

**ОТЧЕТ О ВЫПОЛНЕНИИ**

**практической работы № 5**

**по предмету МДК 3.1**

**«Моделирование и анализ ПО»**

Выполнили: ст. гр. 2ИСП11-31ВБ Синицын О. Е. Кузнецов Е. А.

Специальность: 09.02.07

Информационные системы и программирование

Проверил: преподаватель Боклач Б. И.

Москва 2023

**Тема:** Планирование CODE-REVIEW.

**Цель работы:** Получение практических навыков подготовки среды совместной разработки для проведения инспекции программного года.

**Ход работы:**

**Задание 1**

Создаем репозиторий

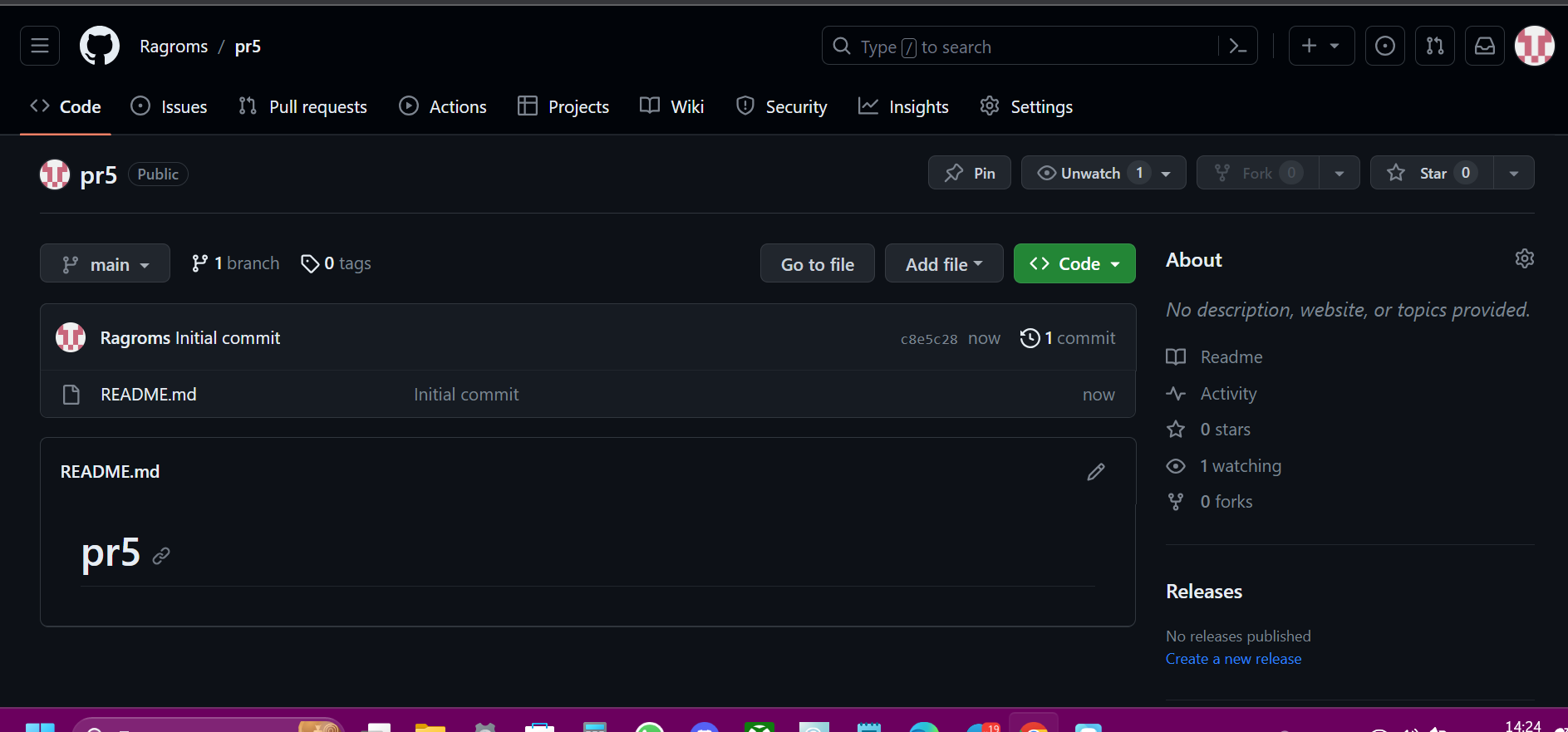
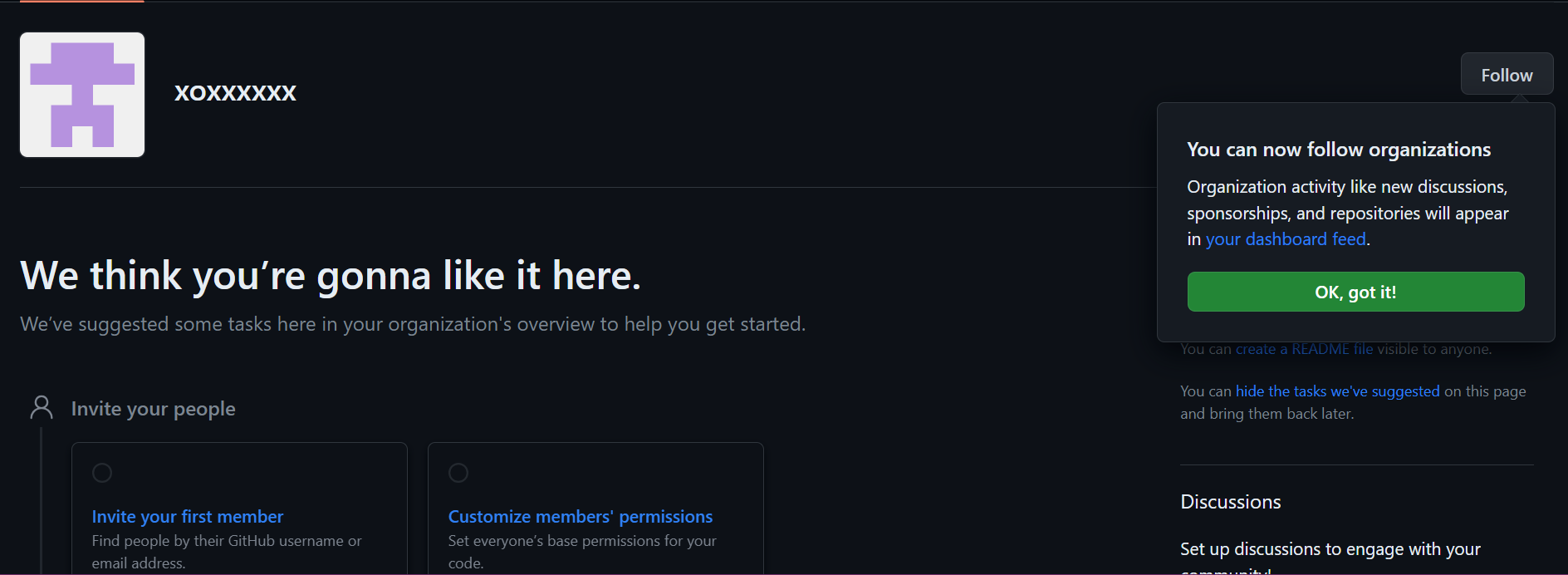


