

Удаленный контроль версий исходного кода проекта на Bitbucket

Стрельцова Марина Николаевна

Приамурский государственный университет им. Шолом-Алейхема

Студент

Аннотация

В настоящее время разработка информационных проектов осуществляется в большинстве случаев командой людей и для удобства их взаимодействия появляется необходимость в использование дополнительных сервисов для проектов. В данной статье рассматривается применение веб-сервиса Bitbucket. С помощью программы Git Bush настроена удаленная загрузка и редактирование исходного кода проекта.

Ключевые слова: Bitbucket, удаленный и локальный репозитории, Git Bush

Remote version control of the project source code on Bitbucket

Streltsova Marina Nikolaevna

Sholom-Aleichem Priamursky State University

Student

Abstract

Currently, the development of information projects is carried out in most cases by a team of people and for the convenience of their interaction, there is a need to use additional services for projects. This article discusses the use of the Bitbucket web service. Using the Git Push program, you can remotely download and edit the project source code.

Keywords: Bitbucket, remote and local repositories, Git Bash

На сегодняшний день актуальна является разработка проектов во всех сферах деятельности. А правильно и грамотно управлять проектами можно с помощью онлайн-сервисов. Например, для проектов в сфере информационных технологий требуется сервис в котором можно вести разработку параллельно всем участникам команды и такая возможность имеется в сервисе Bitbucket [1]. Bitbucket — это инструмент управления кодом в Git. Он предоставляет командам единый центр для планирования проектов, совместной работы над кодом, тестирования и развертывания.

В статье Ю. В. Торкунова, Н. Н. Халимова рассматривается среда коллективного программирования, показываются ее преимущества и возможности использования при обучении студентов [2]. С. Treude и др. описывают исследование о необычных событиях в репозиториях Github [3]. В исследовании V. Salis и D. Spinellis предоставляется инструмент с открытым исходным кодом, который позволит интуитивно и эффективно

анализировать репозитории Git, а также иллюстрируют методы анализа, используемых этим инструментом [4]. А. А. Ефимов в статье рассматривает основные концепции и возможности инструмента контроля качества кода SonarQube [5]. О способах поиска по примеру в репозиториях SQL с использованием структурного и интенционального сходства рассказывают в своей статье G. Borodin и Y. Kanza [6]. В научной работе A. Goyal, N. Sardana описаны характеристики производительности процесса исправления ошибок в репозиториях ошибок с открытым исходным кодом [7]. И.В. Бахтин в своей статье описывает обзор веб-сервиса GitHub, показывает преимущества сервиса в процессе организации работы работников в сфере информационных технологий [8].

Целью данной статьи является удаленная загрузка и редактирование исходного кода проекта на Bitbucket.

В начале работы с веб-сервисом Bitbucket необходимо авторизоваться (Рис.1).

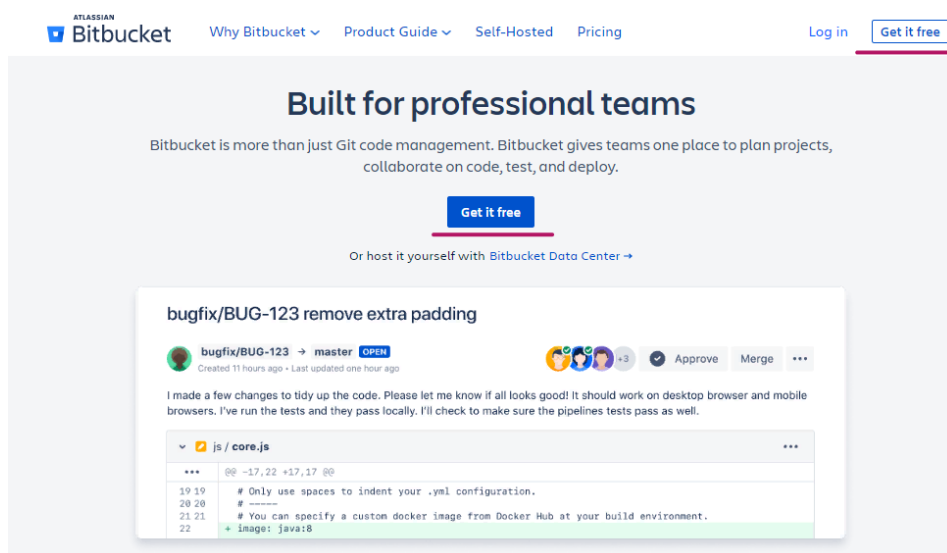


Рисунок 1 – Главная страница веб-сервиса bitbucket

После авторизации создаем новый проект. Во вкладке Project необходимо в правом верхнем углу нажать Create project (Рис.2).

