**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА   
И ГОСУДАРСТВЕННОЙ СЛУЖБЫ**

**ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»**

**НИЖЕГОРОДСКИЙ ИНСТИТУТ УПРАВЛЕНИЯ – филиал РАНХиГС**

Факультет: управления\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра информатики и информационных технологий \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Направление подготовки / специальность: 09.03.03 прикладная информатика\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*(код, наименование)*

Направленность (профиль) / специализация: корпоративные системы управления \_\_\_\_\_\_\_\_\_\_\_\_\_\_

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ОТЧЁТ (ПО ПРАКТИЧЕСКОЙ/ЛАБОРАТОРНОЙ РАБОТЕ)**

|  |  |
| --- | --- |
| по дисциплине: | Программная инженерия\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| на тему: | «Освоение работы в Git, GitHub, GitLab»\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |
|  |

|  |  |  |
| --- | --- | --- |
|  |  | **АВТОР** |
|  |  | Обучающийся(иеся) \_3\_\_ курса группы ИБ-321\_\_ |
|  |  | \_\_\_\_\_\_\_\_заочной\_\_\_\_\_\_\_\_\_ формы обучения |
|  |  |  |
|  |  | \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_Петраков А.В.\_\_\_\_\_ |
|  |  | *(подпись) (фамилия, инициалы)* |
|  |  |  |
|  |  | **РУКОВОДИТЕЛЬ** |
|  |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  | *(ученая степень, ученое звание)* |
|  |  | оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  | «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_ г. |
|  |  | *(дата защиты)* |
|  |  | \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_Окулич В.И.\_\_\_\_\_\_ |
|  |  | *(подпись) (фамилия, инициалы)* |

Нижний Новгород, 2024г.

**Введение**

Git является распределенной системой управления версиями, разработанной для эффективного управления и контроля изменений в исходном коде и других текстовых файлах проекта. Его основное назначение - обеспечить командную работу над проектами различного масштаба, сохраняя историю изменений, облегчая слияние кода и управление различными версиями проекта.

Основные особенности Git включают:

* описать назначение, возможности и особенности Git
* сравнить GitHub и GitLab
* описать GitFlick
* пройти регистрацию на гитхаб
* создать репозиторий
* провести загрузку файлов в репозиторий

**Общие сведения о Git**

Git является распределенной системой управления версиями, предназначенной для эффективного отслеживания изменений в исходном коде проекта. Она обеспечивает возможность координированной работы нескольких разработчиков над одним и тем же проектом, сохраняя историю изменений и облегчая процесс слияния внесенных изменений.

Git предоставляет инструменты для создания, объединения и управления ветвями разработки, что позволяет разработчикам эффективно работать над отдельными функциональными блоками проекта без возникновения конфликтов между изменениями. Кроме того, система Git обладает высокой скоростью работы и эффективно использует ресурсы компьютера, обеспечивая быстрое выполнение операций над репозиторием.

Основное назначение Git заключается в обеспечении удобного и безопасного управления версиями проекта, повышении прозрачности процесса разработки и обеспечении возможности восстановления предыдущих версий кода в случае необходимости. Git является стандартом в индустрии разработки программного обеспечения и широко используется разработчиками по всему миру для управления версиями своих проектов.

GitHub - это веб-платформа для хостинга репозиториев Git, предназначенная для хранения, управления и совместной работы над проектами разработки программного обеспечения. Она обеспечивает разработчикам инструменты для контроля версий, отслеживания изменений, управления задачами и совместной разработки кода.

GitHub предоставляет возможность создания открытых и частных репозиториев, которые могут содержать исходный код проектов, документацию, изображения и другие файлы. Он поддерживает функции, такие как ветвление и слияние кода, запросы на вытягивание (Pull Requests), отслеживание ошибок (Issue Tracking), управление доступом и многое другое.

**Популярные веб платформы для управления проектами**

GitHub является популярным инструментом для совместной работы над открытыми проектами с открытым исходным кодом, а также для коммерческого использования в различных организациях и командах разработчиков. Он предоставляет широкий набор функций и интеграций с другими сервисами разработки, что делает его одним из ведущих инструментов для управления разработкой программного обеспечения.

