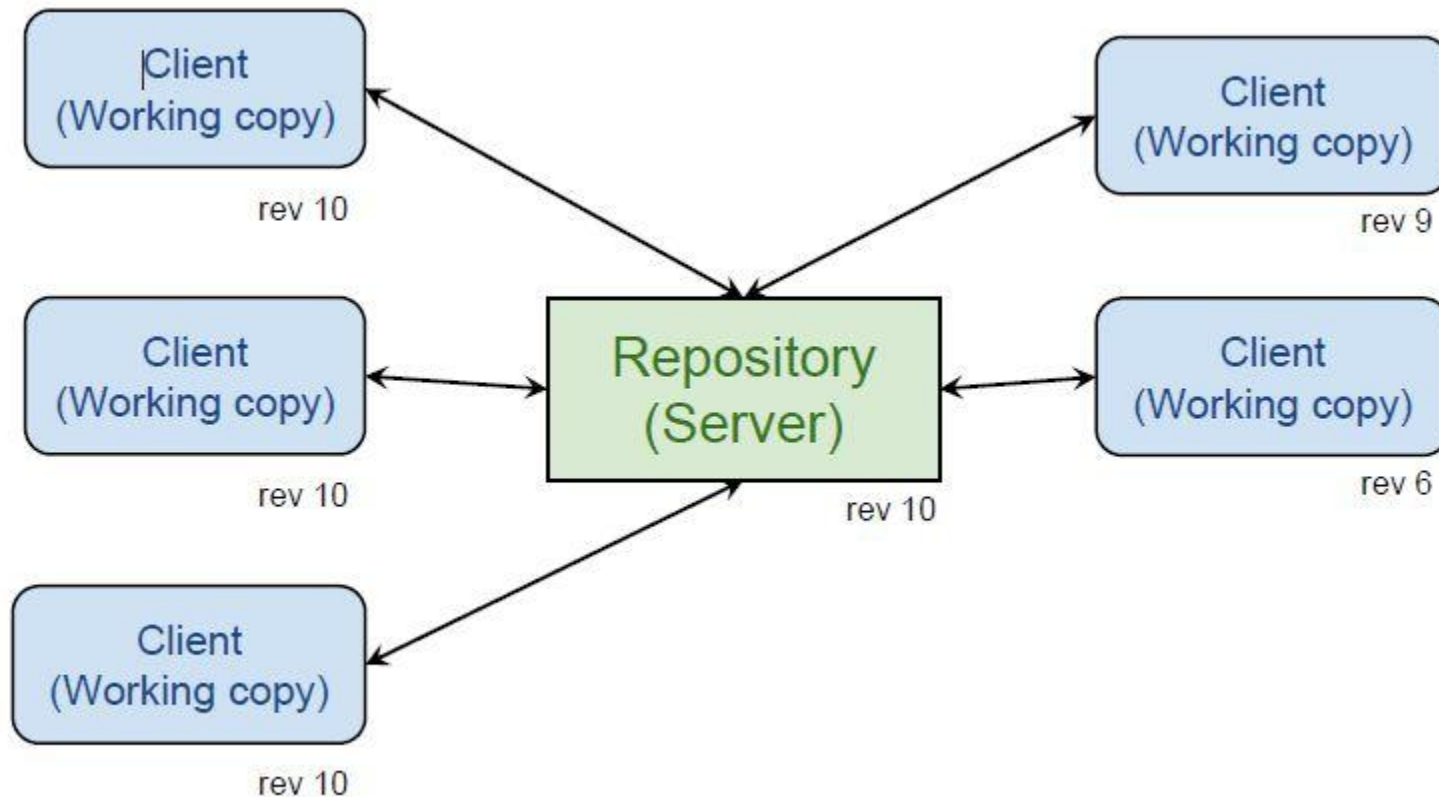


Система
КОНТРОЛЯ
версий
Git

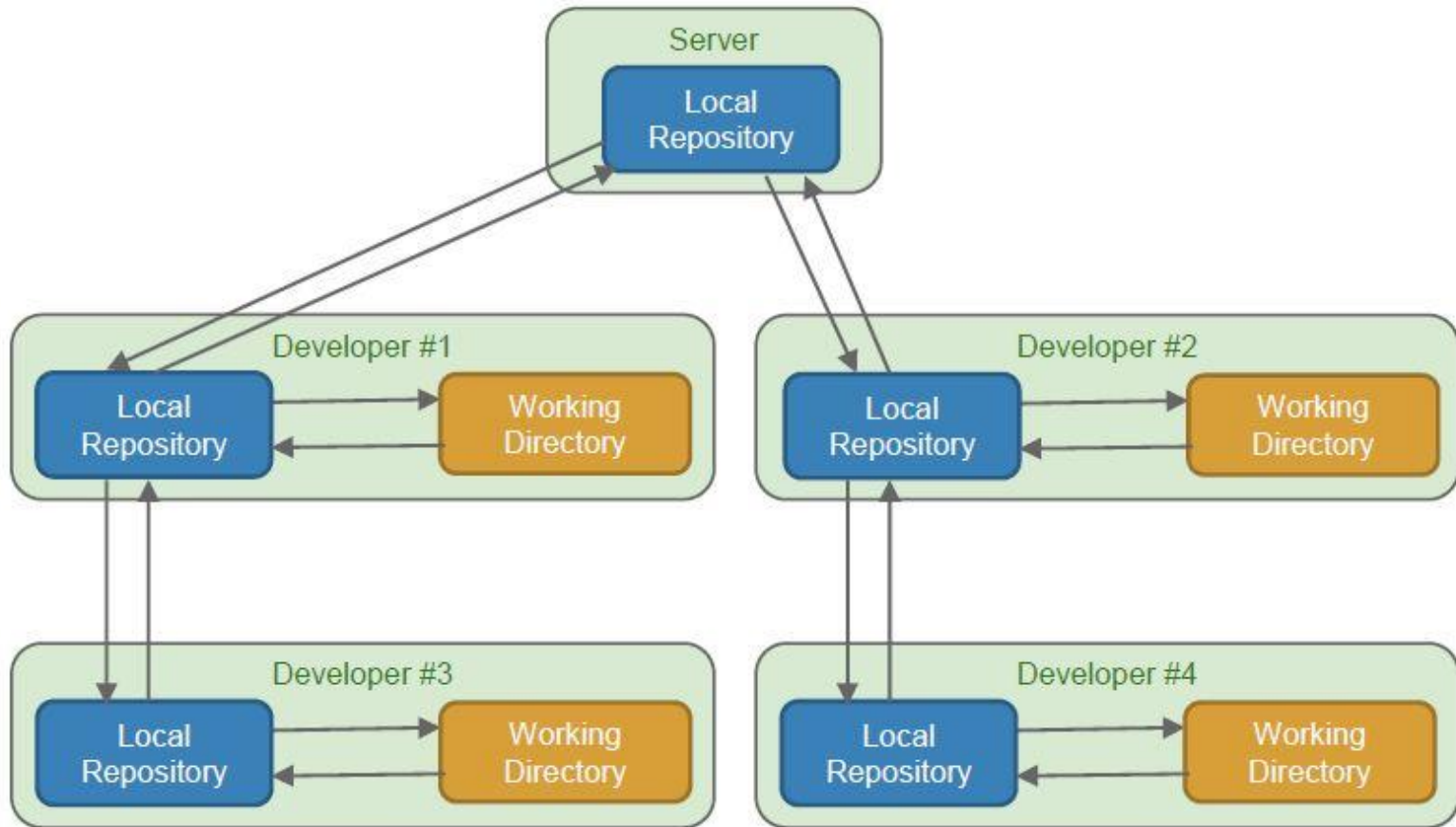
Централизованные VCS



Централизованные VCS

- Достоинства
 - Централизованное администрирование
 - Привычный workflow
 - Управление правами доступа
- Недостатки
 - Единая точка отказа – сервер
 - Любые изменения влияют на всех пользователей
 - Неудобная работа с ветками – легко создать, тяжело сдвинуть
 - Возможны блокировки - захват пессимистической блокировки одним пользователем

