

Практическая работа № 4

Тема: Использование систем контроля версий исходного кода программного продукта.

Цель: изучить на практике понятия и компоненты систем контроля версий (СКВ), приемы работы с ними; получение первоначальных навыков использования СКВ, получение первоначальных навыков организации коллективной разработки программного продукта. Освоить специализированное ПО и распространенный сервис для работы с распределенной СКВ Git – TortoiseGit и GitHub.

Задание

1. Изучить литературу и теоретические сведения.
2. Отработать навыки использования хранилища на локальной машине с использованием одной из СКВ (операции по настройке хранилища, загрузке и выгрузке кода).
3. Освоить передачу истории хранилища по сети.
4. Изучить действия, связанные с ветвлениями и разрешением конфликтов (взаимодействие в команде над проектом).
5. Согласовать вариант с преподавателем и выбрать список задач для демонстрации своих навыков работы с СКВ.
6. Составить последовательность команд Git, которые выполняют требуемые задачи (табл. 1).
7. Презентовать результаты операций преподавателю и защитить работу.

Варианты учебных заданий

Вариант	Действия с репозиторием проекта
1	2
1.	<ul style="list-style-type: none"> – создание (инициализация) локального репозитория в директории «project1» (предварительно создайте эту директорию); – сделать локальную копию удаленного репозитория проекта по адресу «https://remote_address/account1/project1»; – добавить в репозиторий файл (проиндексировать его);
2.	<ul style="list-style-type: none"> – добавить в репозиторий файл (проиндексировать его); – сделать коммит этого файла в репозиторий, с комментарием; – добавить связь локального репозитория с удаленным, который располагается по адресу «https://remote_address/account1/project1»;
3.	<ul style="list-style-type: none"> – сделать коммит файла в репозиторий, с комментарием; – добавить связь локального репозитория с удаленным, который располагается по адресу «https://remote_address/account1/project1»; – отправить в «Основную ветку» удаленного проекта все локальные изменения;
4.	<p>{создать репозиторий, создать все ветки согласно workflow (master, develop)}</p> <ul style="list-style-type: none"> – создать локальный репозиторий; – создать ветку {develop}; – добавить ссылку на удаленный репозиторий; – отправить все изменения;
5.	<p>{мерж конфликты}</p> <ul style="list-style-type: none"> – создать 2 ветки feature_merge_conflict{1, 2} из ветки develop; – создать коммиты в ветках (добавить файл с одинаковым полным именем, но абсолютно разным содержанием); – слить (смержить ветки) в develop. Разрешить конфликт в пользу ветки feature_merge_conflict2; – отправить на удаленный сервер все изменения;
6.	<p>{откат, метки}</p> <ul style="list-style-type: none"> – откатить изменения ветки develop до мержа ветки feature_merge_conflict2; – слить (смержить ветку) develop в master; – создать метку v0.1 на ветке master; – отправить все изменения.