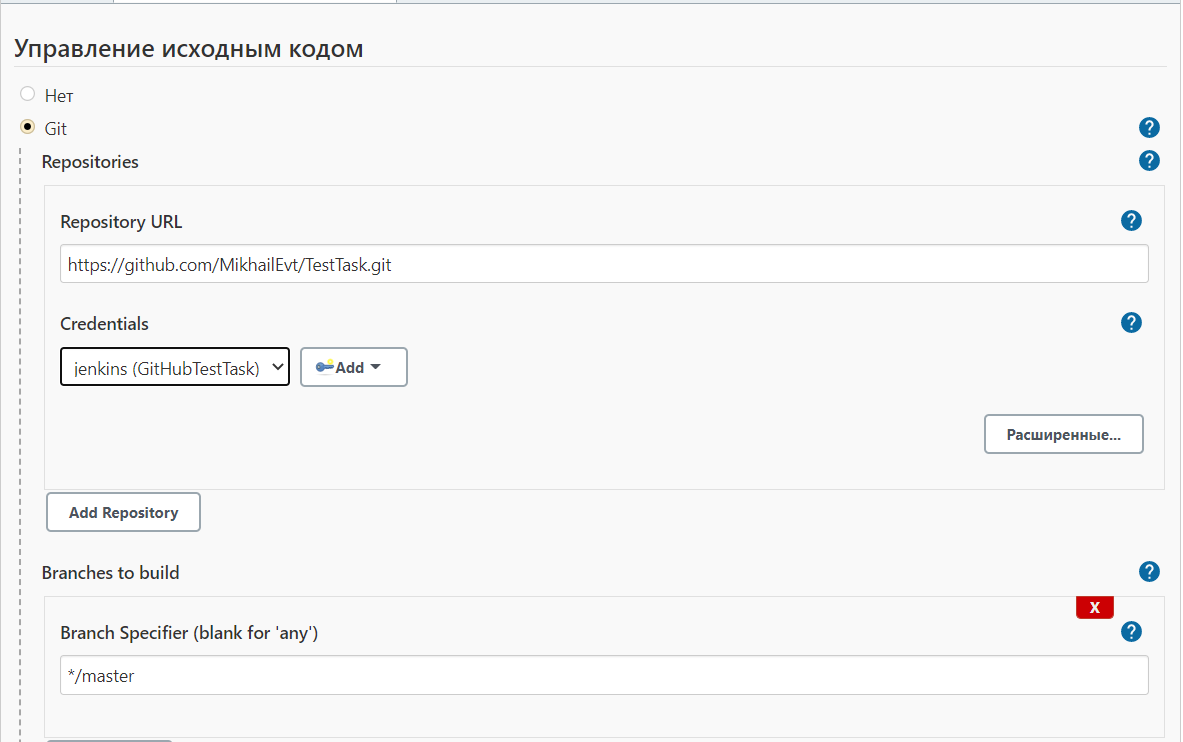
**Пояснения к тестовому заданию.**

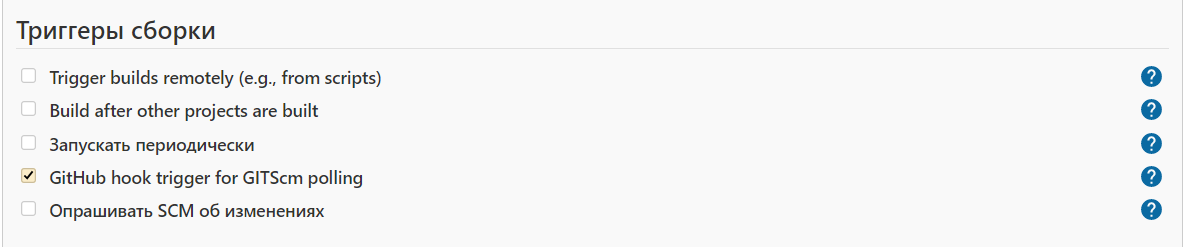
URL-адрес Jenkins: <http://3.92.68.231:8080> Логин, пароль: admin:admin.