Рис.1

**Задание 2**

Создаем организацию

Рис.2

**Задние 3**

Приглашаем доп. участника через почту, после того как второй участник присоединяется к организации после подтверждения запроса отправленного ему на почту.

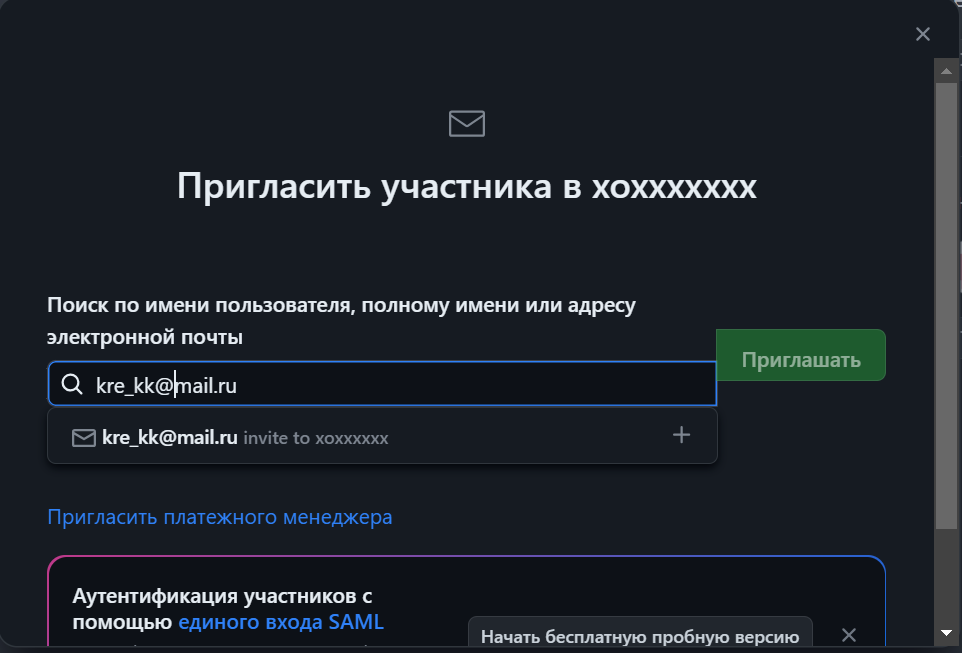


Рис.3

**Задание 4**

Задаем параметры необходимые для отправки и слияния

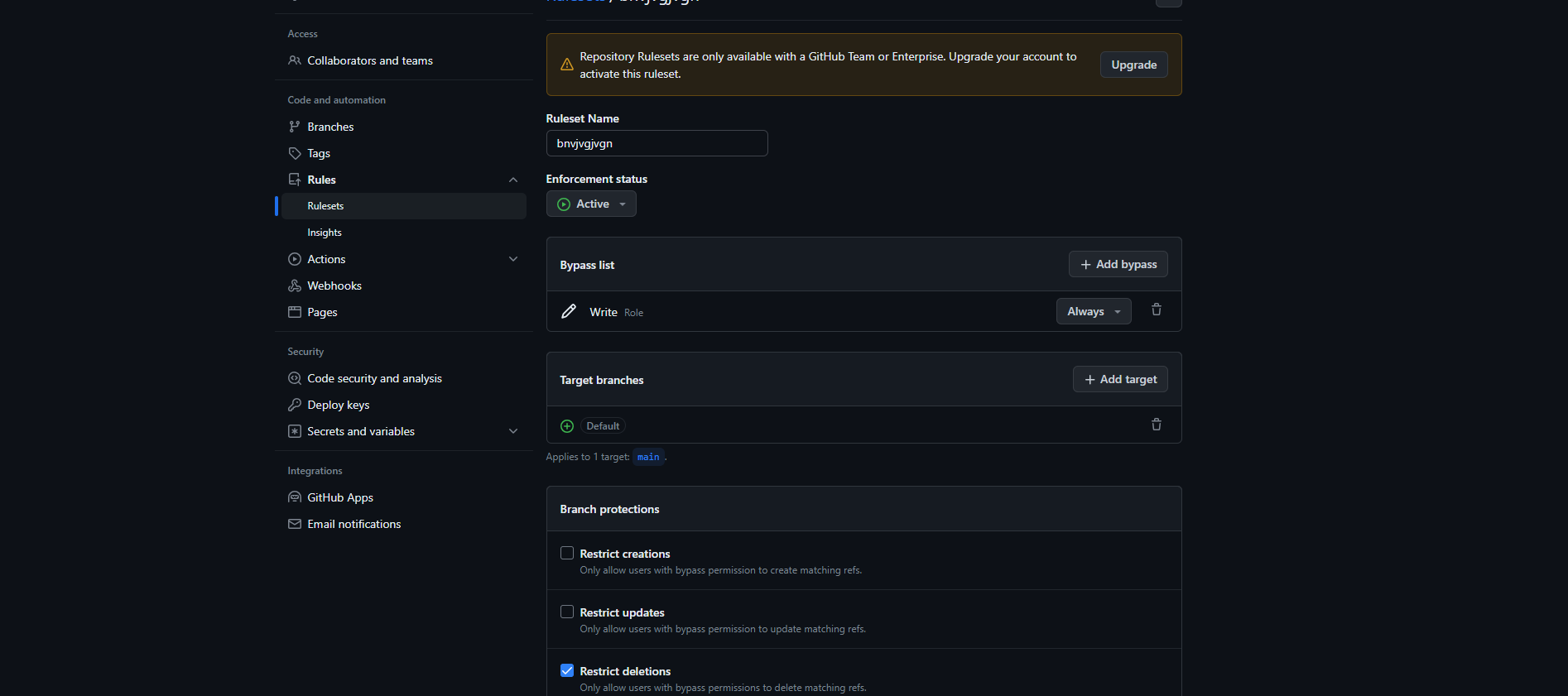


Рис.4

**Вывод:** Мы получили навыки подготовки среды совместной разработки для проведения инспекции программного года.

**Ответы на контрольные вопросы:**

1. **Какие действия необходимо выполнить что бы подготовить среду разработки для инспекции кода?**

- Установить и настроить необходимые инструменты, такие как статический анализатор кода (например, SonarQube) и систему контроля версий (например, Git).

- Создать проект или репозиторий для хранения исходного кода.

- Настроить инструменты для автоматической проверки кода при каждом коммите или пуше в репозиторий.

- Запустить инспекцию кода и проанализировать полученные результаты.

1. **Какие методы существуют для настройки Github для совместной работы нескольких разработчиков?**

- Создание команды или организации в Github, где участники могут работать над проектом вместе.

- Назначение разработчикам соответствующих прав доступа к репозиторию, чтобы они могли вносить изменения и рецензировать код.

- Использование функциональности pull request для обсуждения и слияния изменений от разных разработчиков.

- Использование функциональности issues для отслеживания задач и обсуждения проблем.

1. **Что такое организация разработки в сервисе Github?**

Организация разработки в сервисе Github представляет собой создание команды или организации, в которой участники могут работать над проектом вместе. В рамках организации можно создавать и управлять репозиториями, назначать разработчикам права доступа и управлять задачами и обсуждениями.

1. **Какие уровни доступа могут быть заданы для организации в Github?**

