

**Министерство науки и высшего образования Российской Федерации**  
Федеральное государственное автономное образование учреждение  
высшего образования “Национальный исследовательский университет  
ИТМО”

## **ОТЧЁТ**

По лабораторной работе №2  
По дисциплине “Основы Программной Инженерии”  
Вариант 12615

**Авторы:**

Билошицкий Михаил Владимирович  
Трошкин Александр Евгеньевич

**Факультет:**

ПИиКТ

**Преподаватель:**

Письмак Александр Евгеньевич

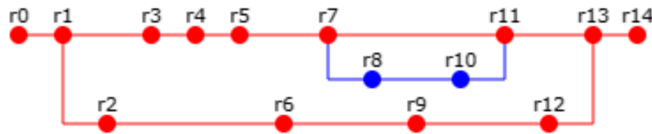


Санкт-Петербург, 2024

<b>Содержание</b>	<b>2</b>
<b>Задание</b>	<b>3</b>
<b>Выполнение</b>	<b>4</b>
Github	4
SVN	4
<b>Вывод</b>	<b>7</b>

# Задание

Вариант 12615



Сконфигурировать в своём домашнем каталоге репозитории svn и git и загрузить в них начальную ревизию файлов с исходными кодами (в соответствии с выданным вариантом).

Воспроизвести последовательность команд для систем контроля версий svn и git, осуществляющих операции над исходным кодом, приведённые на блок-схеме.

При составлении последовательности команд необходимо учитывать следующие условия:

- Цвет элементов схемы указывает на пользователя, совершившего действие (красный - первый, синий - второй).
- Цифры над узлами - номер ревизии. Ревизии создаются последовательно.
- Необходимо разрешать конфликты между версиями, если они возникают.

Отчёт по работе должен содержать:

1. Задание и блок-схему в соответствии с вариантом.
2. Список команд, использованных при создании и конфигурации репозитория в домашнем каталоге пользователя.
3. Номера ревизий и соответствующие им последовательности команд с комментариями (для svn и git).
4. Выводы по работе.

Вопросы к защите лабораторной работы:

1. Системы контроля версий - назначение, примеры решений.
2. Ревизии и ветки.
3. Основные операции над данными в системах контроля версий. Основные команды svn и git.
4. Виды конфликтов и способы их решения.

# Выполнение

## Git

Поскольку тут всё итак достаточно очевидно, мы решили даже не расписывать, что и почему. В случае чего можем пересказать устно

## SVN

```
svnadmin create svnRepo
cd svnRepo/
pathf=file:///home/studs/s367101/svnRepo
svn mkdir ${pathf}/trunk ${pathf}/branches ${pathf}/tags -m "feat:
create repo" --username="MichaelB"
svn checkout ${pathf}/trunk workspace
cd workspace

# r0
cp -f ~/commits/commit0/* ./
svn add *
svn commit -m "r0" --username="MichaelB"

# r1
cp -f ~/commits/commit1/* ./
svn add *
svn commit -m "r1" --username="MichaelB"

# r2
svn copy ${pathf}/trunk/ ${pathf}/branches/feature-1 -m "feature-1"
--username="klnd_cat"
svn switch ${pathf}/branches/feature-1
cp -f ~/commits/commit2/* ./
svn add *
svn commit -m "r2" --username="klnd_cat"

# r6
cp -f ~/commits/commit6/* ./
svn add *
```

```
svn commit -m "r6" --username="klnd_cat"

# r9
cp -f ~/commits/commit9/* ./
svn add *
svn commit -m "r9" --username="klnd_cat"

# r12
cp -f ~/commits/commit12/* ./
svn add *
svn commit -m "r12" --username="klnd_cat"

# r3
svn switch ${pathf}/trunk
cp -f ~/commits/commit3/* ./
svn add *
svn commit -m "r3" --username="MichaelB"

# r4
cp -f ~/commits/commit4/* ./
svn add *
svn commit -m "r4" --username="MichaelB"

# r5
cp -f ~/commits/commit5/* ./
svn add *
svn commit -m "r5" --username="MichaelB"

# r7
cp -f ~/commits/commit7/* ./
svn add *
svn commit -m "r7" --username="MichaelB"

# r8
svn copy ${pathf}/trunk/ ${pathf}/branches/feature-2 -m "feature-2"
--username="MichaelB"
svn switch ${pathf}/branches/feature-2
cp -f ~/commits/commit8/* ./
```

```
svn add *
svn commit -m "r8" --username="k1nd_cat"

# r10
cp -f ~/commits/commit10/* ./
svn add *
svn commit -m "r10" --username="MichaelB"

# r11
svn switch ${pathf}/trunk
svn merge ${pathf}/branches/feature-2 ${pathf}/trunk
--username="MichaelB"
cp -f ~/commits/commit11/* ./
svn add *
svn commit -m "r11" --username="MichaelB"

#13
svn merge ${pathf}/branches/feature-1 ${pathf}/trunk
--username="MichaelB"
cp -f ~/commits/commit13/* ./
svn add *
svn commit -m "r13" --username="MichaelB"

# r14
cp -f ~/commits/commit14/* ./
svn add *
svn commit -m "r14" --username="MichaelB"
```

## Вывод

В данной лабораторной мы ближе ознакомились с наиболее популярными в наше время системами контроля версий, поработали с git и svn, поняли, чем они отличаются друг от друга, какие проблемы могут возникать при работе с ними и как с этими проблемами справляться.