

СИСТЕМА КОНТРОЛЯ ВЕРСИЙ GIT И ВЕБ-СЕРВИС GITHUB

Большинство пользователей прибегают к практике копирования файлов в отдельные директории (папки) в качестве механизма управления версиями. Этот подход является довольно простым, однако он подвержен риску ошибок из-за возможности забыть текущее местоположение файла и случайно модифицировать неправильный файл или скопировать неверные файлы. Для преодоления данной проблемы программисты разработали системы контроля версий.

Система контроля версий (Version Control System) – система, которая отслеживает изменения, произведенные в файле или наборе файлов в течение времени, обеспечивая возможность возврата к определенной версии в будущем.

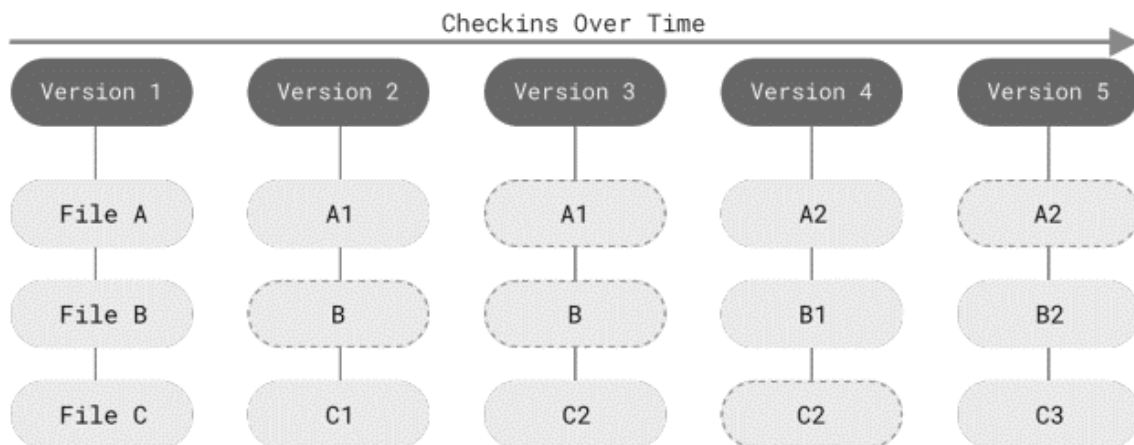
GitHub – это веб-платформа для хостинга проектов, использующих систему контроля версий Git.

Git – это система контроля версий, работающая локально на компьютере разработчика. Она позволяет отслеживать изменения и управлять версиями проекта.

Git предоставляет набор команд для работы с репозиторием, таких как `git init` для инициализации нового репозитория, `git add` для добавления файлов в индекс, `git commit` для фиксации изменений и создания новой версии, `git push` для отправки изменений на удаленный сервер и многие другие.

Подход Git к хранению данных похож на набор снимков миниатюрной файловой системы. Каждый раз, когда вы делаете коммит (сохранение изменений) своего проекта в Git, система запоминает, как выглядит каждый файл в этот момент, и сохраняет ссылку на этот снимок. Если файлы не были изменены, Git не запоминает эти файлы вновь, а создает ссылку на предыдущую версию идентичного файла, который уже сохранен. Git представляет свои данные как поток снимков.

Хранение данных как снимков проекта во времени



Примечание – Источник [1].

Git используется для локального отслеживания и управления версиями, в то время как GitHub используется для хранения удаленных репозиториев, совместной работы и обмена кодом между разработчиками.

GitHub обеспечивает дополнительные функции, такие как возможность разработчикам хранить, совместно работать над проектами, отслеживать изменения, вносить вклад в открытые проекты, управление задачами, отслеживание ошибок, вики и документацию, в то время как Git предоставляет только основные возможности контроля версий.

Репозитории на GitHub могут быть публичными, частными или внутренними (доступными только для членов организации). Пользователи могут создавать ветки (branches) для экспе-

риментов, разработки новых функций или исправления ошибок, а затем создавать pull request's (запросы на внесение изменений), чтобы интегрировать свои изменения в основную ветку проекта.

Принцип работы GitHub основан на централизованном хранении и совместной работе над проектами с использованием системы контроля версий Git. Основные принципы работы GitHub:

- Хранение кода и файлов: GitHub предоставляет облачное хранилище (репозитории) для хранения кода и других файлов. Репозиторий является местом, где разработчики могут сохранять свой код и отслеживать его изменения.

- Управление версиями: Git и GitHub позволяет отслеживать изменения в коде, создавать версии проекта и возвращаться к предыдущим состояниям кода при необходимости. Это обеспечивает безопасность и контроль над развитием проекта.

- Ветвление (Branching) и слияние (Merging): GitHub позволяет создавать отдельные ветки (branches) кода для разработки новых функций или исправления ошибок, не затрагивая основной код проекта. После завершения работы над веткой изменения могут быть интегрированы обратно в основную ветку с помощью процесса слияния (merging).

- Pull Requests: позволяет разработчикам предложить изменения в коде проекта. Другие разработчики могут просмотреть эти изменения, оставить комментарии, провести обсуждение и, если все в порядке, внести изменения в основной код проекта.

- Управление задачами и ошибками: GitHub предоставляет инструменты для управления задачами и отслеживания ошибок в проекте. Разработчики могут создавать задачи, присваивать им ответственных, отслеживать их выполнение и связывать с ними соответствующие коммиты кода.

- Документация и обсуждение: GitHub также предоставляет возможности для создания и ведения документации проекта, а также обсуждения кода и других важных аспектов проекта через комментарии и обсуждения.

В целом, GitHub обеспечивает среду для совместной работы над проектами, эффективно-го управления версиями кода, обмена идеями и обратной связью между разработчиками.

Список использованной литературы

1. **Установка Git** [Электронный ресурс] // Git. – Режим доступа: <https://git-scm.com/book/ru/v2/>. – Дата доступа: 29.02.2023.