|  |  |  |
| --- | --- | --- |
|  |  | УТВЕРЖДАЮ |
|  |  | Начальник тех.отдела |
|  |  | \_\_\_\_\_\_\_\_\_\_\_\_\_ Хххххх Х.Х. |
|  |  | “\_\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_200\_\_ |

**Система контроля версий git**

**Техническое задание**

**А.В.00001-01 ТЗ 01-лу**

**А.В.00001-01 ТЗ 01**

**гибкий магнитный диск**

Инв. № подл.

Подпись и дата

Взам. инв. №

Инв. № дубл.

Подпись и дата

|  |  |  |
| --- | --- | --- |
|  |  | Руководитель разработки |
|  |  | Начальник ХХХХ |
|  |  | \_\_\_\_\_\_\_\_\_\_\_Хххххххх Х.Х. |
|  |  | “\_\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_200\_\_ |
|  |  |  |
|  |  | Исполнитель |
|  |  | Вед. инженер ХХХХ |
|  |  | \_\_\_\_\_\_\_\_\_\_\_\_\_Хххххх Х.Х. |
|  |  | “\_\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_200\_\_ |

Литера

**УТВЕРЖДЕНО**

А.В.00001-01 ТЗ 01-лу

**система контроля версий git**

**Техническое задание**

**Инв. № подл.**

**Подпись и дата**

**Взам. инв. №**

**Инв. № дубл.**

**Подпись и дата**

**А.В.00001-01 ТЗ 01**

**ЛИСТОВ 16**

Литера

# Содержание

Содержание………………………………………………………………….2

1. Введение…………………………………………………………………..4

1.1. Наименование команды…………………………………………………...4

1.2. Краткая характеристика области применения команды………………...4

[2. Основание для разработки……………………………………………….5](#_Toc119204107)

[2.1. Основание для проведения разработки…………………………………..5](#_Toc119204108)

3. Назначение разработки…………………………………………………..6

3.1. Функциональное назначение команды…………………………………...6

3.2. Эксплуатационное назначение команды…………………………………6

[4. Требования к программе…………………………………………………](#_Toc119204113)7

[4.1. Требования к функциональным характеристикам………………………](#_Toc119204114)7

[4.1.1. Требования к составу выполняемых функций…………………………….](#_Toc119204115)8

[4.1.2. Требования к организации входных данных](#_Toc119204116)……………………………...8

[4.1.3. Требования к временным характеристикам](#_Toc119204118)…………………………...…...8

[4.2. Требования к надежности](#_Toc119204119) 9

[4.2.1. Требования к обеспечению надежного (устойчивого) функционирования программы](#_Toc119204120)…………………………………………………………………………...9

[4.2.2.](#_Toc119204121) Время восстановления после отказа…………………………………………9

[4.2.3.](#_Toc119204121) Отказы из-за некорректных действий оператора…………………………...10

[4.3. Условия эксплуатации](#_Toc119204122)……………………………………………………...10

[4.3.1.](#_Toc119204123) Климатические условия эксплуатации………………………………………10

[4.3.2.](#_Toc119204124) Требования к видам обслуживания…………………………………………10

[4.3.3.](#_Toc119204124) Требования к численности и квалификации персонала…………...………10

[4.4. Требования к составу и параметрам технических средств](#_Toc119204125)……….…………………………………………………………………..11

[4.5. Требования к информационной и программной совместимости](#_Toc119204128)……….11

[4.5.1.](#_Toc119204129) Требования к информационным структурам и методам решения……………...11

[4.5.2.](#_Toc119204130) Требования к исходным кодам и языкам программирования…………….11

[4.5.3.](#_Toc119204130) Требования к программным средствам, используемым программой…...11

[4.5.4.](#_Toc119204130) Требования к защите информации и программ …………………………...12

[4.6. Специальные требования](#_Toc119204131) 12

[5. Требования к программной документации](#_Toc119204134) 13

[5.1. Предварительный состав программной документации](#_Toc119204135) 13

[5.2. Специальные требования к программной документации](#_Toc119204136) 13

[6. Технико-экономические показатели](#_Toc119204137) 14

[6.1. Ориентировочная экономическая эффективность](#_Toc119204138) 14

[6.2. Предполагаемая годовая потребность](#_Toc119204139) 14

[7. Стадии и этапы разработки](#_Toc119204141) 15

[7.1. Стадии разработки](#_Toc119204142) 15

[8. Порядок контроля и приемки](#_Toc119204147) 16

[8.1. Виды испытаний](#_Toc119204148) 16

[8.2. Общие требования к приемке работы](#_Toc119204149) 16

[9. Уточнение ТЗ](#_Toc119204147) 17

# Введение

## **Наименование команды**

Наименование - «Git».