Распределенные VCS



Распределенные VCS

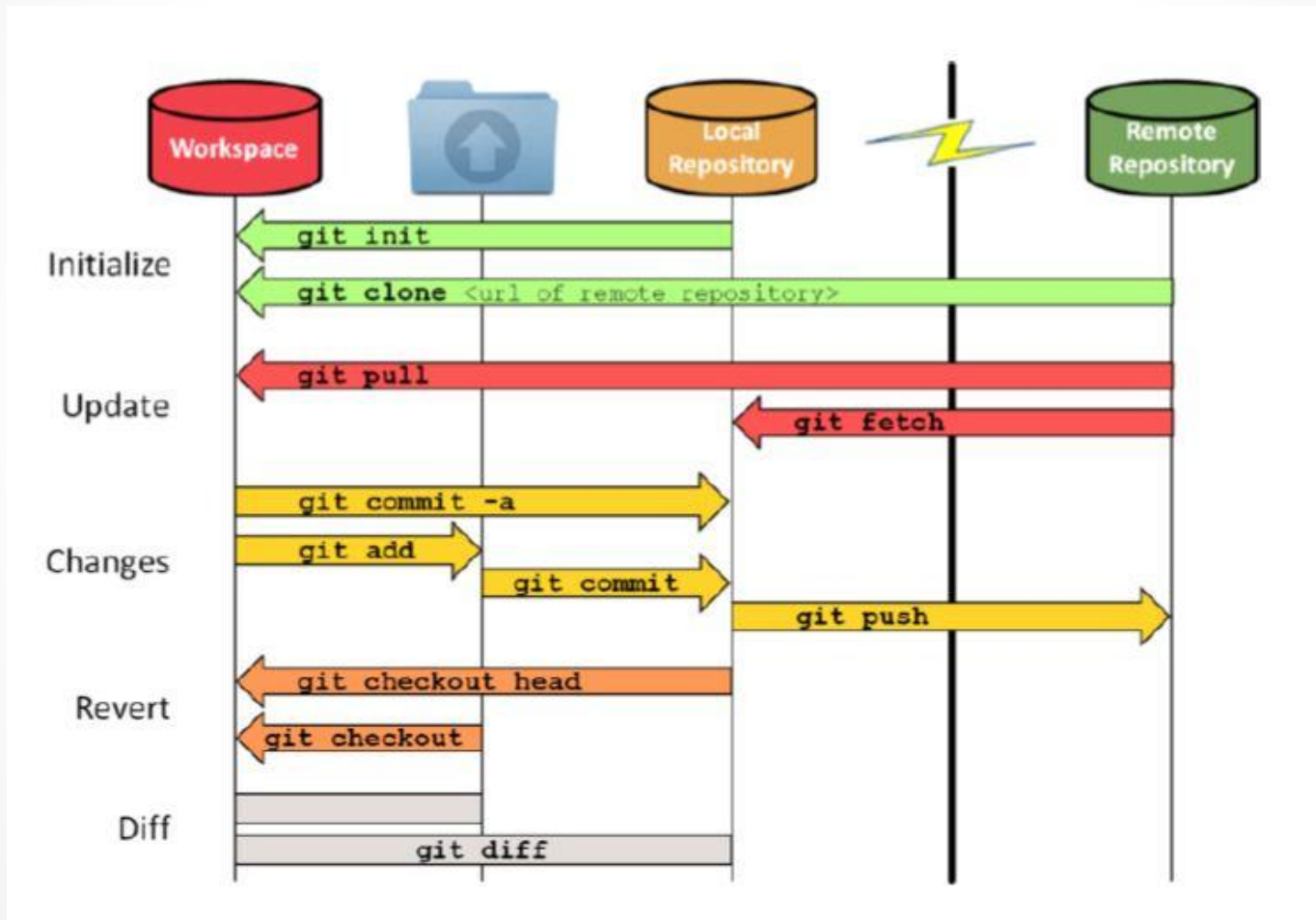
- Достоинства

- Гибкая работа с ветками
- Автономность (как каждого разработчика, так и от сервера вообще)
- Сборка артефактов отделена от разработки
- Локальные операции работают быстро
- Разделены операции фиксации изменений (commit) и публикации изменений(push)

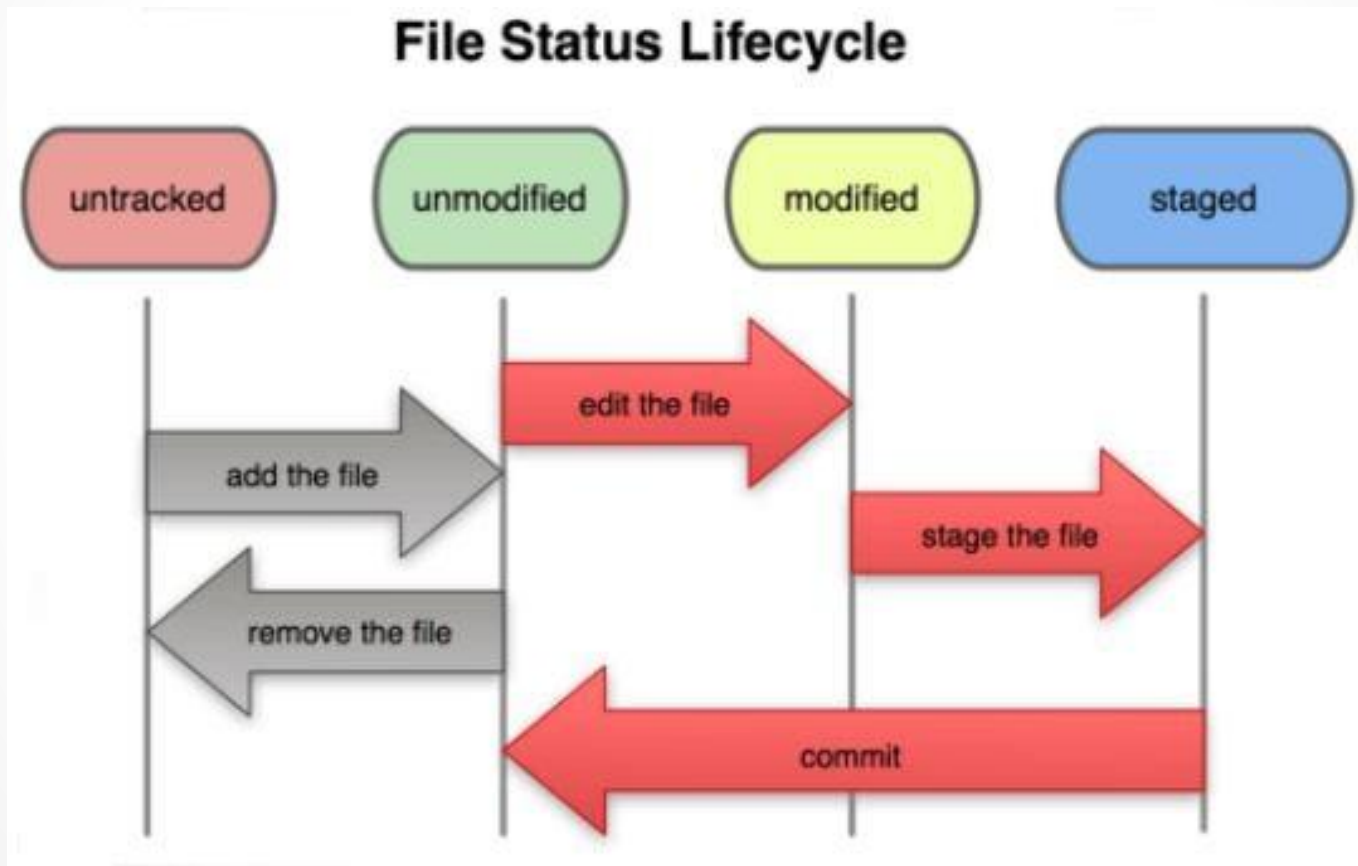
- Недостатки

- В каждой копии необходимо хранить всю историю изменений (иногда считается плюсом)
- Требуется более тщательное управление доступом (иногда считается плюсом)
- Сложны в использовании

