Министерство образования Новосибирской области

ГБПОУ НСО «Новосибирский авиационный технический колледж имени Б.С. Галущака»

# СРС 1 Публикация проекта в Git

Методическое пособие

# Учебная дисциплина: МДК.01.01 Разработка программных модулей

Специальность: 09.02.07 Информационные системы и программирование

Разработал:

Плешков А. А.

2022

**Содержание**

[1 Установка GIT Bash 3](#_Toc116848577)

[2 Настройка GIT 4](#_Toc116848578)

[3 Работа с репозиторием 5](#_Toc116848579)

[3.1 Создание нового репозитория 5](#_Toc116848580)

[3.2 Определение состояния репозитория 5](#_Toc116848581)

[4 Подготовка файлов 6](#_Toc116848582)

[4.1 Добавление файлов 6](#_Toc116848583)

[4.2 Создание коммитов 6](#_Toc116848584)

[5 Отправка файлов на сервер 7](#_Toc116848585)

# **1 Установка GIT Bash**

Для установки GIT Bash, следует перейти на электронный ресурс gitforwindows и на главной странице нажать кнопку «Download».

Главная страница сайта изображена на рисунке 1



Рисунок 1 — Главная страница сайта

После нажатия кнопки, скачается установщик Git. Необходимо запустить установщик и нажимать кнопку «Далее» до конца, пока не начнется установка.

# **2 Настройка GIT**

После установки GIT, необходимо добавить пару настроек. А именно добавим наше имя пользователя и адрес электронной почты. Откроем консоль и введем команды:

* git config --global user.name "My Name"
* git config --global user.email "myEmail@example.com"

Теперь каждое наше действие будет отмечено именем и почтой. Таким образом, пользователи всегда будут в курсе, кто отвечает за какие изменения — это вносит порядок.

# **3 Работа с репозиторием**

# **3.1 Создание нового репозитория**

Чтобы создать новый репозиторий, нам нужно открыть терминал, зайти в папку нашего проекта и выполнить команду init. Это включит приложение в этой конкретной папке и создаст скрытую директорию .git, где будет храниться история репозитория и настройки.

Для этого в окне терминала введите команду:

* git init

# **3.2 Определение состояния репозитория**

status — это еще одна важнейшая команда, которая показывает информацию о текущем состоянии репозитория: актуальна ли информация на нём, нет ли чего-то нового, что поменялось, и так далее.

Для этого в окне терминала введите команду:

* git status

# **4 Подготовка файлов**

# **4.1 Добавление файлов**

В git есть концепция области подготовленных файлов. Можно представить ее как холст, на который наносят изменения, которые нужны в коммите. Сперва он пустой, но затем мы добавляем на него файлы (или части файлов, или даже одиночные строчки) командой add и, наконец, коммитим все нужное в репозиторий (создаем слепок нужного нам состояния) командой commit.  
Добавить файл можно командами:

* git add "Имя файла"
* git add –A
* git add .

# **4.2 Создание коммитов**

Для сохранения изменений, их необходимо закоммитить с помощью команды:

* git commit -m 'Add some code'

Флажок -m задаст commit message - комментарий разработчика. Он необходим для описания закоммиченных изменений. И здесь работает золотое правило всех комментариев в коде: «Максимально ясно, просто и содержательно обозначь написанное!»

Для просмотра все выполненных фиксаций можно воспользоваться историей коммитов. Она содержит сведения о каждом проведенном коммите проекта. Запросить ее можно при помощи команды:

* git log

# **5 Отправка файлов на сервер**

Можно переслать наш локальный коммит на сервер Этот процесс происходит каждый раз, когда мы хотим обновить данные в удаленном репозитории.

Команда, предназначенная для этого:

* git push origin master

Команда принимает два параметра: имя удаленного репозитория (например origin) и ветку, в которую необходимо внести изменения (master — это ветка по умолчанию для всех репозиториев).

**Список источников**

gitforwindows [Электронный ресурс].— Загрузка Git Bash — URL: https://gitforwindows.org/ (дата обращения: 16.10.2022)