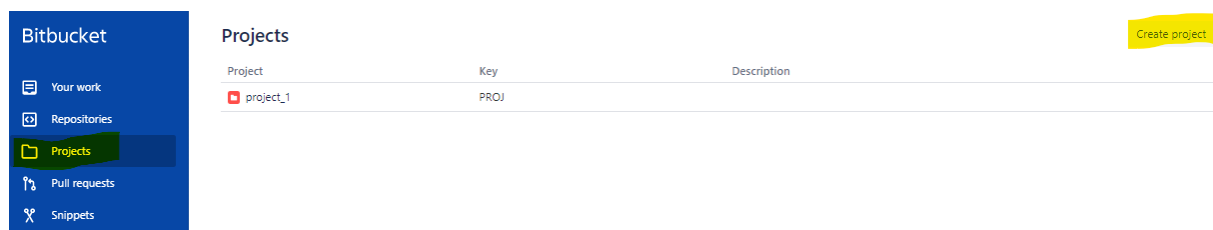


Рисунок 2 – Создание проекта

Далее создаем репозиторий. Во вкладке Repositories необходимо в правом верхнем углу нажать Create Repository (Рис.3).

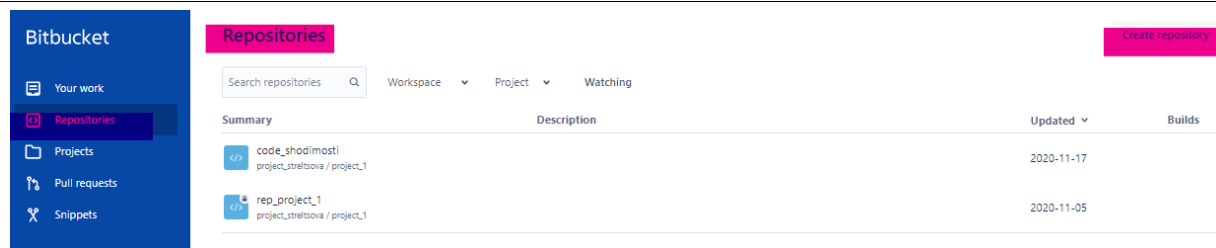


Рисунок 3 – Создание репозитория

Для работы с исходным кодом проекта и последующей синхронизации локального репозитория с репозиторием на веб-сервисе скачиваем и устанавливаем Git Bush [9].

Для начала работы в программе Git Bush прописываем следующие команды, которые показаны на рисунке 4. Данные команды прописываются для идентификации пользователя.

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты (master)
$ git config --global user.email "streltsovamarinochka@yandex.ru"

admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты (master)
$ git config --global user.name "MNStreltsova"
```

Рисунок 4 – Команды для идентификации пользователя

После необходимо выбрать директорию с проектом (Рис.5).

```
admin@DESKTOP-2IHOHME MINGW64 ~ (master)
$ cd 'студия красоты'/'
```

Рисунок 5 – Выбор директории проекта

Далее прописываем код, который будет инициализировать git проект (Рис.6).

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты (master)
$ git init
Initialized empty Git repository in C:/Users/админ/студия красоты/.git/
```

Рисунок 6 – Код для инициализирования

Код git add. добавит в git проект все папки и файлы текущей директории (Рис.7).

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты (master)
$ git add .
warning: LF will be replaced by CRLF in media-queries.css.
The file will have its original line endings in your working directory
```

Рисунок 7 – Код git add

Следующая команда git commit, которая делает для проекта снимок текущего состояния изменений (Рис.8).