## **Краткая характеристика области применения команды**

GIT – это распределенная система контроля версий, она используется для облегчения работы с изменяющейся информацией. Система контроля версий позволяет хранить несколько версий одного и того же документа, при необходимости возвращаться к более ранним версиям. Распределенные системы контроля версий имеют ряд преимуществ. Они не нуждаются в централизованном хранилище: вся история документов хранится на каждом компьютере, в локальном хранилище, и при необходимости отдельные фрагменты истории локального хранилища синхронизируется с аналогичным хранилищем на другом компьютере.

# Основание для разработки

## **Основание для проведения разработки**

Основанием для проведения разработки является задание на лабораторную работу по дисциплине «Теория разработки программного обеспечения».

# Назначение разработки

## **Функциональное назначение команды**

Функциональным назначением программы является автоматизирование работы с историей файла (или группы файлов), обеспечение мониторинга изменений, синхронизация данных и организация хранилища проекта.

## **Эксплуатационное назначение команды**

Команда должна эксплуатироваться в профильных подразделениях на объектах ОАО «НИИАА».

Конечными пользователями команды должны являться сотрудники профильных подразделений ОАО «НИАА».

# Требования к программе

## **Требования к функциональным характеристикам**

### Требования к составу выполняемых функций

**4.1.1.1 Команда git add**

Данная команда производит добавление содержимого рабочей директории в индекс (staging area) для последующего коммита.

**4.1.1.2 Команда git status**

Данная команда выводит на экран информацию о состоянии файлов в рабочей директории и индексе.

**4.1.1.3 Команда git diff**

Данная команда производит вычисление разницы между любыми двумя Git деревьями.

**4.1.1.4 Команда git difftool**

Данная команда производит запуск внешней утилиты сравнения для показа различий в двух деревьях.

**4.1.1.5 Команда git commit**

Данная команда сохраняет слепок всех данных, добавленных в индекс.

**4.1.1.6 Команда git reset**

Данная команда производит отмену изменения.

**4.1.1.7 Команда git rm**

Данная команда производит удаление файлов из индекса и рабочей директории.

**4.1.1.8 Команда git mv**

Данная команда производит перемещение файла.

**4.1.1.9 Команда git clean**

Данная команда производит очищение рабочей директории от результатов сборки проекта, файлов конфликтов состояний и т.д.

**4.1.1.10 Команда git branch**

Данная команда запускает утилиту “Менеджер веток”.

**4.1.1.11 Команда git checkout**

Данная команда производит переключение веток и выгрузку их в рабочую директорию.

**4.1.1.12 Команда git merge**

Данная команда производит слияние одной или нескольких веток в текущую.

**4.1.1.13 Команда git log**

Данная команда выводит на экран информацию истории коммитов.

**4.1.1.14 Команда git stash**

Данная команда производит временное сохранение всех незаконченных изменений для очистки рабочей директории.

**4.1.1.15 Команда git tag**

Данная команда задает постоянную метку на какой-либо момент в истории проекта.

**4.1.1.16 Команда git fetch**

Данная команда связывается с удаленным репозиторием и загружает все изменения.

**4.1.1.17 Команда git push**

Данная команда устанавливает связь с удаленным репозиторием, вычисляет локальные изменения, отсутствующие в нем и передает их в репозиторий.

**4.1.1.18 Команда git archive**

Данная команда производит выгрузку в архив указанных коммитов или всего репозитория.

**4.1.1.19 Команда git show**

Данная команда отображает информацию о метке или о коммите.

**4.1.1.20 Команда git bisect**

Данная команда запускает утилиту для поиска коммита, в котором впервые появился баг.

**4.1.1.21 Команда git grep**

Данная команда производит поиск любой строки или регулярного выражения в любом из файлов проекта.

**4.1.1.22 Команда git gc**

Данная команда производит удаление ненужных файлов из хранилища объектов и сортировка оставшихся файлов.

**4.1.1.23 Команда git fsck**

Данная команда производит проверку внутренней базы данных на предмет наличия ошибок и несоответствий.

