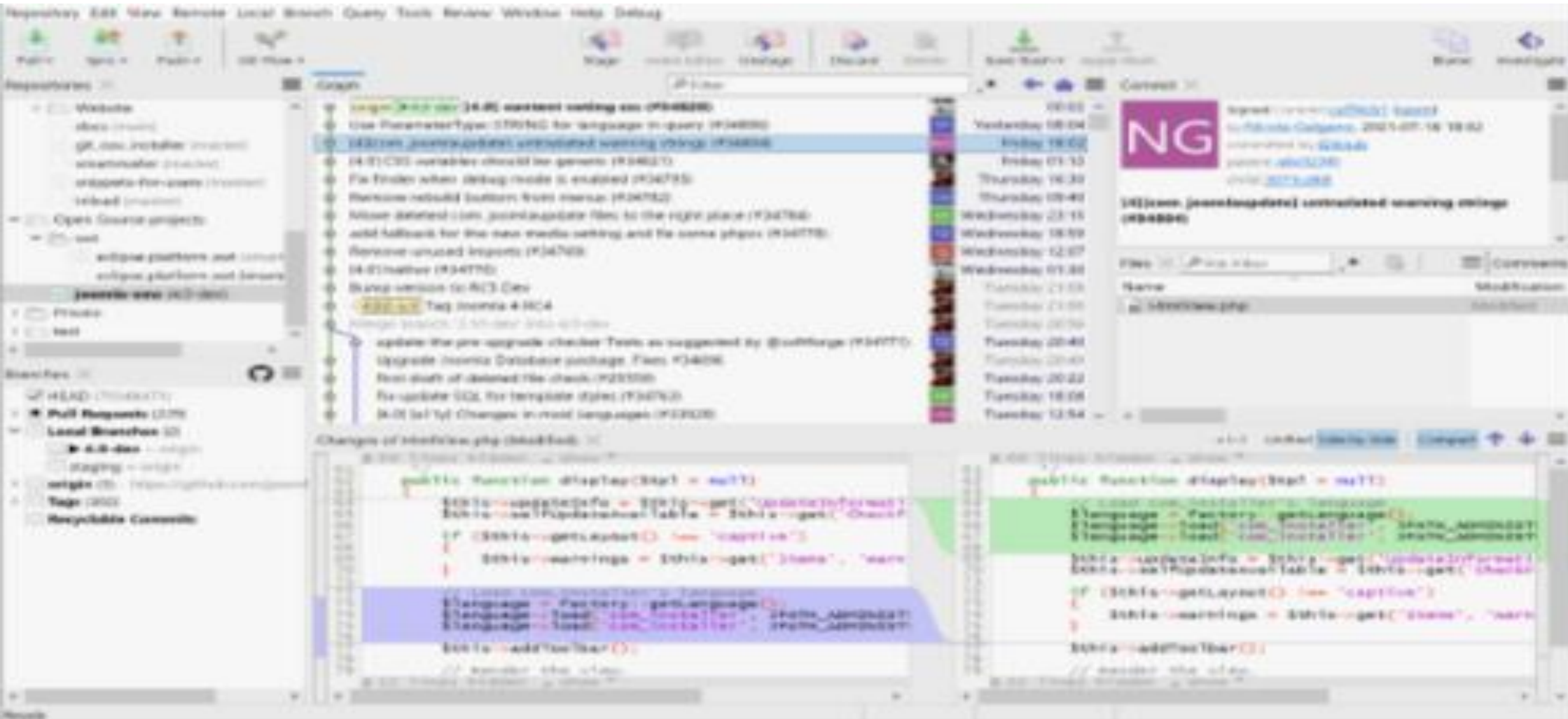
```
$ git checkout -b <name of branch>
```

Если пользователю нужно получить список определенного множества веток, то тогда можно воспользоваться ключами. Одними из самых распространенных будут:

- **-r** — при использовании этого ключа мы получим список удаленных веток,
- **-a** — используя этот параметр, в выводе будут удаленные и локальные ветки.



