

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2
дисциплины «Основы программной инженерии»

Выполнил:
Плугатырев Владислав Алексеевич
1 курс, группа ПИЖ-б-о-22-1,
09.03.04 «Программная инженерия»,
направленность (профиль) «Разработка и
сопровождение программного
обеспечения», очная форма обучения

(подпись)

Доцент кафедры инфокоммуникаций
Воронкин Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

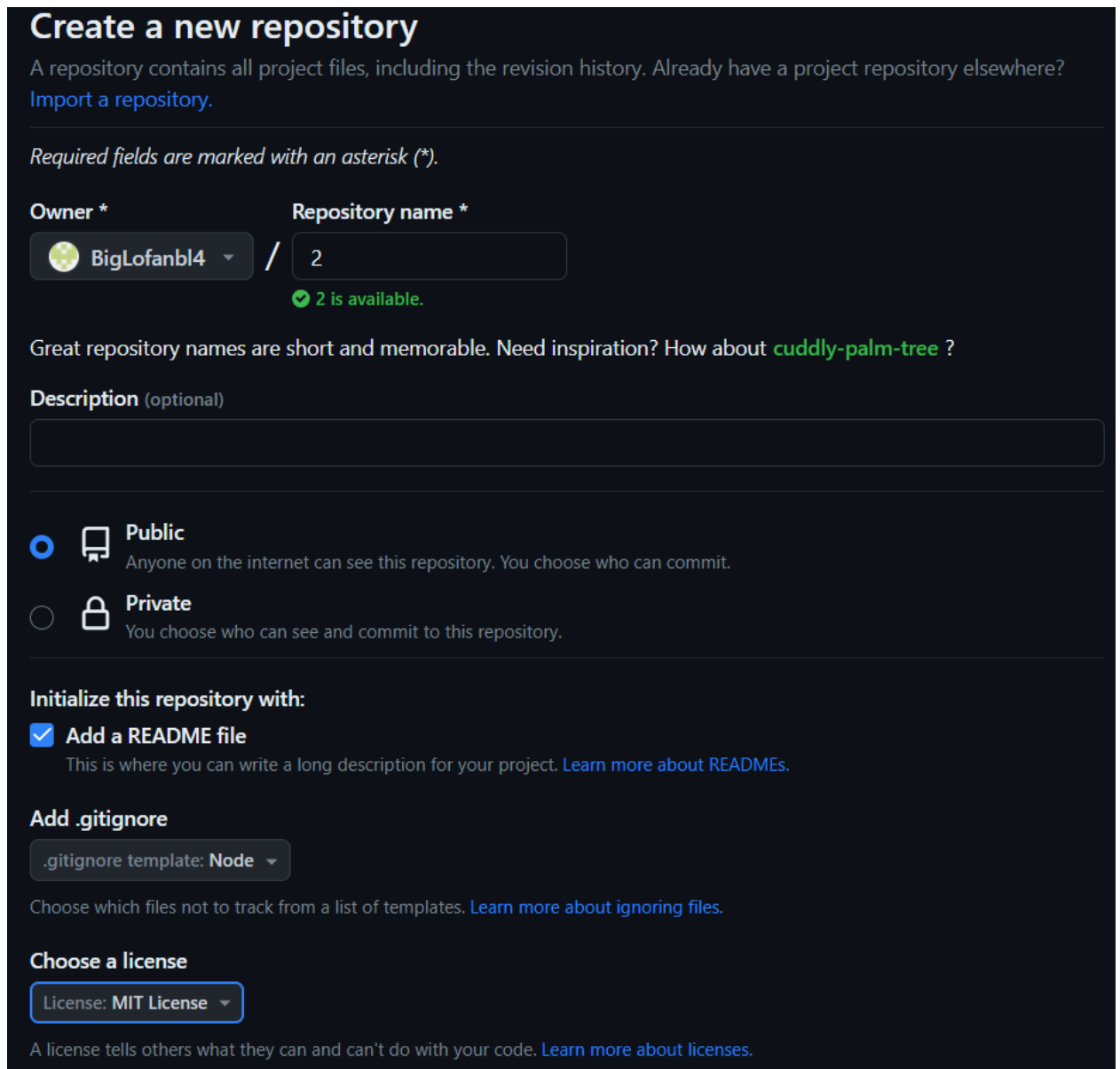
Ставрополь, 2023 г.

Тема: Исследование возможностей Git для работы с локальными репозиториями.

Цель работы: исследовать базовые возможности системы контроля версий Git для работы с локальными репозиториями.

Порядок выполнения работы

1. Создал репозиторий GitHub.