- Владелец (Owner): имеет полный доступ ко всем репозиториям и может управлять настройками организации.

- Администратор (Admin): имеет полный доступ к репозиториям организации, но не может управлять настройками организации.

- Писатель (Write): может изменять код и управлять задачами и обсуждениями в репозитории.

- Читатель (Read): имеет только чтение доступа к репозиторию.

1. **Какие параметры должны быть заданы при настройки отправки и слияния копии репозитория?**

При настройке отправки и слияния копии репозитория необходимо задать следующие параметры:

- Исходный репозиторий (Source repository): указывает на репозиторий, из которого будут отправляться изменения.

- Целевой репозиторий (Target repository): указывает на репозиторий, в который будут сливаться изменения.

- Ветка исходного репозитория (Source branch): указывает на ветку, из которой будут отправляться изменения.

- Ветка целевого репозитория (Target branch): указывает на ветку, в которую будут сливаться изменения.

- Описание изменений (Change description): описывает суть внесенных изменений.

1. **Какие модели применяются при создания копии репозитария?**

- Fork: создается точная копия репозитория, в которой разработчик может вносить изменения и предлагать их для слияния обратно в исходный репозиторий.

- Clone: создается локальная копия репозитория на компьютере разработчика, где он может работать над проектом и отправлять изменения обратно в удаленный репозиторий.

- Branch: создается новая ветка в репозитории, где разработчик может вносить изменения и предлагать их для слияния с другими ветками.

1. **Что такое «обзор кода»?**

Обзор кода (code review) - это процесс проверки и анализа кода другими разработчиками с целью выявления ошибок, улучшения качества и обмена знаниями. В ходе обзора кода разработчики могут комментировать и предлагать изменения в коде, обсуждать возможные улучшения и рекомендовать исправления. Обзор кода является важной практикой в разработке программного обеспечения для повышения качества и эффективности работы.