Жизненный цикл Git



Состояния файлов



Работа с удаленным репозиторием

- Команда **clone** клонирует репозиторий и создаёт рабочую копию
- Команда **push** отправляет изменения в удаленный репозиторий
- Команда **pull** забирает изменения указанной ветки из удаленного репозитория и сливает их в текущую ветку
- Команда **fetch** забирает все изменения из удаленного репозитория

Работа с локальным репозиторием

- Команда **add** добавляет измененные файлы в stage
- Команда **rm** помечает файл в stage как удаленный
- Команда **reset** сбрасывает изменения в текущем stage
- Команда **commit** сохраняет текущий stage в локальный репозиторий

Работа с ветками

- Команда **branch** создаёт ветку
- Команда **checkout** переключает рабочую копию на другую ветку
- Команда **merge** сливает изменения веток
- Команда **stash** помещает изменения из stage во временное хранилище и сбрасывает рабочую копию

Полезные ссылки

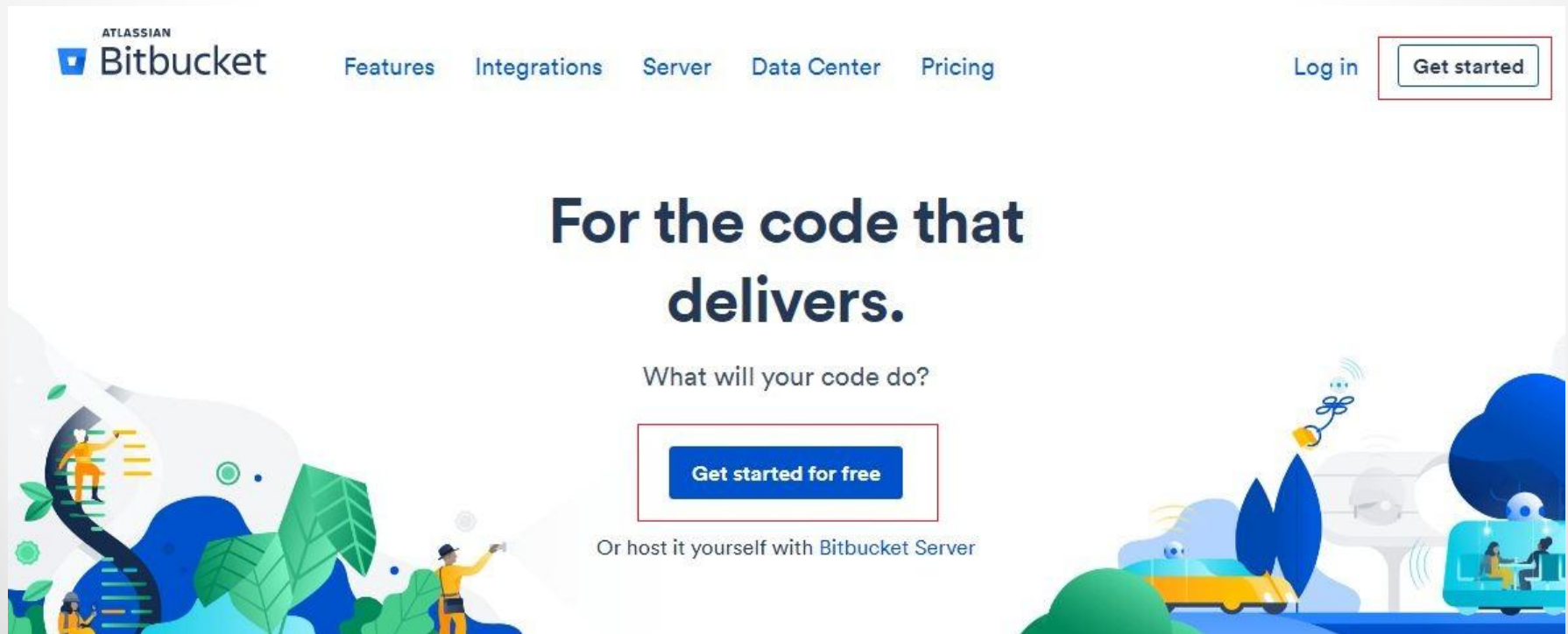
- <http://proselyte.net/tutorials/git/introduction/> - русскоязычная документация, начальные шаги
- <https://git-scm.com/book/en/v2> - основательная документация, почти на все случаи жизни

Bitbucket

- <https://bitbucket.org/>
- веб-сервис для хостинга проектов и их совместной разработки, основанный на системе контроля версий Mercurial и Git.

Регистрация

Заходим на <https://bitbucket.org/>



The image shows the Bitbucket website homepage. At the top left is the Atlassian Bitbucket logo. To its right are navigation links: Features, Integrations, Server, Data Center, and Pricing. On the top right, there is a 'Log in' link and a 'Get started' button. The main content area features the headline 'For the code that delivers.' followed by the question 'What will your code do?'. Below this is a blue 'Get started for free' button, which is highlighted with a red border. Underneath the button is the text 'Or host it yourself with Bitbucket Server'. The page is decorated with colorful illustrations of people working, trees, and a car on the left, and a car and a person on the right.

ATLASSIAN
Bitbucket

Features Integrations Server Data Center Pricing

Log in **Get started**

For the code that delivers.

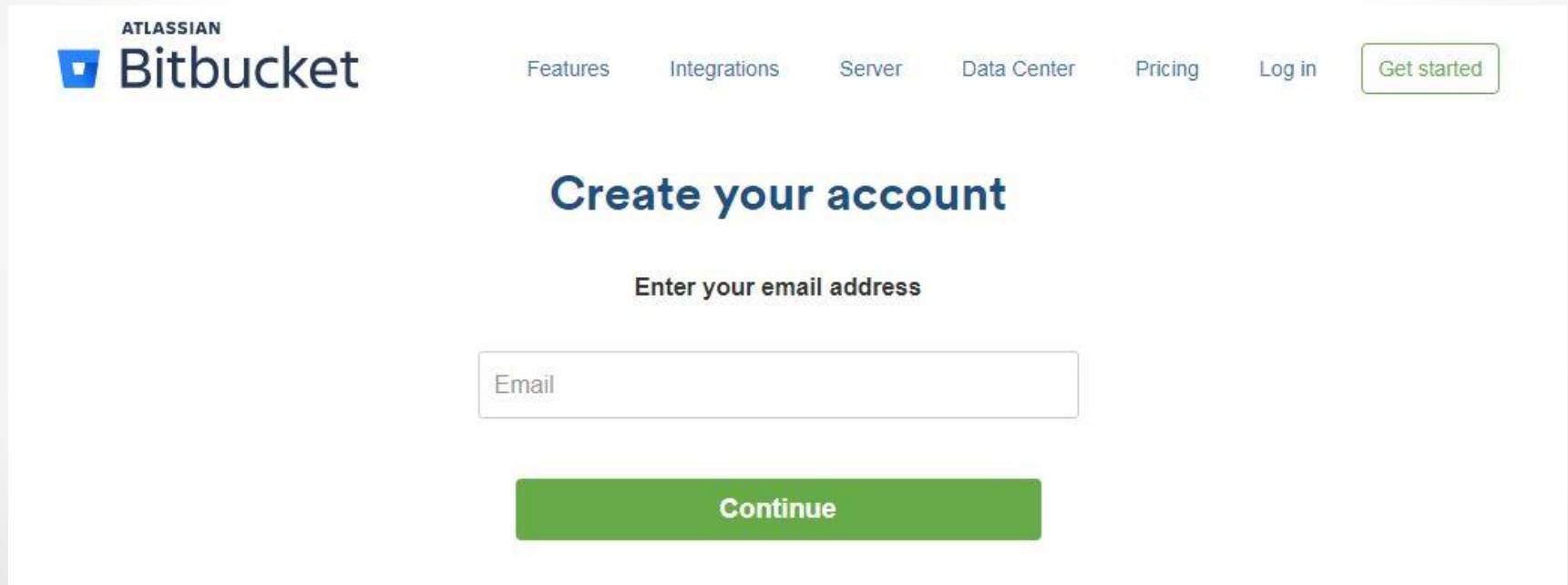
What will your code do?