```

admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты (master)
$ git commit -m "initial commit"
[master (root-commit) d53b99c] initial commit
14 files changed, 381 insertions(+)
 create mode 100644 2.jpg
 create mode 100644 brovi.jpg
 create mode 100644 glavnaya.htm
 create mode 100644 karta.png
 create mode 100644 kontakt.htm
 create mode 100644 makeup.jpg
 create mode 100644 media-queries.css
 create mode 100644 nogti.jpg
 create mode 100644 prais.htm
 create mode 100644 reset.css
 create mode 100644 resnica.jpg
 create mode 100644 shuga.jpg
 create mode 100644 style.css
 create mode 100644 "\320\235\320\276\320\262\321\213\320\271 \321\202\320\265\320\272\321\201\321\202\320\276\320\262\321\213\320\271 \320\264\320\276\320\272\321\203\320\274\320\265\320\275\321\202.txt"

```

Рисунок 8 – Команда git commit

Далее для синхронизации двух репозиторий (локального и удаленного) прописываем команду `git pull origin master --allow-unrelated-histories` (Рис.9).

```

admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты (master)
$ git pull origin master --allow-unrelated-histories
From https://bitbucket.org/project-streltsova/code_shodimosti
 * branch                master       -> FETCH_HEAD
Merge made by the 'recursive' strategy.
 .gitignore | 50 ++++++
 README.md  | 45 ++++++
 2 files changed, 95 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 README.md

```

Рисунок 9 – Синхронизация репозиторий

Для следующей команды надо скопировать URL с Bitbucket вашего репозитория (Рис.10).



Рисунок 10 – Копирование URL

В командной строке печатаем `git remote add origin + URL` (Рис.11). Данная команда записывает URL в `origin` для того чтобы каждый раз не прописывать свой URL.

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты (master)
$ git remote add origin https://MNStreltsova@bitbucket.org/project-streltsova/code_shodimosti.git
```

Рисунок 11 – Команда `git remote add origin`

Следующим шагом прописываем команду, которая позволяет внести изменения в проект на удаленном компьютере (Рис.12).