# Всё ещё проектируем в гите



# Этап проектирования

Работа с веткой style, ниже см. скриншоты (приложение 1.5.)

```
git checkout style
Already on 'style'

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (style)
git hist
74729af 2025-06-01 | Merged master conflict (HEAD -> style) [Jorik]
* a000f19 2025-06-01 | I love YOU (master) [Jorik]
b85870c 2025-06-01 | Merge branch 'master' into style [Jorik]
* 03c6cae 2025-06-01 | Added README.html [Jorik]
38be4d8 2025-06-01 | Update index.html [Jorik]
e8b7815 2025-06-01 | hello.html uses style.css [Jorik]
10d6daf 2025-06-01 | Added css stylesheet [Jorik]
dfad5ae 2025-06-01 | Added css stylesheet [Jorik]
e7961d8 2025-06-01 | Added index.html [Jorik]
217287e 2025-06-01 | Moved gogo.html to lib [Jorik]
87742d4 2025-06-01 | Add an author and email [Jorik]
555312d 2025-05-28 | add html (tag: ver1) [Jorik]
2ce7aad 2025-05-28 | add standart html (tag: ver1-beta) [Jorik]
25e28bb 2025-05-28 | added h2 tag [Jorik]
ee8df9b 2025-05-28 | Changes 2 [Jorik]
13b3c81 2025-05-28 | Changes [Jorik]
a82d661 2025-05-28 | FM [Jorik]

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (style)
git reser --hard e7961d8
it: 'reser' is not a git command. See 'git --help'.

The most similar commands are
  reset
  restore

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (style)
git reset --hard e7961d8
HEAD is now at e7961d8 Added index.html

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (style)
git hist --all
a000f19 2025-06-01 | I love YOU (master) [Jorik]
03c6cae 2025-06-01 | Added README.html [Jorik]
e7961d8 2025-06-01 | Added index.html (HEAD -> style) [Jorik]
217287e 2025-06-01 | Moved gogo.html to lib [Jorik]
87742d4 2025-06-01 | Add an author and email [Jorik]
* f85ffc6 2025-05-28 | Revert "oops" (tag: oopgs) [Jorik]
* 7300cc4 2025-05-28 | oops [Jorik]
/
555312d 2025-05-28 | add html (tag: ver1) [Jorik]
2ce7aad 2025-05-28 | add standart html (tag: ver1-beta) [Jorik]
25e28bb 2025-05-28 | added h2 tag [Jorik]
ee8df9b 2025-05-28 | Changes 2 [Jorik]
13b3c81 2025-05-28 | Changes [Jorik]
a82d661 2025-05-28 | FM [Jorik]

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (style)
git checkout master
Switched to branch 'master'

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
git reset --hard a000f19
HEAD is now at a000f19 I love YOU

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/gogo (master)
git hist --all
a000f19 2025-06-01 | I love YOU (HEAD -> master) [Jorik]
03c6cae 2025-06-01 | Added README.html [Jorik]
e7961d8 2025-06-01 | Added index.html (style) [Jorik]
217287e 2025-06-01 | Moved gogo.html to lib [Jorik]
```

```
$ git clone gogo cloned_gogo
Cloning into 'cloned_gogo'...
done.

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop
$ ls
A Way Out.url'      'Dota 2.url'      Pinokio.lnk*      'Visual Studio Code.lnk*'  drivers/
Apps/               Games/            'Plants vs. Zombies'/  'Wallpaper Engine.url'    gogo/
Counter-Strike 2.url' 'GeoGuessr Steam Edition.url' 'Spotify.lnk*'      cloned_gogo/              place/
Discord.lnk*        'Logitech G HUB.lnk*'  TLauncher.lnk*      desktop.ini                практика/

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop
$ cd cloned_gogo

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/cloned_gogo (master)
$ ls
README.html a.html b.html c.html index.html lib/ v.html

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/cloned_gogo (master)
$ git hist --all
a000f19 2025-06-01 | I love YOU (HEAD -> master, origin/style, origin/master, origin/HEAD) [Jorik]
03c6cae 2025-06-01 | Added README.html [Jorik]
e7961d8 2025-06-01 | Added index.html [Jorik]
217287e 2025-06-01 | Moved gogo.html to lib [Jorik]
87742d4 2025-06-01 | Add an author and email [Jorik]
* f85ffc6 2025-05-28 | Revert "oops" (tag: oopgs) [Jorik]
* 7300cc4 2025-05-28 | oops [Jorik]
/
555312d 2025-05-28 | add html (tag: ver1) [Jorik]
2ce7aad 2025-05-28 | add standart html (tag: ver1-beta) [Jorik]
25e28bb 2025-05-28 | added h2 tag [Jorik]
ee8df9b 2025-05-28 | Changes 2 [Jorik]
13b3c81 2025-05-28 | Changes [Jorik]
a82d661 2025-05-28 | FM [Jorik]

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/cloned_gogo (master)
$ git remote
origin

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/cloned_gogo (master)
$ git remote show origin
remote origin
Fetch URL: C:/Users/Пользователь/Desktop/gogo
Push URL: C:/Users/Пользователь/Desktop/gogo
HEAD branch: master
Remote branches:
  master tracked
  style tracked
Local branch configured for 'git pull':
  master merges with remote master
Local ref configured for 'git push':
  master pushes to master (up to date)

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/cloned_gogo (master)
$ git branch
* master

lex3s@DESKTOP-OQK9QN7 MINGW64 ~/Desktop/cloned_gogo (master)
$ git branch -a
* master
remotes/origin/HEAD -> origin/master
remotes/origin/master
remotes/origin/style
```