Get started for free

Or host it yourself with Bitbucket Server

Регистрация

Вводим email, подтверждаем его



The image shows a screenshot of the Bitbucket registration page. At the top left is the Atlassian Bitbucket logo. The navigation menu includes links for Features, Integrations, Server, Data Center, Pricing, and Log in, along with a 'Get started' button. The main heading is 'Create your account', followed by the instruction 'Enter your email address'. Below this is a text input field labeled 'Email' and a green 'Continue' button.

ATLASSIAN
Bitbucket

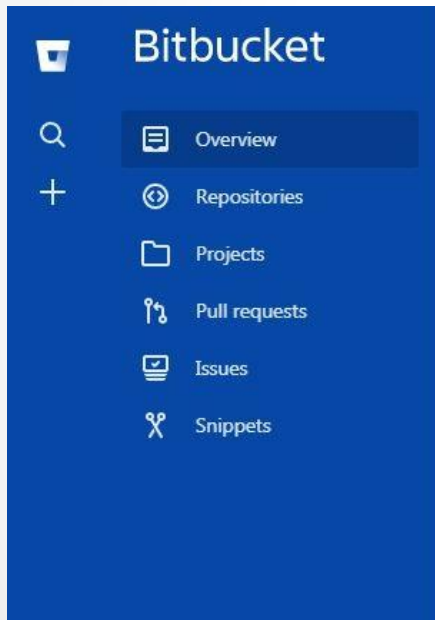
Features Integrations Server Data Center Pricing Log in [Get started](#)

Create your account

Enter your email address

[Continue](#)

Создание репозитория



Bitbucket

- Overview
- Repositories
- Projects
- Pull requests
- Issues
- Snippets

Dashboard



Everything's awesome!

All your pull requests and reviews are done and dusted.

[View all pull requests](#)



Your repository list is empty

Repositories that you own or watch will show up here.

[Create a repository](#)

Создание репозитория

Create a new repository Import repository

Repository name ^{*}

Access level This is a private repository

Repository type Git Mercurial

▼ Advanced settings

Description

Forking ▼

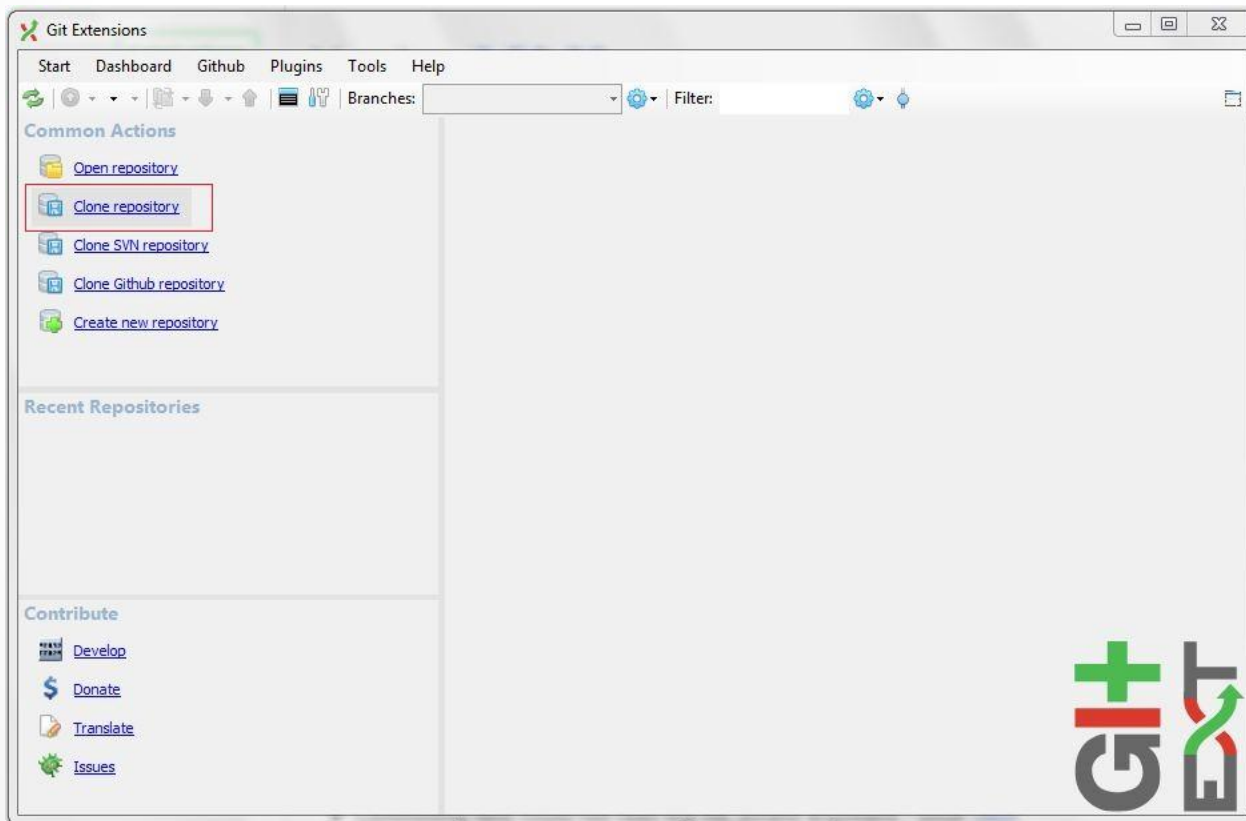
Project management Issue tracking Wiki

Language × ▼

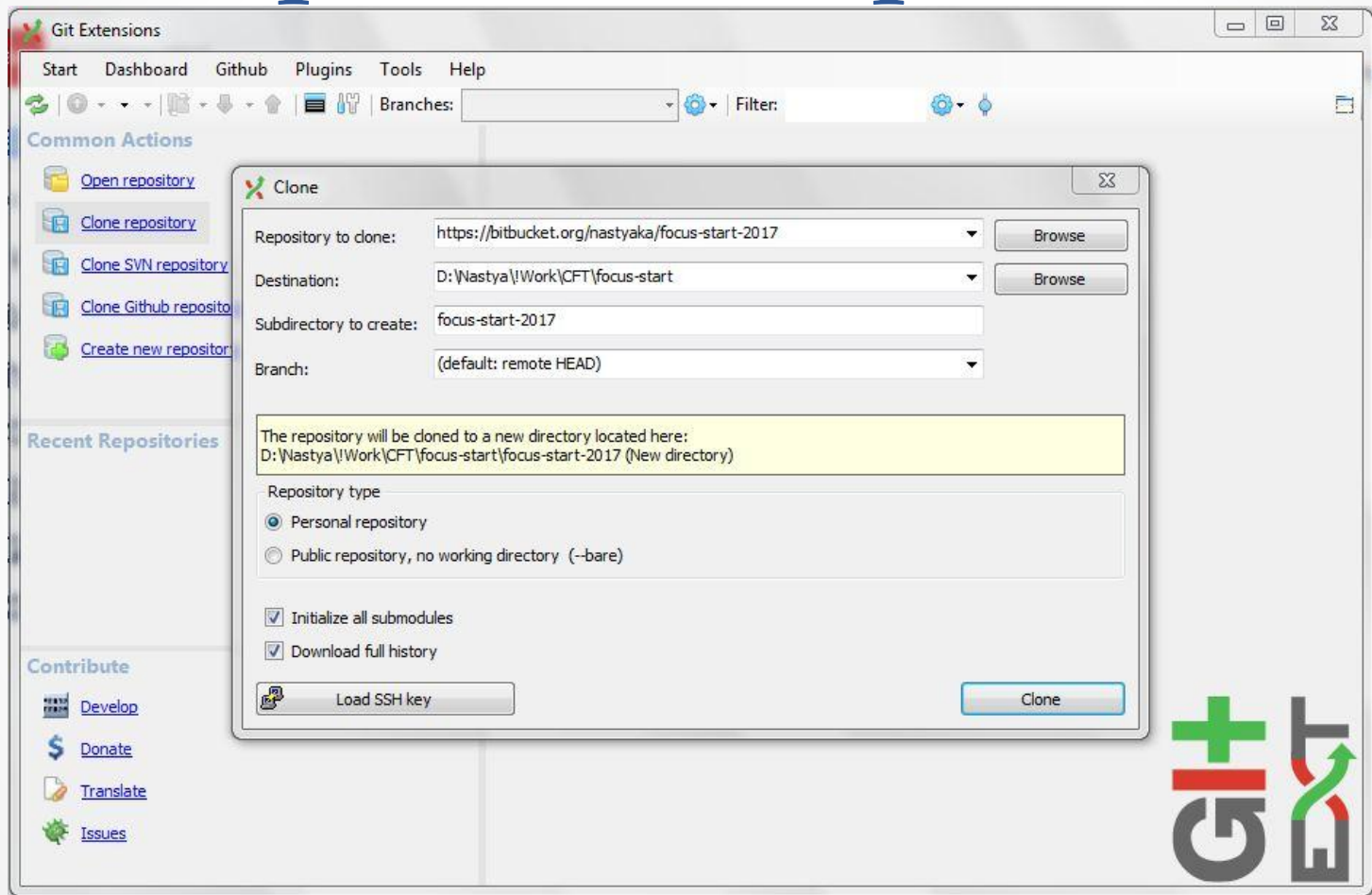
Integrations Enable Hipchat notifications