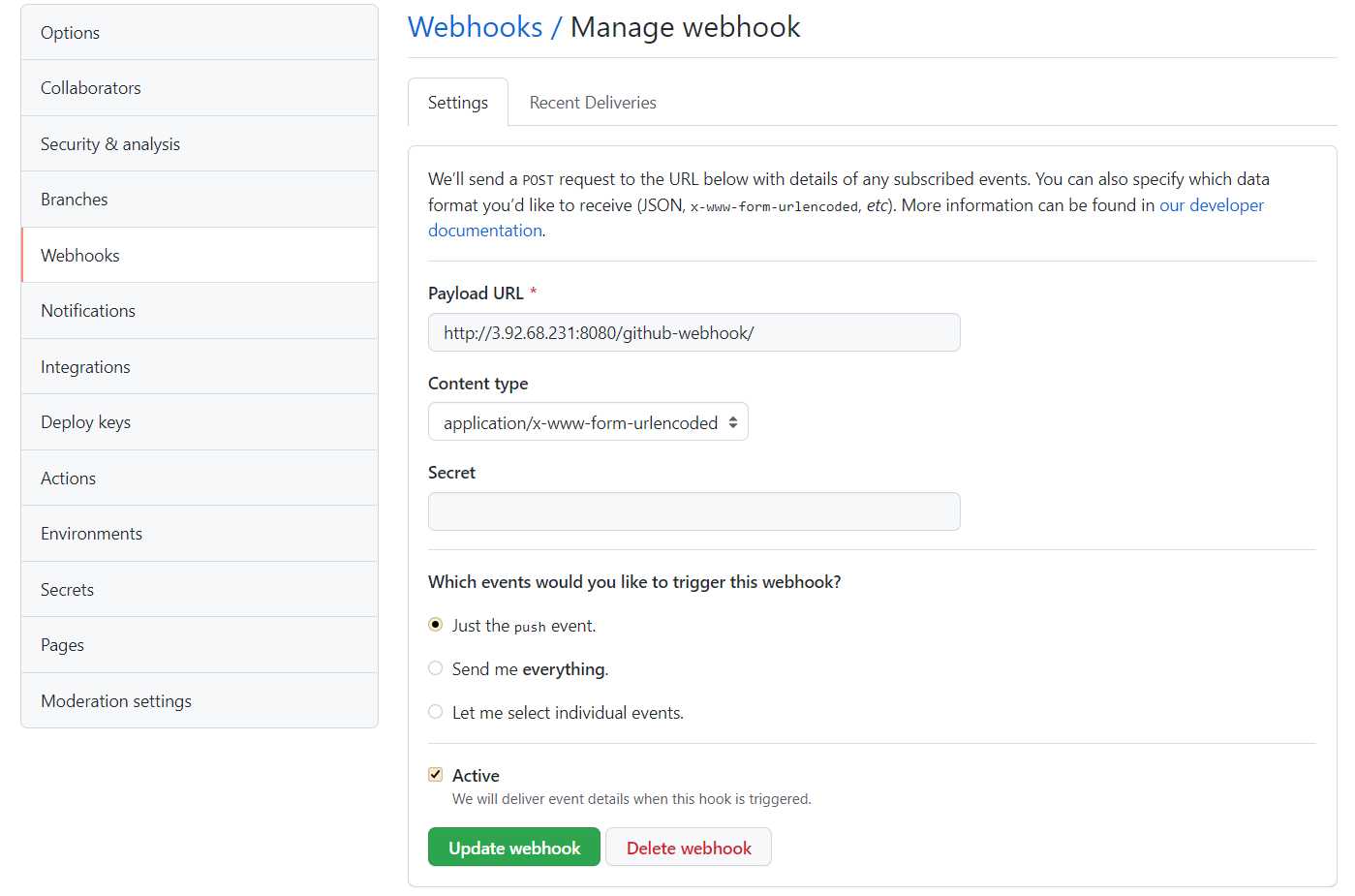
Для выполнения тестового задания были выполнены следующие действия.

1. Была создана библиотека классов и в ней добавлен метод RandNumb возвращающий случайное число от 0 до 5.
2. К проекту добавлен тест, который при значении метода 0 или 1 считается пройденным и при других значениях не пройденным.
3. Проект был помещен в репозиторий на GitHub.
4. Для размещения Jenkins был выбран виртуальный сервер на Ubuntu Server 20.04
5. Для сборки проекта на сервере был установлен .NET SDK.
6. Также была установлена необходимая для Jenkins версия JAVA и сам Jenkins.
7. Была произведена первоначальная настройка Jenkins, а также добавлены плагины MSTest plugin и MSTestRunner plugin для последующего просмотра результатов тестов.
8. Далее была создана задача со свободной конфигурацией.
   1. Добавлена ссылка на GitHub репозиторий и создана пара ключей для доступа Jenkins.



* 1. Триггером для сборки выступают изменения в репозитории GitHub, для этого в Jenkins выставлен соответствующий параметр и добавлен webhook в GitHub.

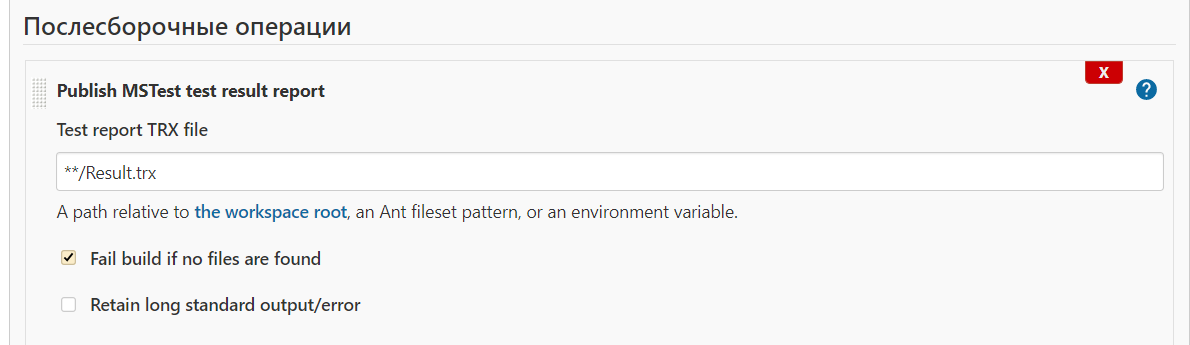




* 1. Первым и единственным шагом сборки была добавлена команда shell. 

dotnet buid выполняет сборку проекта, dotnet test запускает тесты и сохраняет результат в файле Result.trx

* 1. Для настройки публикации тестов добавляется соответствующий шаг и указывается путь к файлу результатов.



* 1. Далее сборка задачи может запускатся вручную или после изменения файлов в репозитории GitHub.
  2. В статусе задачи теперь можно посмотреть последние результаты тестов, график результатов, лог запросов от GitHub, историю сборок.