# Отчетный этап

## Формирование отчетной документации по результатам работ

### При оформлении отчетных материалов следует придерживаться действующих стандартов.

- В соответствии с ГОСТ 2.105-79 «Общие требования к текстовым документам» иллюстрации (графики, схемы, диаграммы) могут быть приведены как в основном тексте, так и в приложении. Все иллюстрации именуют рисунками. Все рисунки, таблицы и формулы нумеруют арабскими цифрами последовательно (сквозная нумерация) или в пределах раздела (относительная нумерация). В приложении - в пределах приложения. Каждый рисунок должен иметь подрисуночную подпись - название, помещаемую под рисунком.
- Рисунки следует размещать так, чтобы их можно было рассматривать без поворота страницы. Если такое размещение невозможно, рисунки следует располагать так, чтобы для просмотра надо было повернуть страницу по часовой стрелке. В этом случае верхним краем является левый край страницы. Расположение и размеры полей сохраняются.
- Номер таблицы размещают в правом верхнем углу или перед заголовком таблицы, если он есть. Заголовок, кроме первой буквы, выполняют строчными буквами. Ссылки на таблицы в тексте пояснительной записки указывают в виде слова «табл.» и номера таблицы. *Например: Результаты тестов приведены в табл. 4.*



# Отчетный этап

## Формирование отчетной документации по результатам работ

### **При оформлении отчетных материалов следует придерживаться действующих стандартов.**

- Список литературы должен включать все использованные источники. Сведения о книгах (монографиях, учебниках, пособиях, справочниках и т.д.) должны содержать: фамилию и инициалы автора, заглавие книги, место издания, издательство, год издания. При наличии трех и более авторов допускается указывать фамилию и инициалы только первого из них со словами «и др.». Издательство надо приводить полностью в именительном падеже: допускается сокращение названия только двух городов: Москва (М.) и Санкт-Петербург (СПб.).
- Сведения о статье из периодического издания должны включать: фамилию и инициалы автора, наименование статьи, издания (журнала), серии (если она есть), год выпуска, том (если есть), номер издания (журнала) и номера страниц, на которых помещена статья.
- При ссылке на источник из списка литературы (особенно при обзоре аналогов) надо указывать порядковый номер по списку литературы, заключенный в квадратные скобки; например: [5].





# Отчетный этап

Выводы о результатах прохождения учебной практики:  
выполняемая работа, приобретенные умения и навыки

## Подведите итоги прохождения учебной практики:

В ходе прохождения учебной практики мной были получены освоены следующие навыки:

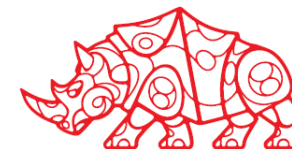
1. Изучил git.
2. Получил навыки работы в консоли.
3. Попробовал разные инструменты для работы в контроле версий гит
4. Изучил основные принципы работы в git extensions



# Отчетный этап

## Список используемой литературы










1. Советов, Б. Я. Базы данных : учебник для среднего профессионального образования / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 420 с. — (Профессиональное образование). — ISBN 978-5-534-09324-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/514585>
2. Нагаева, И.А. Основы алгоритмизации и программирования: практикум : [12+] / И.А. Нагаева, И.А. Кузнецов. — Москва ; Берлин : Директ-Медиа, 2021. — 169 с. : схем. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/>
3. <https://habr.com/ru/articles/541258/> ГИТ для новичков





# Отчетный этап

## Приложения

- 1.1. Гит.docx 
  - 1.2. Гит.docx 
  - 1.3. Гит.docx 
  - 1.4. Гит.docx 
  - 1.5. Гит.docx 
  - 1.6. Гит.docx 
  - 1.7. Гит.docx 
  - 1.8. Гит.docx 
  - 1.9. Гит.docx 
- 1.9git