Клонирование репозитория

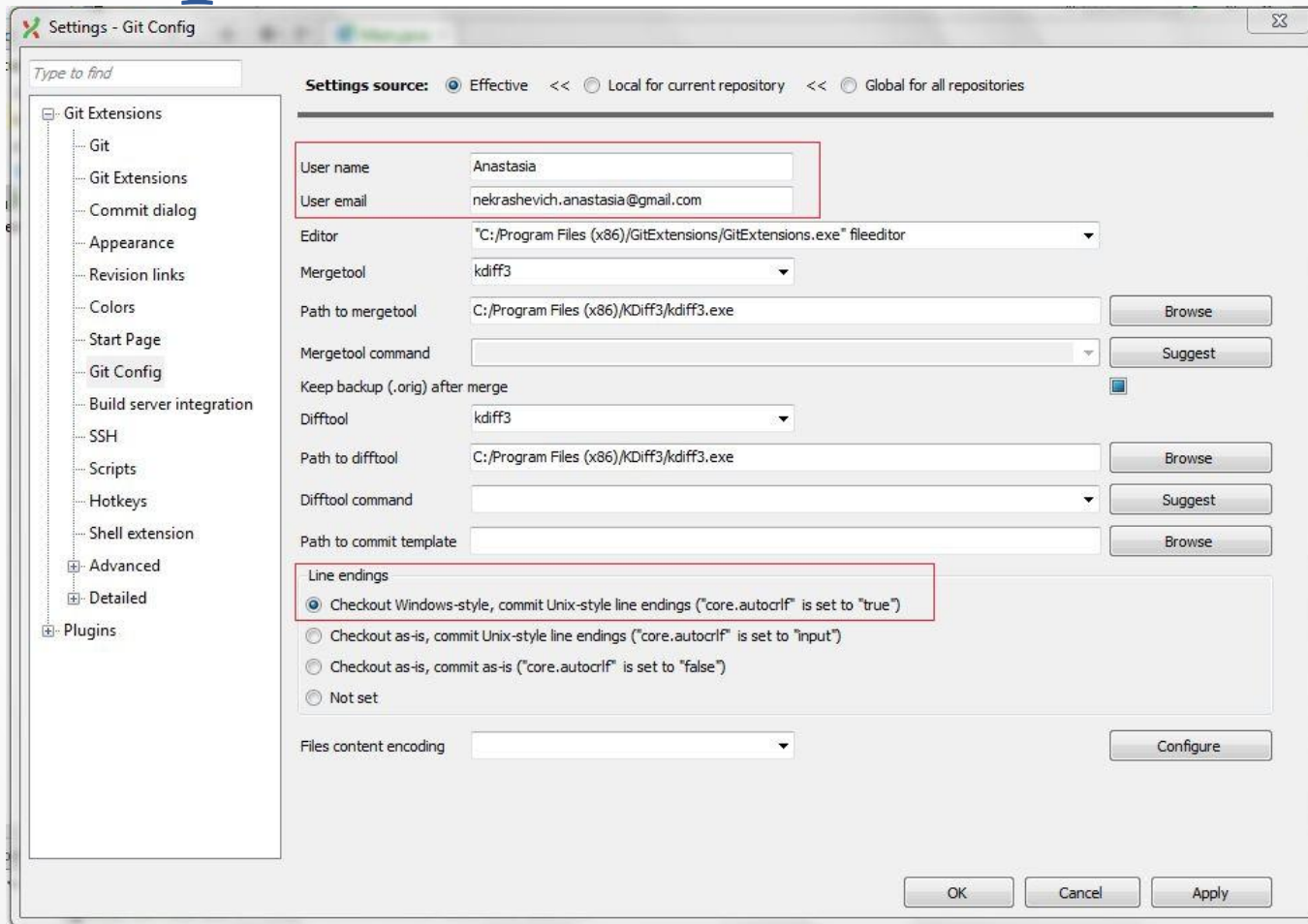
Открываем Git Extensions



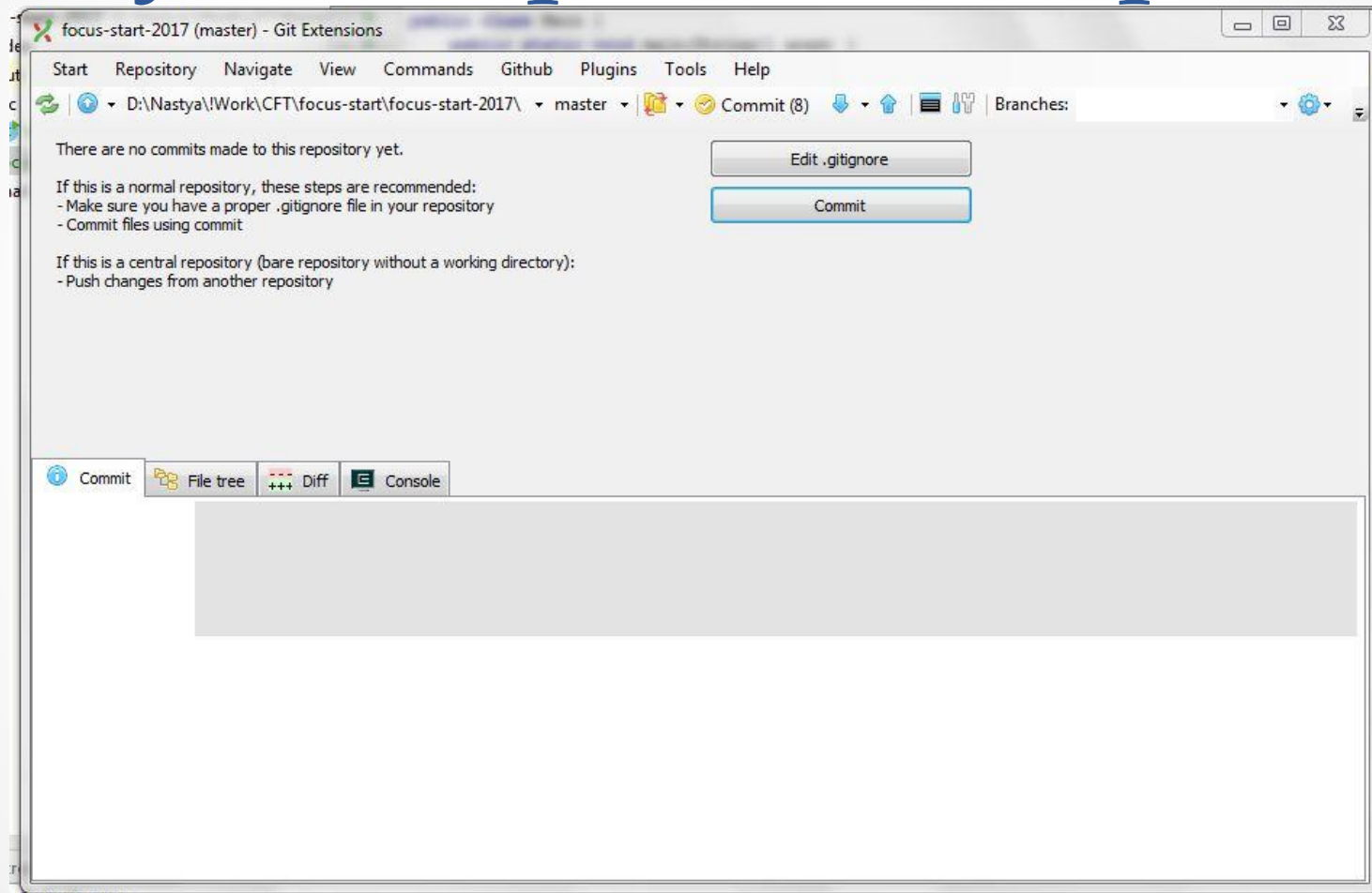
Клонирование репозитория



Настройки GitExtensions



Пустой репозиторий



Создаём Hello world
проект в Idea

...

