* 1. Система контроля версий Git

Система контроля версий - это система, записывающая изменения в файл или набор файлов в течение времени и позволяющая вернуться позже к определённой версии.

Git - это быстрая, масштабируемая, распределенная система контроля версий с необычайно богатым набором команд, которая обеспечивает как высокоуровневые операции, так и полный доступ к внутренним компонентам. С Git почти все операции выполняются локально, что дает ему огромное преимущество в скорости в централизованных системах, которым постоянно приходится взаимодействовать с сервером где-либо.

Базовый подход в работе с Git выглядит так:

* Вы изменяете файлы в вашей рабочей директории.
* Вы выборочно добавляете в индекс только те изменения, которые должны попасть в следующий коммит, добавляя тем самым снимки только этих изменений в область подготовленных файлов.
* Когда вы делаете коммит, используются файлы из индекса как есть, и этот снимок сохраняется в вашу Git-директорию.

Особенность Git, которая действительно выделяет его почти из всех остальных SCM, - это модель ветвления. Git позволяет и поощряет вас иметь несколько локальных веток, которые могут быть полностью независимы друг от друга. Создание, объединение и удаление этих линий разработки занимает секунды.

Это означает, что вы можете делать такие вещи, как:

* Переключение контекста без трения. Создайте ветку, чтобы опробовать идею, сделайте коммит несколько раз, вернитесь туда, откуда вы вышли, примените патч, вернитесь туда, где вы экспериментируете, и объедините его.
* Ролевые кодовые линии. Иметь ветку, которая всегда содержит только то, что идет в производство, другую, в которую вы объединяете работу для тестирования, и несколько более мелких для повседневной работы.
* Функциональный рабочий процесс. Создайте новые ветви для каждой новой функции, над которой вы работаете, чтобы вы могли плавно переключаться между ними, а затем удаляете каждую ветку, когда эта функция объединяется с вашей основной линией.
* Одноразовые эксперименты. Создайте ветку для экспериментов, осознайте, что она не сработает, и просто удалите ее, оставив работу, и никто больше ее не увидит (даже если вы в это время нажали другие ветви).

У Git есть три основных состояния, в которых могут находиться ваши файлы: зафиксированное (committed), изменённое (modified) и подготовленное (staged).

* Зафиксированный значит, что файл уже сохранён в вашей локальной базе.
* К изменённым относятся файлы, которые поменялись, но ещё не были зафиксированы.
* Подготовленные файлы — это изменённые файлы, отмеченные для включения в следующий коммит.
  1. Работа с системой контроля версий в среде разработки PHPStorm

После того, как git установлен в вашей операционной системе, нужно подключить его к PhpStorm. Сделать это можно в настройках IDE по адресу File | Settings | Version Control | Git. Тут есть опция Path to Git executable, в которой нужно задать путь до исполняемого файла git. Если путь не определился автоматически, то необходимо самостоятельно указать путь до расположения git.exe. После указания нужно нажать на кнопочку Test, чтобы убедиться, что все работает.

Для того чтобы подключить свой репозиторий следует сделать следующее: перейти в VCS | Get from version control и в открывшемся диалоговом окне ввести URL на ваш репозиторий. После чего нажать clone.

Внеся изменения в проект, для его коммита следует нажать сочетание клавиш cntrl+k, и в диалоговом окне написать комментарий при необходимости и затем нажать «commit». Для публикации измененных файлов на гитхабе следует нажать сочетание клавиш cntrl+shift+k и в диалоговом окне нажать «push».

* 1. Веб-сервис гитхаб

GitHub - крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки.

Создатели сайта называют GitHub «социальной сетью для разработчиков». Кроме размещения кода, участники могут общаться, комментировать правки друг друга, а также следить за новостями знакомых. С помощью широких возможностей Git программисты могут объединять свои репозитории — GitHub предлагает удобный интерфейс для этого и может отображать вклад каждого участника в виде дерева. Для проектов есть личные страницы, небольшие Вики и система отслеживания ошибок.

Прямо на сайте можно просмотреть файлы проектов с подсветкой синтаксиса для большинства языков программирования.

* Можно создавать приватные репозитории, которые будут видны только вам и выбранным вами людям;
* Раньше возможность создавать приватные репозитории была платной;
* Есть возможность прямого добавления новых файлов в свой репозиторий через веб-интерфейс сервиса;
* Код проектов можно не только скопировать через Git, но и скачать в виде обычных архивов с сайта;
* Кроме Git, сервис поддерживает получение и редактирование кода через SVN и Mercurial;
* На сайте есть pastebin-сервис gist.github.com для быстрой публикации фрагментов кода.