GitLab - это веб-платформа для управления репозиториями Git, которая предоставляет инструменты для совместной разработки программного обеспечения. Она позволяет разработчикам хранить код проектов, отслеживать изменения, управлять задачами и вести совместную работу над кодом.

Сравнение основных различий платформ:

* Модель лицензирования. GitHub предоставляет бесплатные и платные планы. Базовые функции доступны бесплатно для открытых проектов, но для частных репозиториев требуется плата. Основные функции GitLab доступны бесплатно для всех типов проектов, включая частные репозитории. Также есть платные планы с дополнительными функциями и поддержкой.
* Функциональные возможности. GitHub предоставляет базовые инструменты для управления версиями, такие как Pull Requests, Issue Tracking и управление доступом. GitLab включает в себя более широкий набор функций, таких как встроенные CI/CD пайплайны, управление задачами, доски задач, вики, регистрация ошибок и многое другое.
* Само-хостинг. GitHub не предоставляет возможность самостоятельного развертывания на собственном сервере. GitLab предлагает возможность самостоятельного развертывания на собственном сервере, что делает его более гибким для организаций, предпочитающих само-управление и контроль над инфраструктурой.
* Цена. Стоимость GitHub зависит от выбранного плана и необходимых функций. Основные функции GitLab доступны бесплатно, но есть платные планы для предприятий и организаций, которые требуют дополнительных возможностей и поддержки.
* Целевая аудитория. GitHub часто используется для открытых проектов с открытым исходным кодом и активно взаимодействует с сообществом разработчиков. GitLab ориентирован как на открытые, так и на частные проекты, включая корпоративные команды и организации, стремящиеся к большей гибкости и контролю.

Обе платформы имеют свои преимущества и недостатки, и выбор между ними зависит от потребностей, бюджета и предпочтений вашей организации или команды разработчиков.

**Российская платформа GitFlick**

GitFlic ― новый отечественный аналог GitHub. С помощью российского облачного сервиса хранения репозиториев исходного кода отечественные вузы и компании, которые находятся под санкциями, найдут поставщика услуг для размещения исходного кода своих технологических решений. При этом данные будут храниться в расположенных на территории РФ сертифицированных дата-центрах.