The screenshot shows the GitHub 'Create a new repository' page. At the top, it says 'Create a new repository' and provides a brief explanation of what a repository is. Below this, there's a section for 'Required fields are marked with an asterisk (*)'. The 'Owner' field is set to 'BigLofanbl4' and the 'Repository name' field is set to '2', with a green checkmark indicating '2 is available'. There's a suggestion for repository names: 'Great repository names are short and memorable. Need inspiration? How about cuddly-palm-tree?'. The 'Description' field is optional and currently empty. Under 'Visibility', the 'Public' option is selected, with a description: 'Anyone on the internet can see this repository. You choose who can commit.' The 'Private' option is also available. In the 'Initialize this repository with:' section, the 'Add a README file' checkbox is checked, with a note: 'This is where you can write a long description for your project. Learn more about READMEs.' The 'Add .gitignore' section shows a dropdown menu set to 'Node'. The 'Choose a license' section shows a dropdown menu set to 'MIT License'. At the bottom, there's a note: 'A license tells others what they can and can't do with your code. Learn more about licenses.'

Рисунок 1- Создание репозитория

2. Проработал примеры из лабораторной работы.

```
commit ca82a6dff817ec66f44342007202690a93763949 (HEAD -> master, origin/master, origin/HEAD)
Author: Scott Chacon <schacon@gmail.com>
Date: Mon Mar 17 21:52:11 2008 -0700

    changed the verison number

commit 085bb3bcb608e1e8451d4b2432f8ecbe6306e7e7
Author: Scott Chacon <schacon@gmail.com>
Date: Sat Mar 15 16:40:33 2008 -0700

    removed unnecessary test code

commit a11bef06a3f659402fe7563abf99ad00de2209e6
commit ca82a6dff817ec66f44342007202690a93763949 (HEAD -> master, origin/master, origin/HEAD)
Author: Scott Chacon <schacon@gmail.com>
Date: Mon Mar 17 21:52:11 2008 -0700

    changed the verison number

commit 085bb3bcb608e1e8451d4b2432f8ecbe6306e7e7
Author: Scott Chacon <schacon@gmail.com>
Date: Sat Mar 15 16:40:33 2008 -0700
```

Рисунок 2.1 – История коммитов

```
C:\Users\MSI\Desktop\Основы программной инженерии\2\simplegit-progit>git log -p -2
commit ca82a6dff817ec66f44342007202690a93763949 (HEAD -> master, origin/master, origin/HEAD)
Author: Scott Chacon <schacon@gmail.com>
Date: Mon Mar 17 21:52:11 2008 -0700

    changed the verison number

diff --git a/Rakefile b/Rakefile
index a874b73..8f94139 100644
--- a/Rakefile
+++ b/Rakefile
@@ -5,7 +5,7 @@ require 'rake/gempackagetask'
 spec = Gem::Specification.new do |s|
   s.platform = Gem::Platform::RUBY
   s.name = "simplegit"
-  s.version = "0.1.0"
+  s.version = "0.1.1"
   s.author = "Scott Chacon"
   s.email = "schacon@gmail.com"
   s.summary = "A simple gem for using Git in Ruby code."

commit 085bb3bcb608e1e8451d4b2432f8ecbe6306e7e7
Author: Scott Chacon <schacon@gmail.com>
Date: Sat Mar 15 16:40:33 2008 -0700

    removed unnecessary test code

diff --git a/lib/simplegit.rb b/lib/simplegit.rb
index a0a60ae..47c6340 100644
--- a/lib/simplegit.rb
+++ b/lib/simplegit.rb
@@ -18,8 +18,3 @@ class SimpleGit
end

end
-
-
-if $0 == __FILE__
```

Рисунок 2.2 – Использование команды git log с аргументов -p

```

C:\Users\MSI\Desktop\Основы программной инженерии\2\simplegit-progit>git log --stat
commit ca82a6dff817ec66f44342007202690a93763949 (HEAD -> master, origin/master, origin/HEAD)
Author: Scott Chacon <schacon@gmail.com>
Date: Mon Mar 17 21:52:11 2008 -0700

    changed the verison number

Rakefile | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)

commit 085bb3bcb608e1e8451d4b2432f8ecbe6306e7e7
Author: Scott Chacon <schacon@gmail.com>
Date: Sat Mar 15 16:40:33 2008 -0700

    removed unnecessary test code

lib/simplegit.rb | 5 ----
1 file changed, 5 deletions(-)

commit a11bef06a3f659402fe7563abf99ad00de2209e6
Author: Scott Chacon <schacon@gmail.com>
Date: Sat Mar 15 10:31:28 2008 -0700

    first commit

README          | 6 ++++++
Rakefile        | 23 ++++++++++++++++++++++
lib/simplegit.rb | 25 ++++++++++++++++++++++
3 files changed, 54 insertions(+)