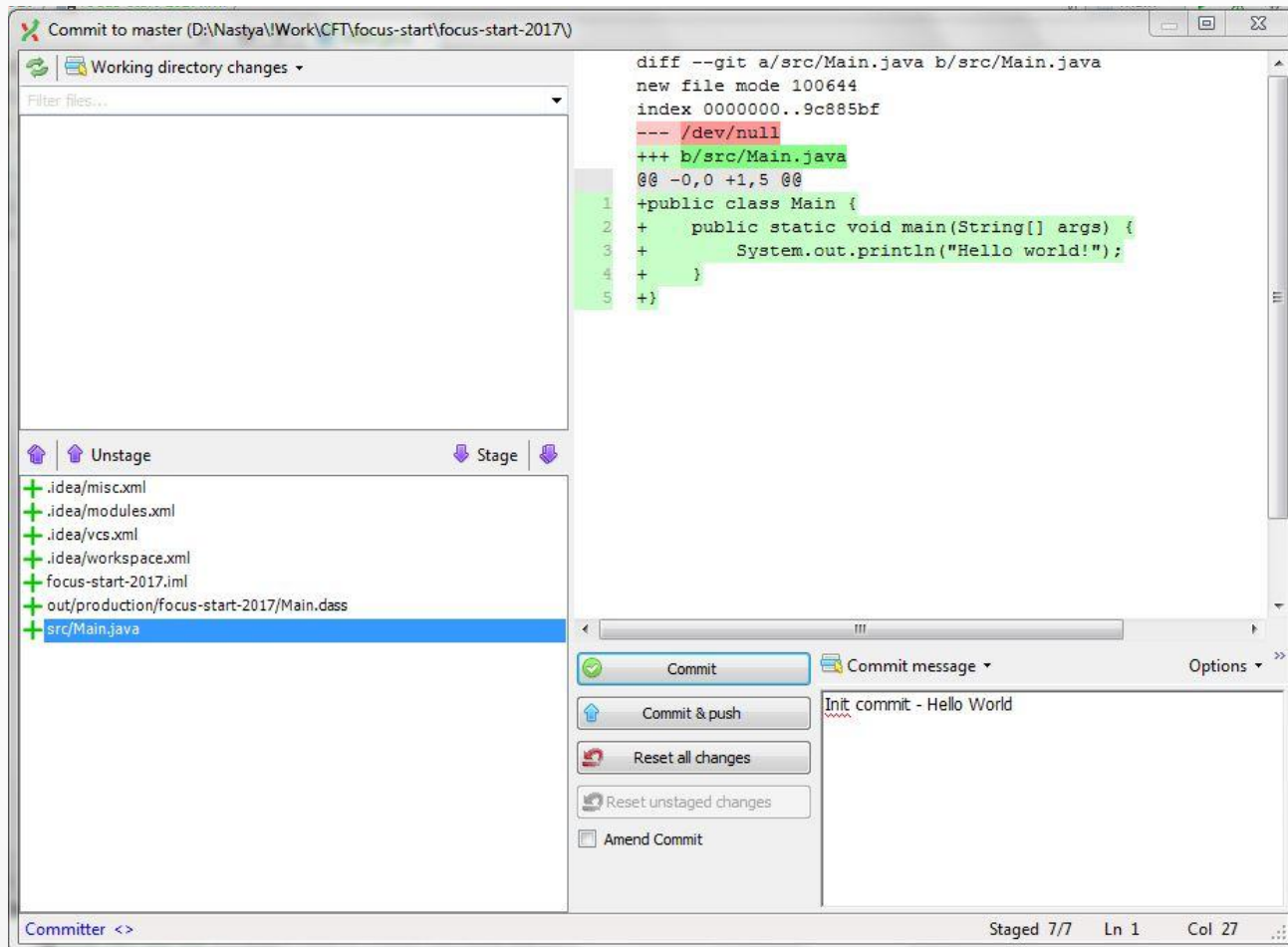
Stage изменений

The screenshot shows the 'Commit to master' dialog in IntelliJ IDEA. The window title is 'Commit to master (D:\Nastya\Work\CFT\focus-start\focus-start-2017)'. The 'Working directory changes' section on the left lists files to be committed: `.idea/workspace.xml`, `out/production/focus-start-2017/Main.dass`, and `src/Main.java`. Below this, the 'Unstage' section lists files that are not staged: `.idea/misc.xml`, `.idea/modules.xml`, `.idea/vcs.xml`, `focus-start-2017.iml`, and `src/Main.java`. A 'Stage' button is visible between the two sections. The main area displays an XML diff for `src/Main.java`, showing changes to the `ChangeListenerManager` component. The diff includes a list of new changes and options like `EXCLUDED_CONVERTED_TO_IGNORED`, `TRACKING_ENABLED`, `SHOW_DIALOG`, `HIGHLIGHT_CONFLICTS`, `HIGHLIGHT_NON_ACTIVE_CHANGE_LIST`, and `LAST_RESOLUTION`. The `FileEditorManager` component is also shown, with a new entry for `Main.java`. At the bottom, there are buttons for 'Commit', 'Commit & push', 'Reset all changes', 'Reset unstaged changes', and 'Amend Commit'. A 'Commit message' field is present with the placeholder text 'Enter commit message'. The status bar at the bottom indicates 'Committer <>' and 'Staged 5/8 Ln 0 Col 0'.