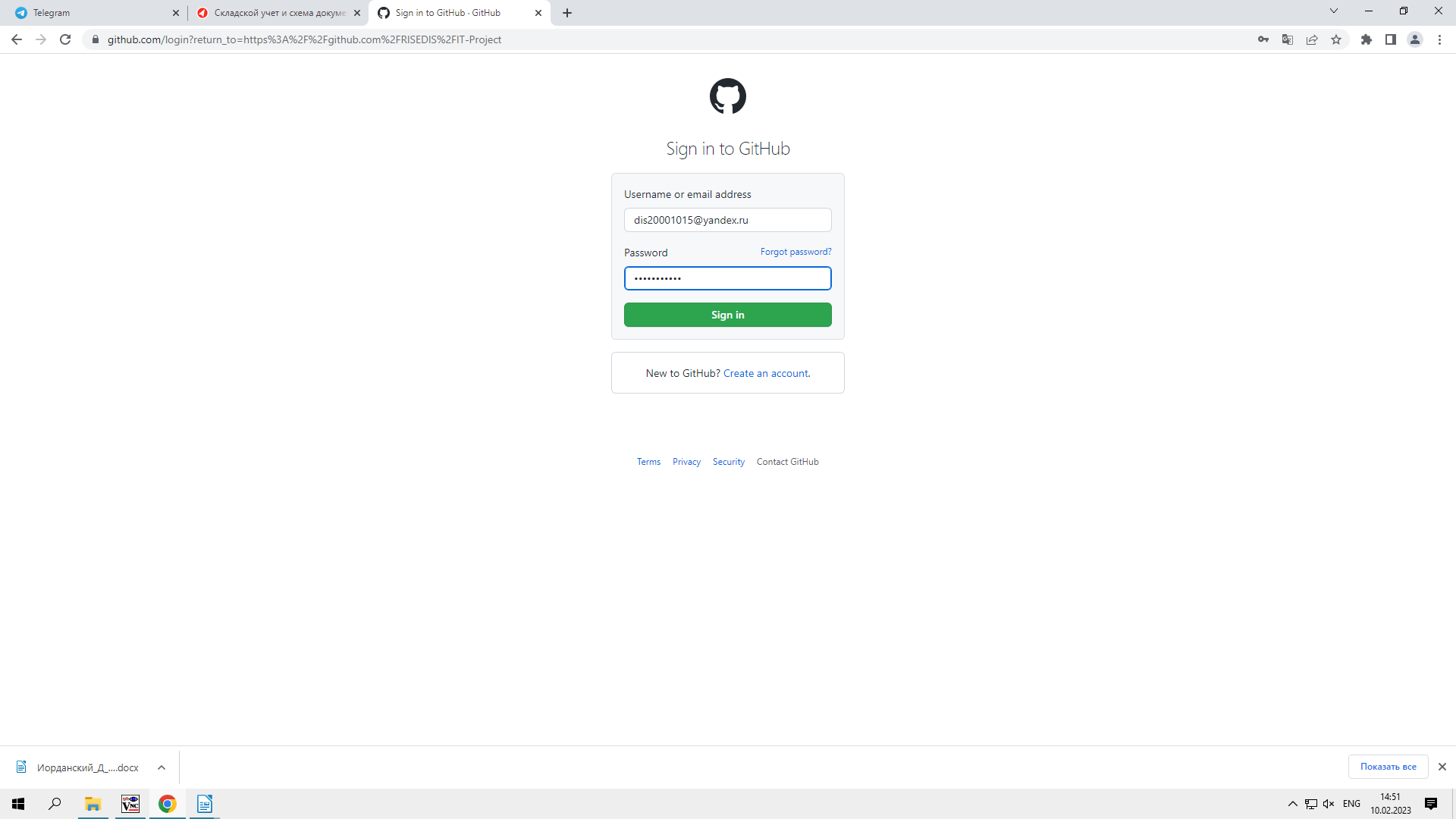
### Задачи, решаемые с помощью платформы:

* Организация рабочего процесса написания кода разработчиками.
* Защита исходного кода от потери. Данные хранятся на удалённом сервере, даже если разработчики удалят файлы с локального компьютера, они останутся в репозитории.
* Обеспечение командной работы.
* Контроль изменений. В любой момент можно вернуться к контрольной точке, сравнить исходный код с текущим и обновить главную ветку после ревью.
* Распределённая работа. Возможность работать с несколькими версиями разрабатываемого проекта.

