

Специализированные инструменты	Обзор	Особенности	Достоинства	Недостатки
GitHub Desktop	GitHub Desktop это визуальный инструмент, позволяющий вам управлять Local Repository (Локальный репозиторий) на вашем компьютере.	Быстро добавляйте соавторов к вашему коммиту. Отлично подходит для сопряжения. Новый GitHub Desktop поддерживает подсветку синтаксиса при просмотре различий для различных языков.	Наследует Git Flow, позволяет в 2 клика сделать Pull request, что очень удобно если вы активный участник open source комьюнити. А также удобный и приятный интерфейс не может не радовать.	—
GitKraken	GitKraken — это кроссплатформенный, удобный и высокоэффективный GUI git клиент на Linux, Windows и macOS. Он поддерживает GitHub, Bitbucket и Gitlab, и конечно же позволяет работать с корпоративными репозиториями с исходным кодом.	<ol style="list-style-type: none"> <li>1. Визуальное взаимодействие и подсказки;</li> <li>2. Поддержка нескольких профилей;</li> <li>3. Поддерживает кнопки отмены и повтора функции;</li> <li>4. Быстрый и интуитивно понятный интерфейс поиска;</li> <li>5. Легко адаптируется к рабочей области пользователя, а также поддерживает подмодули и Git-flow;</li> <li>6. Интегрируется с аккаунтами на GitHub или BitBucket;</li> </ol>	Работа с репозиториями с использованием GitKraken действительно упрощается, и речь идет не только о начинающих разработчиках. Он позволяет быстро выполнять как базовые операции, так что-то более сложное: решать конфликты слияния, объединять, переставлять коммиты, при необходимости переписывать историю.	Без платной версии урезанная функциональность.

		7. Горячие клавиши и многое другое.	Конечно, все это можно выполнять в консоли, но не всегда удастся вспомнить правильный синтаксис команды и приходится тратить время на поиск решения. В случае сложного ветвления, в любом случае, приходится использовать какое-либо графическое представление дерева.	
Tower	Tower - это Git-клиент для macOS, который делает Git простым и эффективным. Пользователи пользуются элегантным и всесторонним интерфейсом и набором функций, который позволяет им наслаждаться полной мощностью Git.	<p>Tower содержит всё самое важное, что касается выбранного вами репозитория:</p> <p>в Working Copy вы постоянно будете заглядывать, чтобы посмотреть, какие файлы изменились, и закоммитить либо откатить эти изменения</p> <p>в Stashes будут сохраняться все ваши стешы</p> <p>в Branches будут отображены все ветки репозитория, для которых вы делали checkout. Невозможно не увидеть HEAD-ветку; кроме того, рядом с каждой веткой наглядно отображается различие по коммитам – сколько нужно</p>	<p>Tower абстрагирует сложность Git в удобном интерфейсе. Например, множество действий можно выполнить просто с помощью перетаскивания, а ошибки можно легко отменить.</p> <p>Продвинутые пользователи могут повысить свою производительность с помощью таких функций, как однострочное размещение,</p>	Платный

		<p>слить и сколько залить. Кстати, ветки можно мёржить прямо в этом списке, простым перетаскиванием.</p> <p>в Tags будут собраны метки, которыми знаменуются важные моменты в ходе работы над проектом</p> <p>в Remotes можно будет посмотреть, что творится в ветках вашего репозитория на удалённом сервере</p> <p>наконец, Submodules пригодится вам, если вы используете дочерние проекты Git</p>	поддержка подмодулей или история файлов.	
Sourcetree	<p>Sourcetree это продукт для программирования и разработки программного обеспечения на технологиях Git и Mercurial, поддерживающий режим визуализации кода репозитория и управление ими. Sourcetree подходит для совместной работы и отслеживания результатов, а также контроля работы</p>	<p>Он имеет визуальный интерфейс и прост в использовании. Это также инструмент для систем контроля версий Mercurial и Subversion. Поддержка создания, отправки, клонирования, push, pull, объединения и других операций.</p>	<p>Даже после того, как код в настоящее время написан и отправлен на локальный склад, вам не нужно бояться его потерять, и вы также можете записать версию. Если вы хотите получить код несколько дней назад в будущем, это также возможно.</p>	<p>Слияние кода заключается в удалении вновь написанного кода из удаленного хранилища, а затем объединении в локальном хранилище (и в случае возникновения конфликта, который должен разрешить программист).</p>