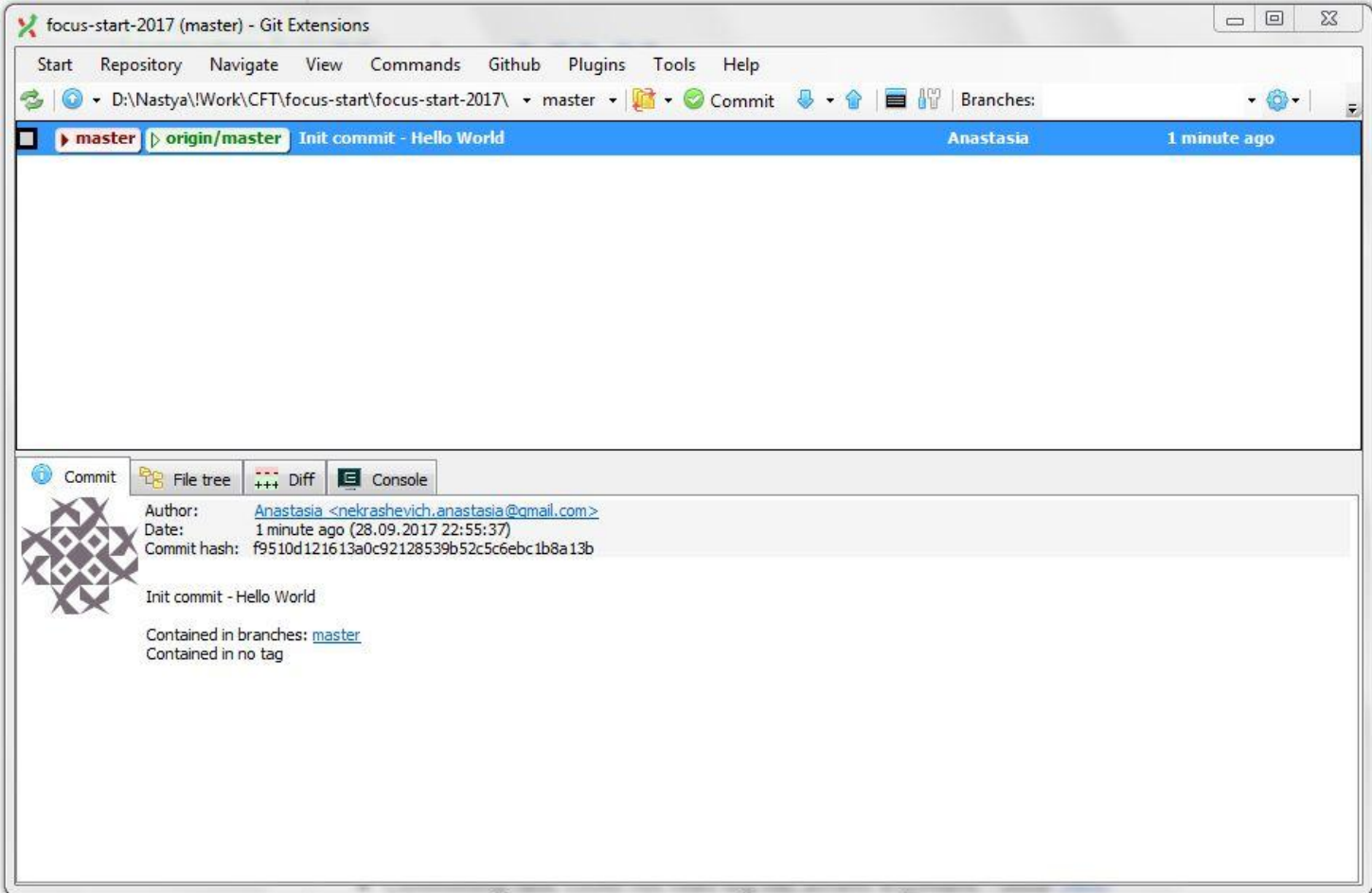
```
<?xml version="1.0" encoding="UTF-8"?>
<project version="4">
  <component name="ChangeListenerManager">
    <list default="true" id="a463172a-3943-4eca-94e9-f99d:
      <change type="NEW" beforePath="" afterPath="$PROJEC:
      <change type="NEW" beforePath="" afterPath="$PROJEC:
      <change type="NEW" beforePath="" afterPath="$PROJEC:
      <change type="NEW" beforePath="" afterPath="$PROJEC:
    </list>
    <ignored path="$PROJECT_DIR$/out/" />
    <option name="EXCLUDED_CONVERTED_TO_IGNORED" value="t:
    <option name="TRACKING_ENABLED" value="true" />
    <option name="SHOW_DIALOG" value="false" />
    <option name="HIGHLIGHT_CONFLICTS" value="true" />
    <option name="HIGHLIGHT_NON_ACTIVE_CHANGE_LIST" value="
    <option name="LAST_RESOLUTION" value="IGNORE" />
  </component>
  <component name="FileEditorManager">
    <leaf>
      <file leaf-file-name="Main.java" pinned="false" cur:
        <entry file="file://$PROJECT_DIR$/src/Main.java">
          <provider selected="true" editor-type-id="text-
          <state relative-caret-position="34">
```

Commit & Push

ИЗМЕНЕНИЙ



История изменений



The screenshot displays the Git GUI interface for a repository named 'focus-start-2017'. The main window shows a commit history table with the following details:

Branch	Commit Message	Author	Time
master	Init commit - Hello World	Anastasia	1 minute ago

Below the table, the 'Commit' tab is active, showing the following commit details:

- Author:** Anastasia <nekrashevich.anastasia@gmail.com>
- Date:** 1 minute ago (28.09.2017 22:55:37)
- Commit hash:** f9510d121613a0c92128539b52c5c6ebc1b8a13b
- Message:** Init commit - Hello World
- Contained in branches:** [master](#)
- Contained in no tag:**

Создание ветки

The screenshot shows the GitHub interface for the repository 'focus-start-2017'. The left sidebar contains navigation links: Overview, Source, Commits, Branches (highlighted with a red box), Pull requests, and Pipelines. The main content area is titled 'Nekrashevich Anastasia / focus-start-2017' and 'Branches'. It features a 'Create branch' button (highlighted with a red box), a search bar 'Find branches', and filter buttons for 'Active' and 'Merged'. A table lists the 'master' branch as the 'MAIN BRANCH', updated '4 days ago'. Below the table, it states 'No matching branches'.