**Работа с GitHub**

**Регистрация на GitHub**

Регистрация и авторизация на сайте GitHub показана на рис.1

Рис.1 Форма авторизации на сайте GitHub

**Создание репозитория**

Переход в профиль во вкладку репозитории показан на рис.2

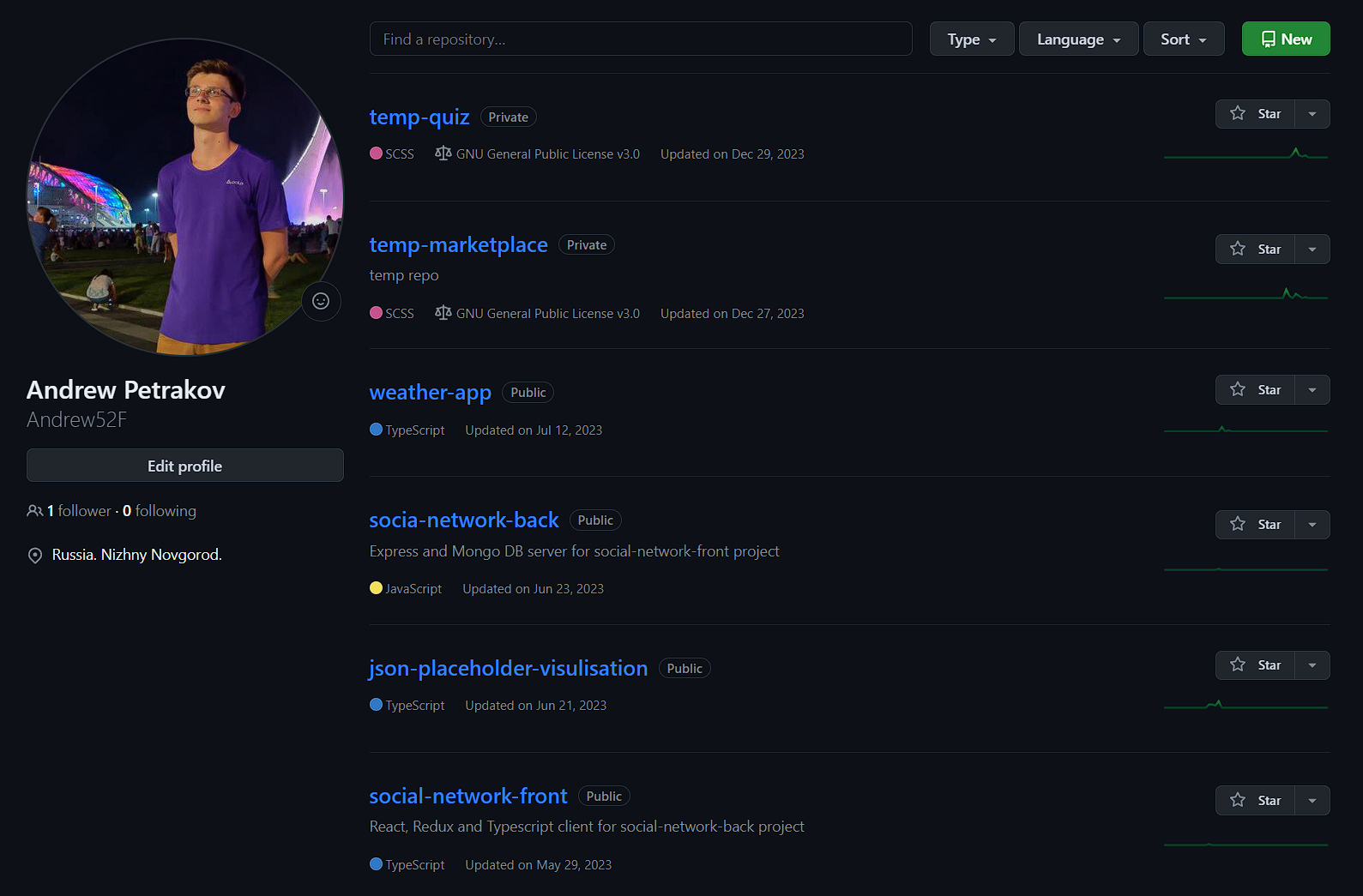


Рис.2 Вкладка репозитории в профиле

Добавление репозитория открытого типа.