```

Рисунок 2.3 – Сокращенная статистика для каждого коммита

```

C:\Users\MSI\Desktop\Основы программной инженерии\2\simplegit-progit>git log --pretty=oneline
ca82a6dff817ec66f44342007202690a93763949 (HEAD -> master, origin/master, origin/HEAD) changed the verison number
085bb3bcb608e1e8451d4b2432f8ecbe6306e7e7 removed unnecessary test code
a11bef06a3f659402fe7563abf99ad00de2209e6 first commit

```

Рисунок 2.4 – Использование команды git log с аргументом --pretty

```

C:\Users\MSI\Desktop\Основы программной инженерии\2\ticgit>git remote -v
origin https://github.com/schacon/ticgit (fetch)
origin https://github.com/schacon/ticgit (push)

```

Рисунок 2.5 – Просмотр удаленных репозиторий

```

C:\Users\MSI\Desktop\Основы программной инженерии\2\ticgit>git remote add pb https://github.com/paulboone/ticgit
error: remote pb already exists.
error: remote pb already exists.

C:\Users\MSI\Desktop\Основы программной инженерии\2\ticgit>git remote -v
origin https://github.com/schacon/ticgit (fetch)
origin https://github.com/schacon/ticgit (push)
pb     https://github.com/paulboone/ticgit (fetch)
pb     https://github.com/paulboone/ticgit (push)

```

Рисунок 2.6 – Добавление удаленного репозитория

```
C:\Users\MSI\Desktop\Основы программной инженерии\2\ticgit>git tag  
v1.0
```

Рисунок 2.7 – Просмотр списка тэгов

```
C:\Users\MSI\Desktop\Основы программной инженерии\2\ticgit>git tag -a v1.4 -m "my version 1.4"  
  
C:\Users\MSI\Desktop\Основы программной инженерии\2\ticgit>git tag  
v1.0  
v1.4
```

Рисунок 2.8 – Добавление тэга

3. Выполнил клонирование репозитория на рабочий компьютер.

```
C:\Users\MSI\Desktop\Основы программной инженерии\2>git clone https://github.com/BigLofanb14/2.git  
Cloning into '2'...  
remote: Enumerating objects: 5, done.  
remote: Counting objects: 100% (5/5), done.  
remote: Compressing objects: 100% (4/4), done.  
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0  
Receiving objects: 100% (5/5), done.
```

Рисунок 3 – Клонирование репозитория

4. Добавил академические данные в файл README.

```
# 2  
Основы программной инженерии  
Плугатырев Владислав Алексеевич ПИЖ-б-о-22-1
```

Рисунок 4 – Измененный файл README

5. Написал небольшую программу на языке JavaScript.

```
"use strict";

function Calculator() {
  this.methods = {
    "-": (a, b) => a - b,
    "+": (a, b) => a + b,
  };

  this.calculate = function(str) {
    let split = str.split("");
    let a = +split[0];
    let op = split[1];
    let b = +split[2];

    if (!this.methods[op] || isNaN(a) || isNaN(b)) {
      return NaN;
    }

    return this.methods[op](a, b);
  }

  this.addMethod = function(name, func) {
    this.methods[name] = func;
  }
}

let powerCalc = new Calculator;

powerCalc.addMethod("/", (a, b) => a / b);
powerCalc.addMethod("**", (a, b) => a ** b);

let result = powerCalc.calculate("2 ** 3");
```

Рисунок 5 – Написанная программа

6. Просмотрел историю коммитов.

```
C:\Users\MSI\Desktop\Основы программной инженерии\2\2>git log
commit 09a54475690066e229d1edd3323f0636d024de35 (HEAD -> main)
Author: BigLofanbl4 <smallofanbl4@gmail.com>
Date: Tue Sep 19 09:54:30 2023 +0300

    Добавление операции и вывод результатов

commit 093932eb3805d228aa0bf945e87d7ae679abcaaa
Author: BigLofanbl4 <smallofanbl4@gmail.com>
Date: Tue Sep 19 09:53:52 2023 +0300

    Создание функции с помощью функции-конструктора и добавление операции

commit a6b1941c33dc7fd63673f37de800700a37c12b75
Author: BigLofanbl4 <smallofanbl4@gmail.com>
Date: Tue Sep 19 09:50:38 2023 +0300

    Создание функции добавления операции

    Проверка правильности строки
```

Рисунок 6 – История коммитов

7. Просмотрел содержимое коммитов командой git show.

```
C:\Users\MSI\Desktop\Основы программной инженерии\2\2>git show HEAD
commit 09a54475690066e229d1edd3323f0636d024de35 (HEAD -> main)
Author: BigLofanbl4 <smallofanbl4@gmail.com>
Date: Tue Sep 19 09:54:30 2023 +0300
```