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты (master)
$ git push origin master
Enumerating objects: 19, done.
Counting objects: 100% (19/19), done.
Delta compression using up to 4 threads
Compressing objects: 100% (17/17), done.
Writing objects: 100% (18/18), 1.38 MiB | 184.00 KiB/s, done.
Total 18 (delta 1), reused 0 (delta 0), pack-reused 0
To https://bitbucket.org/project-streltsova/code_shodimosti.git
 5de4011..1341252 master -> master
```

Рисунок 12 – Команда для внесения изменений

Результатом всех этих команд является синхронизация локального и удаленного проекта (Рис.13,14).

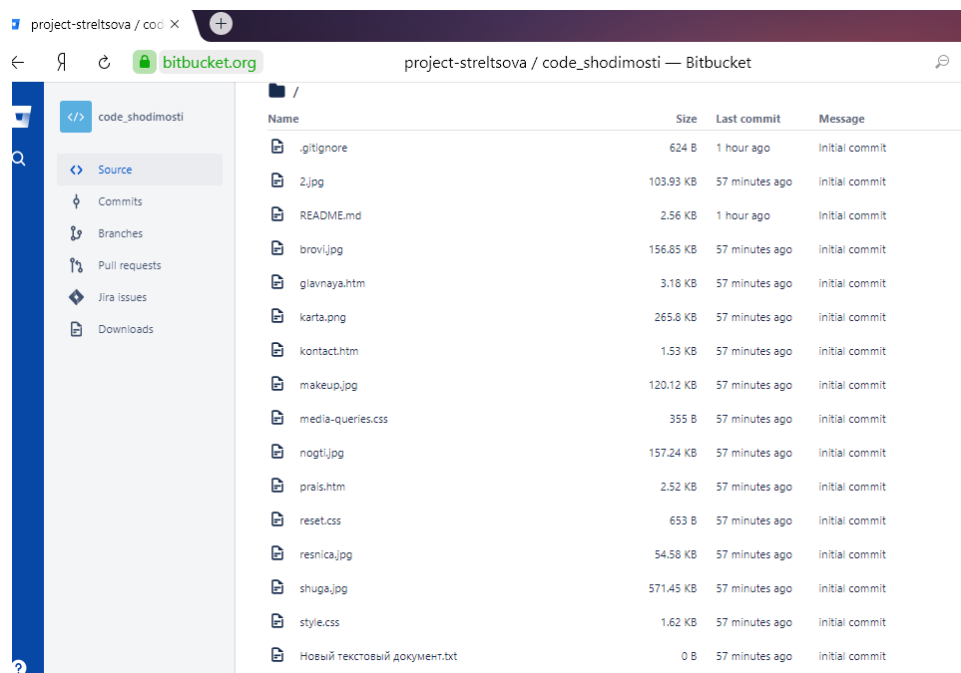


Рисунок 13 – Результат выполнения команд на удаленном репозитории

компьютер > OS (C:) > Пользователи > админ > студия красоты

Имя	Дата изменения
.gitignore	17.11.2020 21:23
2.jpg	09.04.2019 17:16
brovi.jpg	26.11.2018 14:38
glavnaya.htm	08.11.2019 16:02
karta.png	10.04.2019 20:16
kontakt.htm	01.11.2019 15:08
makeup.jpg	27.11.2018 9:47
media-queries.css	29.08.2011 23:25
nogti.jpg	25.11.2018 22:12
prais.htm	18.12.2019 22:03
README.md	17.11.2020 21:23
reset.css	11.08.2011 18:19
resnica.jpg	09.04.2019 16:15
shuga.jpg	09.04.2019 16:18
style.css	01.11.2019 15:07
Новый текстовый документ.txt	09.04.2019 14:54

Рисунок 14 – Результат выполнения команд на локальном репозитории

Изменим имя одного из файлов, добавим новый файл и удалим существующий файл для проверки работоспособности проекта. После изменения имени файла и добавление нового файла следует прописывать команду `git add` “Имя файла” (Рис.15). А после удаления следует прописать команду `git rm` “Имя файла” (Рис.16).

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты/code_shodimosti (master)
$ git add '12.txt'

admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты/code_shodimosti (master)
$ git add '1.txt'
```

Рисунок 15 – Команда git add

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты/code_shodimosti (master)
$ git rm 'karta.png'
rm 'karta.png'
```

Рисунок 16 – Команда git rm

Проверим статус измененных и удаленных файлов командой `git status` (Рис.17).

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты/code_shodimosti (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
   new file:   1.txt
   new file:   12.txt
   deleted:    karta.png
```

Рисунок 17 – Команда git status

Далее закоммитим измененные файлы командой `git commit` (Рис.18).

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты/code_shodimosti (master)
$ git commit
Aborting commit due to empty commit message.
```

Рисунок 18 – Команда `git commit`

Затем следует синхронизироваться с удаленным проектом командой `git pull origin master --allow-unrelated-histories` (Рис.19).

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты/code_shodimosti (master)
$ git pull origin master --allow-unrelated-histories
From https://bitbucket.org/project-streltsova/code_shodimosti
 * branch                master          -> FETCH_HEAD
Already up to date.
```

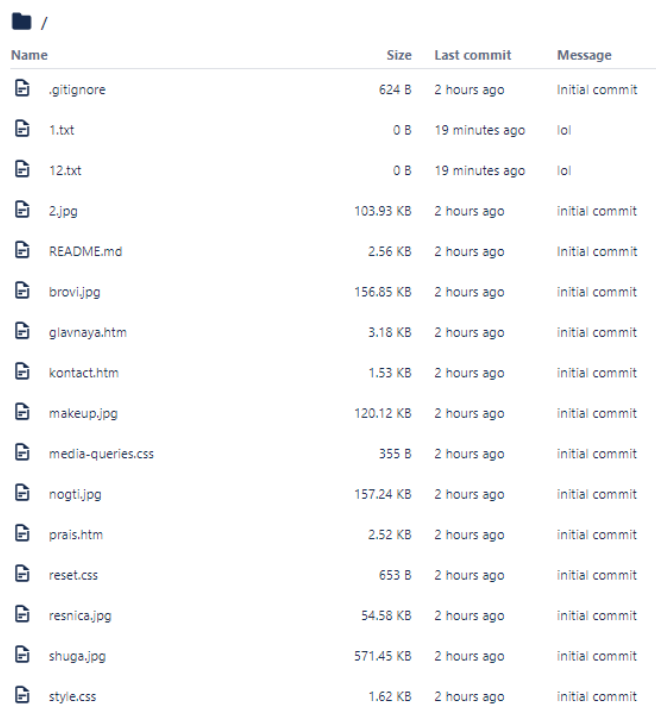
Рисунок 19 – Синхронизация с удаленным проектом