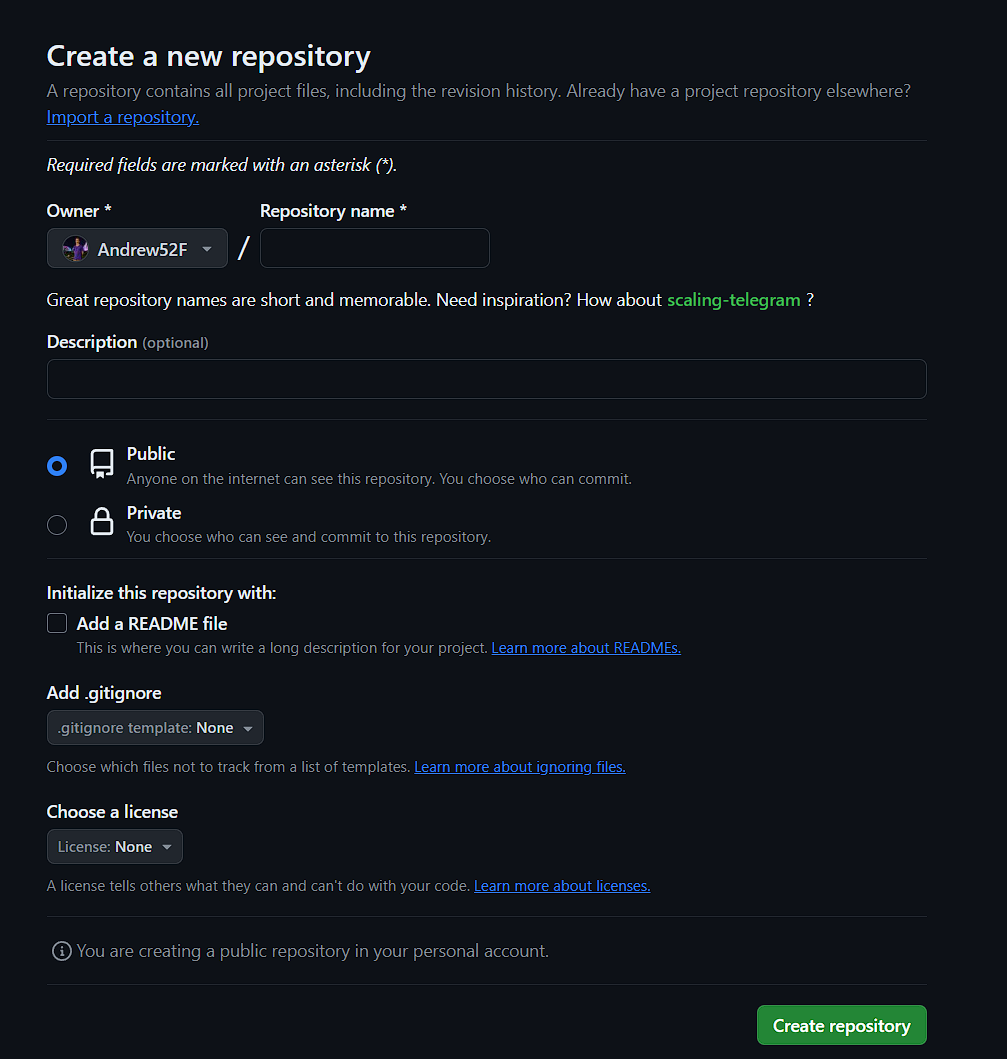


Рис.3 Форма создания репозитория и выбора режима доступности

Успешно загрузили файлы в репозиторий.