java focus-start-2017

Overview

Source

Commits

Branches

Pull requests

Pipelines

Nekrashevich Anastasia / focus-start-2017

Create branch

FILTERS: Active Merged

Find branches

Branch	Behind	Ahead	Updated	Pull request	Builds
master			4 days ago		


No matching branches

Создание ветки

Create branch

Branch from

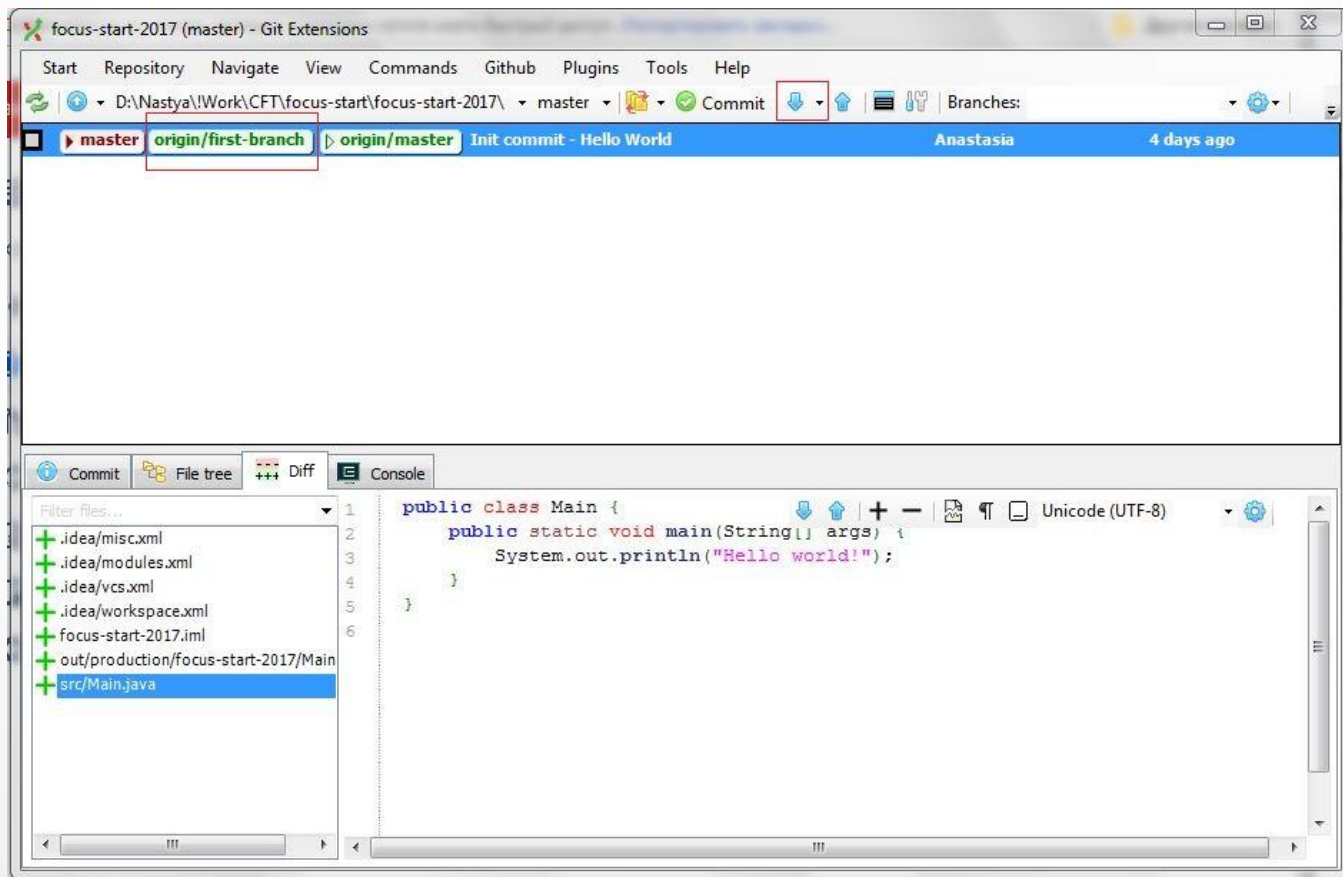
Branch name*



The diagram illustrates the branch structure. It shows a horizontal line representing the 'master' branch, starting from a grey dot on the left and ending at a grey dot on the right. From the right end of this line, a blue curved line branches off downwards and to the right, ending at a blue circle representing the 'first-branch' branch. To the right of the diagram are two buttons: the top one is labeled 'v master' and the bottom one is labeled 'v first-branch', both with a small 'v' icon.

Переключение на ветку

Обновляем проект (pull или fetch)
Переключаемся на ветку - checkout



Делаем изменения в Hello world проекте в Idea

...

Commit изменений

The screenshot displays the IntelliJ IDEA interface for a Git commit. The top toolbar includes 'Start', 'Repository', 'Navigate', 'View', 'Commands', 'Github', 'Plugins', 'Tools', and 'Help'. The current repository path is 'D:\Nastya\!Work\CFT\focus-start\focus-start-2017\'. The selected branch is 'first-branch', and the commit message is 'Init commit - Hello World'. The commit history shows two entries: 'first-branch' (4 seconds ago) and 'master' (4 days ago).

The bottom panel shows a diff view for the file 'src/Main.java'. The diff highlights a modification in the 'src/Main.java' file, showing the following changes:

```
diff --git a/.idea/workspace.xml b/.idea/workspace.xml
index b3003d7..d5cbb98 100644
--- a/.idea/workspace.xml
+++ b/.idea/workspace.xml
@@ -2,11 +2,7 @@
 <project version="4">
   <component name="ChangeListManager">
     <list default="true" id="a463172a-3943-4eca-94e9-f99d37b817f1" name="...
-   <change type="NEW" beforePath="" afterPath="$PROJECT_DIR$/.idea/mis
-   <change type="NEW" beforePath="" afterPath="$PROJECT_DIR$/.idea/mod
-   <change type="NEW" beforePath="" afterPath="$PROJECT_DIR$/.idea/vcs
-   <change type="NEW" beforePath="" afterPath="$PROJECT_DIR$/focus-sta
-   <change type="NEW" beforePath="" afterPath="$PROJECT_DIR$/src/Main.
+   <change type="MODIFICATION" beforePath="$PROJECT_DIR$/src/Main.java
```

Push изменений

The screenshot shows the IntelliJ IDEA Git Extensions interface. The top bar indicates the current repository is 'focus-start-2017 (first-branch)'. The main area displays a list of commits:

Branch	Commit Message	Author	Time
first-branch	commit into branch	Anastasia	1 minute ago
origin/first-branch		Anastasia	1 minute ago
master	origin/master	Anastasia	4 days ago
	Init commit - Hello World	Anastasia	4 days ago

The bottom panel shows a diff view for the file 'src/Main.java'. The diff highlights the changes between the parent branch (a/src/Main.java) and the current branch (b/src/Main.java):

```
diff --git a/src/Main.java b/src/Main.java
index 9c885bf..6cdd004 100644
--- a/src/Main.java
+++ b/src/Main.java
@@ -1,5 +1,6 @@
1 1 public class Main {
2 2     public static void main(String[] args) {
3 3         System.out.println("Hello world!");
4 4 +     System.out.println("Double Hello world!");
4 5     }
5 6 }
```

Ветки в Bitbucket

The screenshot shows the Bitbucket interface for a repository named 'focus-start-2017'. The left sidebar contains navigation links: Overview, Source, Commits, Branches (highlighted), Pull requests, Pipelines, and Downloads. The main content area shows the 'Branches' section for user 'Nekrashevich Anastasia'. It includes a 'Create branch' button, filter buttons for 'Active' and 'Merged', and a search box 'Find branches'. A table lists the branches:

Branch	Behind	Ahead	Updated	Pull request	Builds
master MAIN BRANCH			4 days ago		
first-branch		1	2 minutes ago		

The 'first-branch' row is highlighted with a red box around the 'Ahead' column value '1', indicating it is one commit ahead of the 'master' branch.

Pull request ветки

The screenshot shows the GitHub interface for a repository named 'focus-start-2017' by user 'Nekrashevich Anastasia'. On the left, a sidebar contains navigation links: Overview, Source, Commits, Branches, Pull requests (highlighted with a red box), and Pipelines. The main content area is titled 'Pull requests' and includes a 'Create pull request' button (highlighted with a red box) in the top right. Below the title, there are filter options: 'FILTER BY: Open', 'Author', 'Target branch', and 'I'm reviewing', each with a dropdown arrow. A search icon is also present. The main area displays the message 'No open pull requests' with a sub-message: 'When a branch is ready to review, create a pull request.'

java focus-start-2017

Nekrashevich Anastasia / focus-start-2017

Create pull request

Pull requests

FILTER BY: Open Author Target branch I'm reviewing

No open pull requests

When a branch is ready to review, create a pull request.

Pull request ветки

Nekrashevich Anastasia / focus-start-2017

Pull requests

Create a pull request

Java nastyaka / focus-start-2017
Created 4 days ago, updated 2 minutes ago

first-branch

Java nastyaka/focus-start-2017

master

Title* commit into branch

Description

Start typing to search for a user

Close branch Close **first-branch** after the pull request is merged

Create pull request

Diff Commits

Author	Commit	Message	Date	Builds
 Nekrashevich A...	3f4d842	commit into branch	4 minutes ago	

Настройка доступа

java focus-start-2017

- Overview
- Source
- Commits
- Branches
- Pull requests
- Pipelines
- Downloads
- Boards
- Settings**

Nekrashevich Anastasia / focus-start-2017

Settings

GENERAL

- Repository details
- User and group access**
- Access keys
- Username aliases

WORKFLOW

- Branch permissions
- Default reviewers
- Webhooks
- Links


FEATURES

- Git LFS
- Wiki
- Issue tracker

User and group access

Here's where you grant users and groups access to this repository. For a list of all users with access to any of your private repositories, see which users count towards your bill on the Users on plan page. [Learn more](#)

Users

Username or email address	Read		Add
 Nekrashevich Anastasia	owner		

Groups

Select a group	Read		Add
----------------	------	--	-----

Настройка доступа

Необходимо добавить права на чтение следующим пользователям:

- Sergey Krupin (cft_krupin)
- Kristina Kolesnikova (k_kolesnikova)
- Nekrashevich Anastasia (nekrashevich)



The screenshot shows a 'Users' management interface. At the top, there is a search bar labeled 'Username or email address', a 'Read' button, a dropdown arrow, and an 'Add' button. Below this, a list of users is displayed with their names and associated permissions. The first user, 'Nekrashevich Anastasia', is listed as the 'owner'. The other two users, 'Kristina Kolesnikova' and 'Sergey Krupin', have 'READ', 'WRITE', and 'ADMIN' permissions. Each user entry has a small 'x' icon in a circle to its right, likely for deleting the user.

Username or email address	Permissions	Action
Nekrashevich Anastasia	owner	
Kristina Kolesnikova	READ WRITE ADMIN	✕
Sergey Krupin	READ WRITE ADMIN	✕

Договоренности по работе с ветками

- каждая задача делается в новой ветке,
- когда задача готова, создается пул-реквест в мастер
- после проверки, сдачи и апрува задачи ветка сливается в мастер
- каждая задача – отдельный модуль идеи