Последним шагом будет загрузка изменений в проекте на удаленный компьютер (Рис.20).

```
admin@DESKTOP-2IHOHME MINGW64 ~/студия красоты/code_shodimosti (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 250 bytes | 125.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
To https://bitbucket.org/project-streltsova/code_shodimosti.git
 1341252..049f34f master -> master
```

Рисунок 20 – Загрузка изменений

Проверим изменения на сайте и удостоверимся в их загрузке (Рис.21).



Name	Size	Last commit	Message
.gitignore	624 B	2 hours ago	initial commit
1.txt	0 B	19 minutes ago	lol
12.txt	0 B	19 minutes ago	lol
2.jpg	103.93 KB	2 hours ago	initial commit
README.md	2.56 KB	2 hours ago	initial commit
brovi.jpg	156.85 KB	2 hours ago	initial commit
glavnaya.htm	3.18 KB	2 hours ago	initial commit
kontakt.htm	1.53 KB	2 hours ago	initial commit
makeup.jpg	120.12 KB	2 hours ago	initial commit
media-queries.css	355 B	2 hours ago	initial commit
nogti.jpg	157.24 KB	2 hours ago	initial commit
prais.htm	2.52 KB	2 hours ago	initial commit
reset.css	653 B	2 hours ago	initial commit
resnica.jpg	54.58 KB	2 hours ago	initial commit
shuga.jpg	571.45 KB	2 hours ago	initial commit
style.css	1.62 KB	2 hours ago	initial commit

Рисунок 21 – Проверка изменений на веб-сервисе

В ходе написания данной исследовательской работы были изучены принципы работы с веб-сервисом Bitbucket. С помощью программы Git Bush была настроена синхронизация локального и удаленного репозитория. А так же были проведены эксперименты над изменениями данных в локальном репозитории, а после синхронизация данных изменений в удаленный репозиторий. Bitbucket является удобным веб-сервисом для управления проектами всеми членами команды.

Библиографический список

1. Bitbucket URL: <https://bitbucket.org/> (дата обращения: 26.12.2020).
2. Торкунова Ю. В., Халимов Н. Н. Организация совместной разработки веб-приложений в рамках веб-сервиса github //Наука и практика в решении стратегических и тактических задач устойчивого развития россии. 2019. С. 26-27.
3. Treude C., Leite L., Aniche M. Unusual events in GitHub repositories //Journal of Systems and Software. 2018. Т. 142. С. 237-247.
4. Salis V., Spinellis D. RepoFS: File system view of Git repositories //SoftwareX. 2019. Т. 9. С. 288-292.
5. Ефимов А. А. Применение инструмента контроля качества кода SonarQube в крупных проектах //Вектор экономики. 2020. №. 5. С. 49-49.
6. Borodin G., Kanza Y. Search-by-example over SQL repositories using structural and intent-driven similarity //Data & Knowledge Engineering. 2020. С. 101811.
7. Goyal A., Sardana N. Performance Assessment of Bug Fixing Process in Open Source Repositories //Procedia Computer Science. 2020. Т. 167. С. 2070-2079.

8. Бахтин И.В. Организация совместной работы с использованием веб-сервиса github // Форум молодых ученых. 2019. №2 (30). с. 268-271.
9. Git for windows URL: <https://gitforwindows.org/> (дата обращения: 26.12.2020).