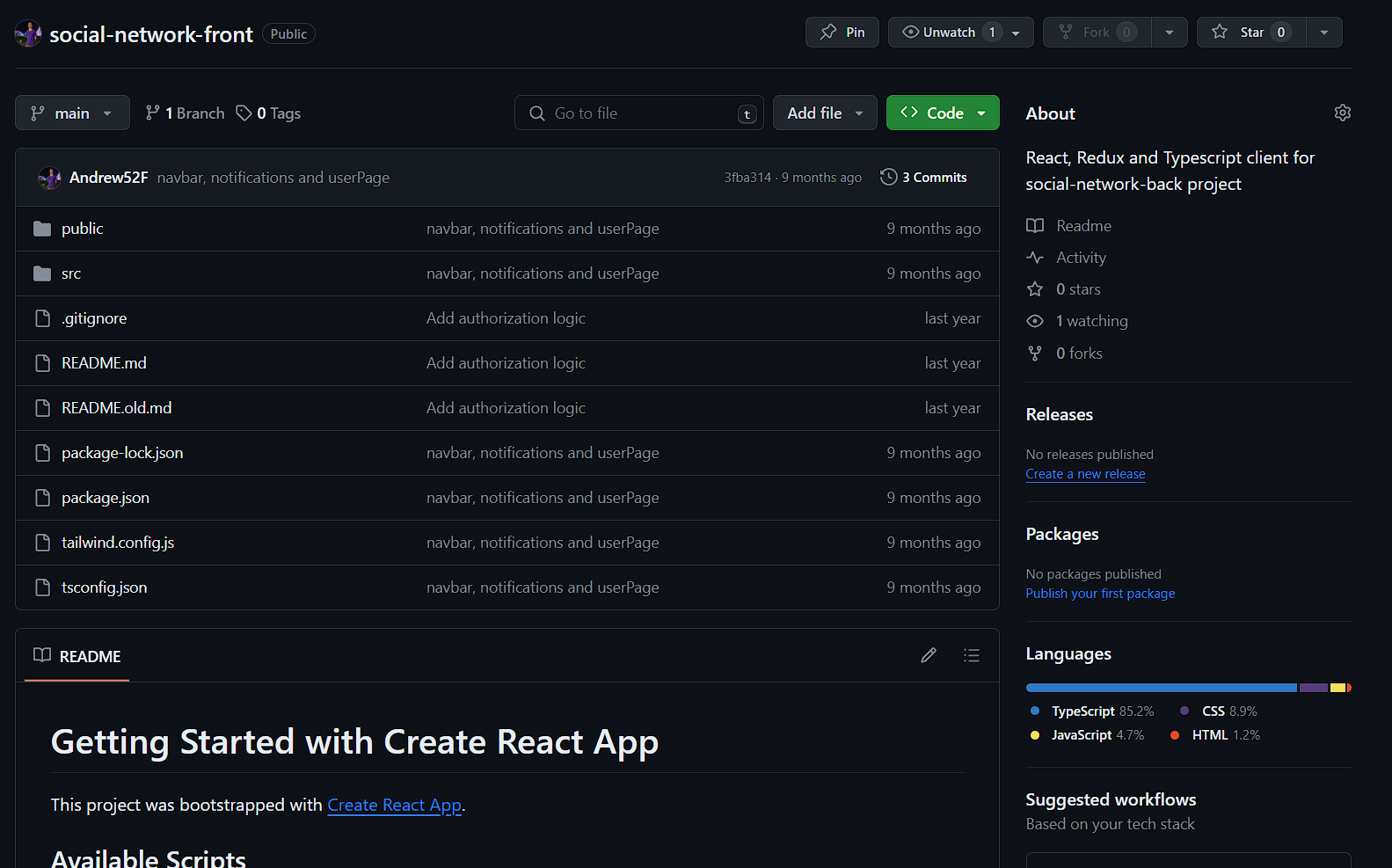


Рис.6 Наполненный репозиторий

Ссылка на репозиторий: https://github.com/Andrew52F/social-network-front

**Заключение**

Успешно был выполнен ряд ключевых задач. Регистрация на платформе GitHub дала нам возможность войти в мир современной разработки программного обеспечения и стать частью развивающегося сообщества разработчиков.

Создание репозитория стало первым шагом к систематизации наших проектов и обеспечило централизованное хранилище для их управления. Загрузка файлов в репозиторий позволила нам эффективно сохранять и организовывать код, документацию и другие ресурсы проекта.

Параллельно с этим, наше погружение в технологию Git, а также изучение особенностей GitHub и GitLab, расширило наши знания и навыки в области современной разработки программного обеспечения. Мы приобрели понимание основных принципов работы с версиями кода, научились эффективно использовать ветвление, слияние и управление изменениями.

Изучение платформ GitHub и GitLab позволило нам оценить их возможности для совместной работы над проектами, управления задачами и использования инструментов непрерывной интеграции и развертывания. Этот опыт придаст нам уверенности в работе с современными инструментами разработки и поможет в дальнейшем участии в профессиональных проектах.

В целом, процесс освоения технологии работы с репозиториями и системами контроля версий открывает перед нами новые горизонты в сфере разработки программного обеспечения и предоставляет нам инструменты для более эффективной и организованной работы над будущими проектами.