МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ БЕЛАРУСЬ  
БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Факультет информационных технологий и робототехники

Кафедра программного обеспечения информационных систем и  
технологий

**Отчёт  
по лабораторной работе №1  
«Знакомство с Git»**  
по дисциплине: «Конструирование программного обеспечения»

Выполнили: ст. гр. 10701121

Васенков А.Д.

Иванов М.Ю.

Мажейко Е.Д.

Принял: Станкевич С.Н.

Минск 2023

**Лабораторная работа №1**

“Знакомство с Git”

Цель: ознакомиться с системой контроля версий Git и создать совместный репозиторий

1. Сведения о Git

Git - система управления версиями с распределённой архитектурой, изначально она была разработана в 2005 году Линусом Торвальдсом — создателем ядра операционной системы Linux. С её помощью можно вести и отслеживать историю изменения файлов.

С помощью Git-a можно откатить свой проект до более старой версии, сравнивать, анализировать или сливать свои изменения в репозиторий. Репозиторием называют хранилище кода и историю его изменений. Git работает локально и репозитории хранятся в определенных папках на жестком диске. Так же ваши репозитории можно хранить и в интернете. Обычно для этого используют три сервиса: GitHub, Bitbucket и GitLab.

Система используется множеством профессиональных разработчиков программного обеспечения. Она превосходно работает под управлением различных операционных систем и может применяться со множеством интегрированных сред разработки. Разработка в Git ориентирована на обеспечение высокой производительности, безопасности и гибкости распределенной системы. Git показывает очень высокую производительность в сравнении со множеством альтернатив. Это возможно благодаря оптимизации процедур фиксации коммитов, создания веток, слияния и сравнения предыдущих версий. Алгоритмы Git разработаны с учетом глубокого знания атрибутов, характерных для реальных деревьев файлов исходного кода, а также типичной динамики их изменений и последовательностей доступа.

Некоторые системы управления версиями руководствуются именами файлов при работе с деревом файлов и ведении истории версий. Вместо обработки названий система Git анализирует содержимое. Это важно, поскольку файлы исходного кода часто переименовывают, разделяют и меняют местами. Объектные файлы репозитория Git формируются с помощью дельта‑кодирования (фиксации отличий содержимого) и компрессии. Кроме того, такие файлы в чистом виде хранят объекты с содержимым каталога и метаданными версий. Вместе с тем распределенная архитектура системы сама по себе обеспечивает существенный прирост производительности.   
Git лёгок в освоении. Каждая точка сохранения проекта носит название коммит (commit). У каждого commit-a есть hash (уникальный id) и комментарий. Из таких commit-ов собирается ветка. Ветка - это история изменений. У каждой ветки есть свое название. Репозиторий может содержать в себе несколько веток, которые создаются из других веток или вливаются в них. Основой интерфейс для работы с Git-ом является консоль/терминал, хотя существуют приложения с дополнительным графическим интерфейсом.

Кроме того Git имеет исходный код, а также активно поддерживается и непрерывно развивается уже более 10 лет. Вокруг Git сформировалось многочисленное сообщество пользователей, а сам проект получает активную поддержку со стороны сообщества. Система обладает подробной и качественной документацией: всем желающим в числе прочего доступны книги, учебные руководства, специализированные веб‑сайты, подкасты и обучающие видеоролики.

1. Создание репозитория

Главным сервисом, предоставляющим удобные механизмы работы с git является GitHub. Регистрация на GitHub ничем не отличается от регистрации на любом другом сайте. После регистрации и авторизации мы можем создать новый репозиторий

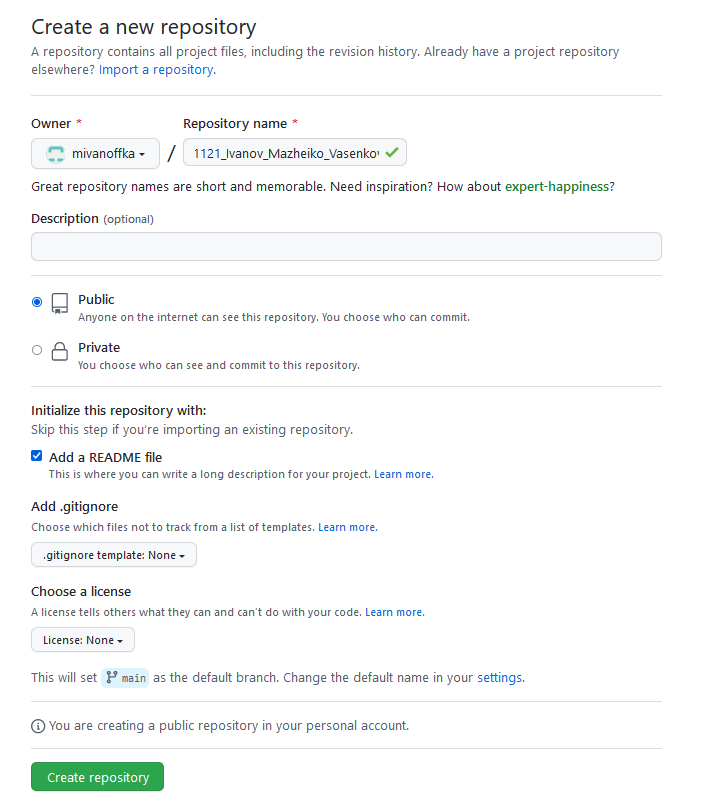
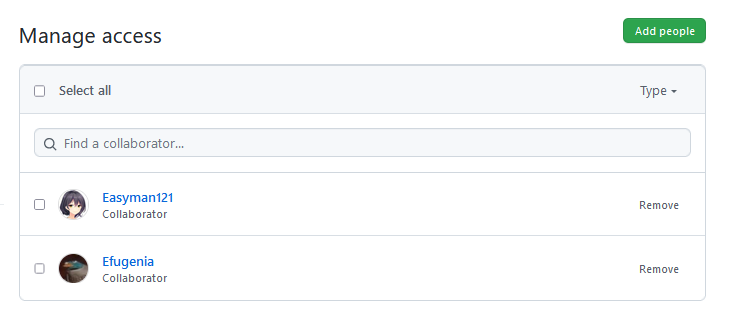