	команды в реальном времени. Имеет возможность интеграции в Bitbucket, Jira, Confluence.			
SmartGit	<p>SmartGit является интерфейсом для распределенной системы управления версиями Git и работает в Windows, Mac OS X и Linux.</p> <p>SmartGit предназначен для разработчиков, которые предпочитают графический пользовательский интерфейс над клиентом командной строки, чтобы быть еще более продуктивным с Git - самым мощным DVCS сегодня.</p>	<p>Нет необходимости устанавливать и настраивать дополнительные инструменты. SmartGit включает в себя все, что вам нужно:</p> <p>интуитивно понятное сравнение файлов (без простой оболочки вокруг git diff )</p> <p>мощное слияние файлов (свободно редактируемый «Решатель конфликтов»)</p> <p>Git-Flow (обработка веток высокого уровня)</p> <p>графическая история слияния и фиксации</p> <p>встроенный Git-клиент</p> <p>SSH-клиент</p>	<p>Идеальный клиент Git должен поддерживать ваши рабочие процессы. Вы можете настроить SmartGit различными способами:</p> <p>Настройки для слияния, перебазирувания</p> <p>Внешние инструменты, Внешние или встроенные инструменты сравнения или разрешения конфликтов, синтаксическая окраска, Горячие клавиши, Расположение определенных видов, Панели инструментов,</p>	—

			Светлые и темные темы	
TortoiseGit	<p>TortoiseGit - это интерфейс оболочки Windows для Git и основанный на TortoiseSVN. Это открытый исходный код и может быть полностью построен с бесплатным программным обеспечением.</p>	<p>Поскольку это не интеграция для конкретной среды разработки, такой как Visual Studio, Eclipse или других, вы можете использовать ее с любыми инструментами разработки, которые вам нравятся, и с любым типом файла. Основное взаимодействие с TortoiseGit будет использовать контекстное меню проводника Windows. TortoiseGit поддерживает вас с помощью обычных задач, таких как фиксация, отображение журналов, различение двух версий, создание ветвей и тегов, создание патчей и т. Д. (См. Наши скриншоты или документацию). Он разработан под лицензией GPL. Это означает, что он абсолютно свободен для любого пользователя, в том числе в коммерческой среде, без каких-либо ограничений. Исходный код также свободно доступен, поэтому вы можете даже разработать свою собственную версию, если хотите.</p>	—	<p>Люди, которые пользовались им, дают много критики по поводу этого инструмента.</p>

Git Extensions	Git Extensions - визуальный клиент системы управления версиями Git, позволяющий использовать Git без использования консольного интерфейса.	Активное сообщество Git Extensions продолжает расти и поддерживает Git Extensions уже более двух лет. Изучайте историю отдельных файлов. Переименованные и перемещенные файлы сопоставляются и отображаются в одной истории. Вам не нужно отмечать файлы как переименованные/перемещенные. Система автоматически определяет переименованные файлы.	Интеграция Windows Explorer в Git Плагин Visual Studio (2008/2010/2012/2013) для Git Богатый пользовательский интерфейс для Git Один инсталлятор устанавливает сразу MSysGit, GitExtensions и слияние KDiff3 Поддержка 32bit и 64bit	—
----------------	--	---	--	---

Используемые ресурсы:

<https://blog.desdelinux.net/ru/gitkraken-un-excelente-cliente-de-git-multiplataforma-para-tu-escritorio/>

<https://appstorrent.ru/322-tower.html>

<https://appstudio.org/reviews/tower.html>

<https://codex.so/github-start>

<https://russianblogs.com/article/7139301903/>

<https://www.syntevo.com/smartgit/>

<https://www.it-roy-ru.com/ru/git/tortoisegit-protiv-git-extensions/972763513/>

[https://ru.bmstu.wiki/Git\\_Extensions](https://ru.bmstu.wiki/Git_Extensions)

<http://lostapp.ru/soft/gitextensions>