Добавление операции и вывод результатов

```
diff --git a/script.js b/script.js
index 67a7213..29b8767 100644
--- a/script.js
+++ b/script.js
@@ -27,4 +27,6 @@ function Calculator() {
    let powerCalc = new Calculator;

    powerCalc.addMethod("/", (a, b) => a / b);
+powerCalc.addMethod("**", (a, b) => a ** b);

+let result = powerCalc.calculate("2 ** 3");
```

Рисунок 7.1 – Содержимое последнего коммита

```
C:\Users\MSI\Desktop\Основы программной инженерии\2\2>git show HEAD~1
commit 093932eb3805d228aa0bf945e87d7ae679abcaaa
Author: BigLofanbl4 <smallofanbl4@gmail.com>
Date: Tue Sep 19 09:53:52 2023 +0300
```

Создание функции с помощью функции-конструктора и добавление операции

```
diff --git a/script.js b/script.js
index 2fa0134..67a7213 100644
--- a/script.js
+++ b/script.js
@@ -21,6 +21,10 @@ function Calculator() {

    this.addMethod = function(name, func) {
        this.methods[name] = func;
+    }
+}
-}
+
+let powerCalc = new Calculator;
+
+powerCalc.addMethod("/", (a, b) => a / b);
```

Рисунок 7.2 – Содержимое предпоследнего коммита

```

C:\Users\MSI\Desktop\Основы программной инженерии\2\2>git show 2cf8309eac5ff3845e8eea4701b5c3830269e6ad
commit 2cf8309eac5ff3845e8eea4701b5c3830269e6ad
Author: BigLofanbl4 <smallofanbl4@gmail.com>
Date: Tue Sep 19 09:46:57 2023 +0300

    Разбиение строки на переменные и операцию

diff --git a/script.js b/script.js
index 790ecdd..047c43f 100644
--- a/script.js
+++ b/script.js
@@ -6,6 +6,11 @@ function Calculator() {
    "+": (a, b) => a + b,
    };

-
+ this.calculate = function(str) {
+   let split = str.split("");
+   let a = +split[0];
+   let op = split[1];
+   let b = +split[2];
+ }
+ }

```

Рисунок 7.3 – Содержимое коммита с указанным хэшем

8. Освоил возможность отката к заданной версии.

```

C:\Users\MSI\Desktop\Основы программной инженерии\2\2>git add .

C:\Users\MSI\Desktop\Основы программной инженерии\2\2>git commit -m "Выполнение отката к заданной версии"
[main abf78c9] Выполнение отката к заданной версии
1 file changed, 32 deletions(-)

C:\Users\MSI\Desktop\Основы программной инженерии\2\2>git reset --hard HEAD~1
HEAD is now at 09a5447 Добавление операции и вывод результатов

```

Рисунок 8 – Откат к предпоследнему коммиту

Ответы на контрольные вопросы

1. С помощью команды `git log`. Существуют множество дополнительных опций вот некоторые из них:

- `-p` или `-patch` – показывает разницу, внесенную в каждый коммит;
 - `--stat` – позволяет увидеть сокращенную статистику;
 - `--pretty` – эта опция меняет формат вывода;
 - `--since` и `--until` – опции для ограничения вывода по времени
2. С помощью команды `git log` и его аргументов, например `--since`.
3. С помощью команды `git commit --amend`.
4. С помощью команды `git reset HEAD`.

5. С помощью команды `git checkout -- <file>`.
6. Удалённые репозитории представляют собой версии вашего проекта, сохранённые в интернете или ещё где-то в сети.
7. С помощью команды `git remote -v`.
8. С помощью команды `git remote add <shortname> <url>`.
9. Получение изменение – `git fetch [remote-name]`, отправка изменений `git push <remote-name> <branch-name>`.
10. С помощью команды `git remote show <remote>`.
11. Тэги Git - это ссылки на определенные коммиты в истории разработки. Они используются для пометки определенных версий или моментов в вашем проекте. Тэги обычно используются для обозначения релизов или важных этапов в разработке.
12. Для просмотра тэгов – `git tag`, для создания аннотированного тэга – `git tag -a v1.4 -m "сообщение"`, для отправки тэга на удаленный сервер – `git push origin <tagname>`, для удаления тэгов – `git tag -d <tagname>`.
13. Флаг `--prune` в командах `git fetch` и `git push` используется для удаления удаленных веток или тэгов, которые больше не существуют на удаленном репозитории. Если выполнить команду `git fetch --prune`, Git удалит локальные ссылки на удаленные ветки и тэги, которые были удалены на удаленном репозитории. Если выполнить команду `git push --prune`, Git удалит удаленные ветки и тэги на удаленном репозитории, которых больше нет в вашем локальном репозитории.