Рисунок 1 – Создание репозитория GitHub

Репозиторий GitHub может использовать не только его непосредственный создатель. Любого другого пользователя сервиса можно присоединить к репозиторию на правах соучастника – “коллаборатора”. Они смогут так же вносить правки в репозиторий.

Рисунок 2 – Коллабораторы репозитория

Для демонстрации работы GitHub создадим тестовый проект в MS Visual Studio и добавим его файлы в наш общий репозиторий

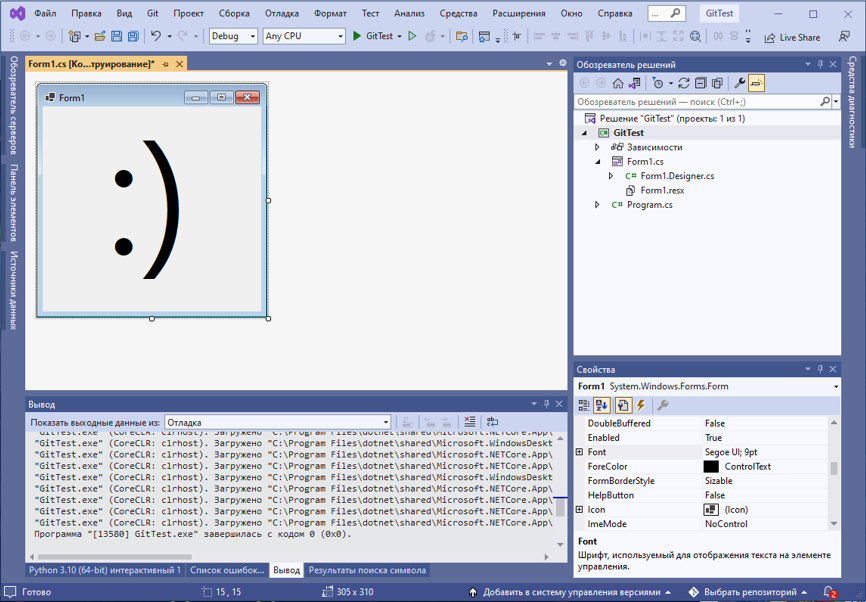


Рисунок 3 – Тестовый проект

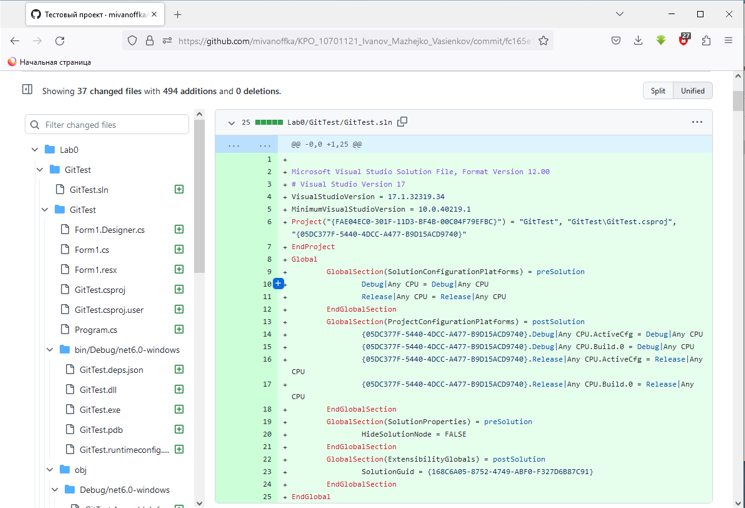


Рисунок 4 – Проект, загруженный в репозиторий

Теперь путём клонирования репозитория проект может быть открыт с другого компьютера.

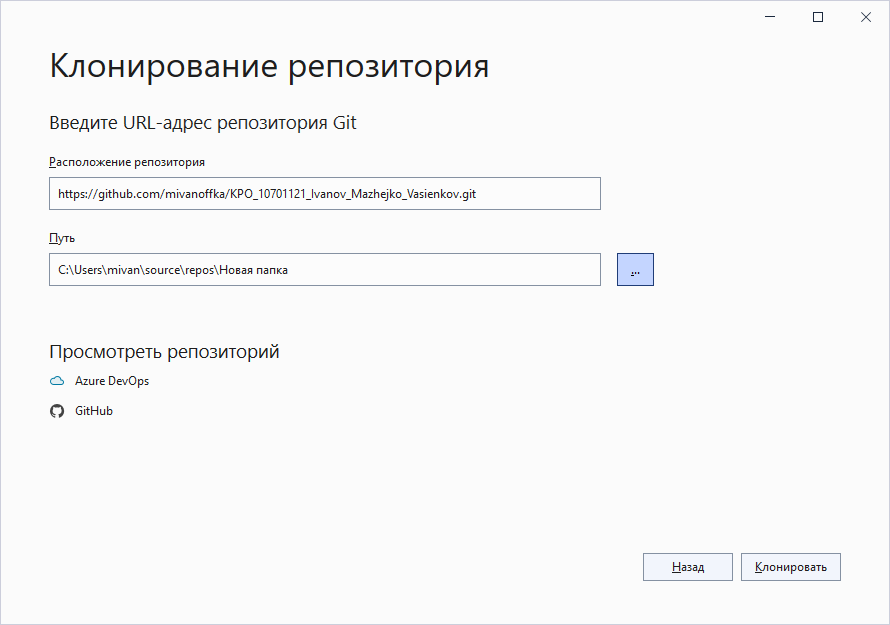


Рисунок 5 – Клонирование репозитория через Visual Studio



Рисунок 6 – Клонирование репозитория через Git Bash

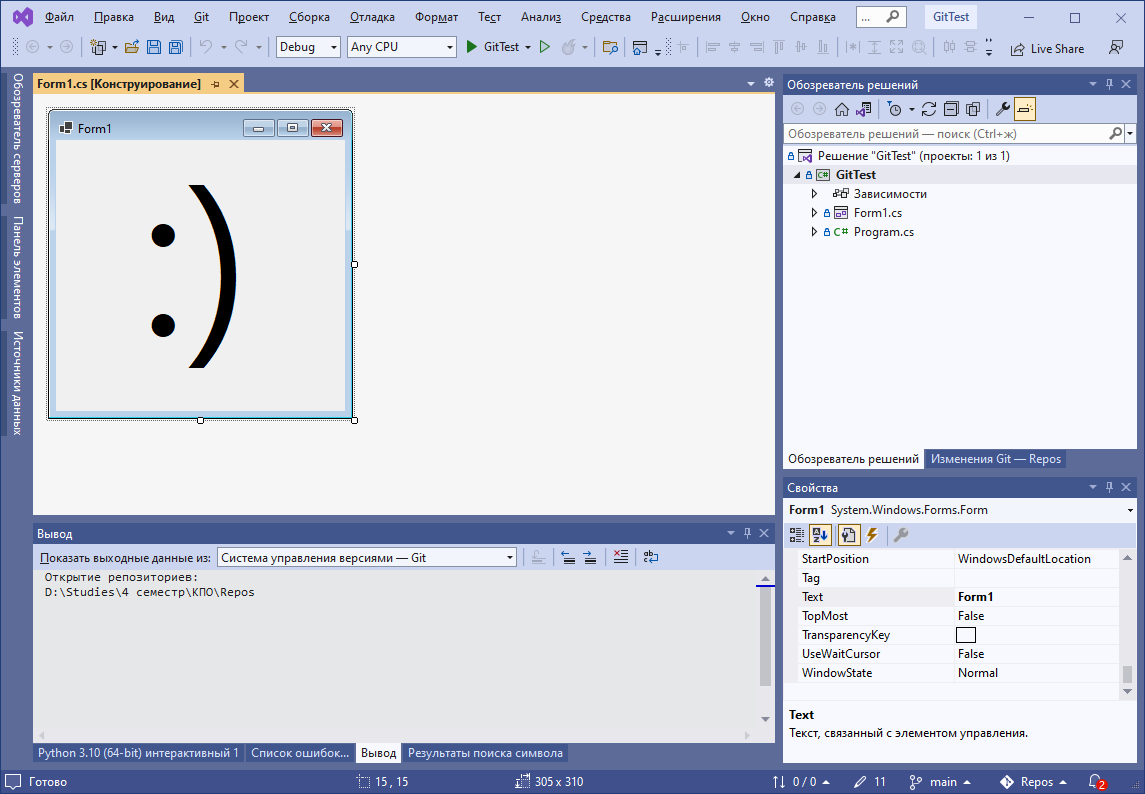


Рисунок 6 – Проект, открытый на другом компьютере

1. Вывод

Система Git и, в частности, сервис GitHub предоставляет широкий набор возможностей для хостинга проектов в сфере информационных технологий, совместной разработки и контроля версий продукта.