### Требования к организации входных данных

Входные данные программы должны вводиться в консоль по определенному шаблону. В системе должны быть реализованы функции проверки корректности данных.

### Требования к временным характеристикам

Требования к временным характеристикам программы не предъявляются.

## **Требования к надежности**

### Требования к обеспечению надежного (устойчивого) функционирования команды

Надежное (устойчивое) функционирование программы должно быть обеспечено выполнением совокупности организационно-технических мероприятий, перечень которых приведен ниже:

а) организацией бесперебойного питания технических средств;

б) регулярным выполнением рекомендаций Министерства труда и социального  
развития РФ, изложенных в Постановлении от 23 июля 1998 г. «Об утверждении  
межотраслевых типовых норм времени на работы по сервисному обслуживанию  
ПЭВМ и оргтехники и сопровождению программных средств»;

в) регулярным выполнением требований ГОСТ 51188-98. Защита информации.  
Испытания программных средств на наличие компьютерных вирусов;

г) необходимым уровнем квалификации сотрудников профильных подразделений.

### Время восстановления после отказа

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем (не крахом) операционной системы, не должно превышать времени, необходимого на перезагрузку операционной системы и запуск командной строки, при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

### Отказы из-за некорректных действий оператора

Отказы программы возможны вследствие некорректных действий оператора (пользователя) при взаимодействии с операционной системой. Во избежание возникновения отказов программы по указанной выше причине следует обеспечить работу конечного пользователя без предоставления ему административных привилегий.

## **Условия эксплуатации**

### Климатические условия эксплуатации

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации.

### Требования к видам обслуживания

См. Требования к обеспечению надежного (устойчивого) функционирования программы.

### Требования к численности и квалификации персонала

Минимальное количество персонала, требуемого для работы команды, должно составлять не менее 1 штатной единицы - конечный пользователь программы - оператор.

Конечный пользователь программы (оператор) должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы.

## **Требования к составу и параметрам технических средств**

В состав технических средств должен входить персональный компьютер. В случае работы системы в сети все компьютеры должны быть подобны. Также необходимы кабеля для создания сети, сетевые карты на каждом компьютере и маршрутизатор. При предоставлении возможности поступления информации через сеть Интернет, один из компьютеров в сети, не являющийся сервером, должен иметь модем.

## **Требования к информационной и программной совместимости**

### Требования к информационным структурам и методам решения

Требования к информационным структурам (файлов) на входе и выходе, а также к методам решения не предъявляются.

### Требования к исходным кодам и языкам программирования

Требования к исходным кодам и языкам программирования не предъявляются.

### Требования к программным средствам, используемым программой

Системные программные средства, используемые программой, должны быть представлены локализованной версией операционной системы Windows или \*nix.

### Требования к защите информации и программ

Требования к защите информации и программ не предъявляются.

## **Специальные требования**

Специальные требования к команде не предъявляются.

или

Команда должна обеспечивать взаимодействие с пользователем (оператором) посредством командной строки.

# Требования к программной документации

## **Предварительный состав программной документации**

Состав программной документации должен включать в себя:

1) техническое задание;

2) спецификация;

3) текст команды;

4) описание команды;

5) методики испытаний;

6) пояснительная записка;

7) ведомость эксплуатационных документов;

8) формуляр;

9) описание применения;

10) руководство оператора;

## **Специальные требования к программной документации**

Специальные требования к программной документации не предъявляются.

# Технико-экономические показатели

## **Ориентировочная экономическая эффективность**

Ориентировочная экономическая эффективность не рассчитываются.

## **Предполагаемая годовая потребность**

Предполагаемое число использования команды в год – круглосуточное выполнение команды на одном рабочем месте.

# Стадии и этапы разработки

## **Стадии разработки**

Сроки и стадии выполнения этапов работ определяются отдельным графиком со сроками предъявления на испытания.

# Порядок контроля и приемки

## **Виды испытаний**

Приемо-сдаточные испытания команды должны проводиться согласно разработанной и согласованной «Программы и методики испытаний».

Ход проведения приемо-сдаточных испытаний документируется в Протоколе проведения испытаний.

## **Общие требования к приемке работы**

После проведения испытаний в полном объеме, на основании «Протокола испытаний» утверждают «Свидетельство о приемке» и производят запись в программном документе «Формуляр».

# уточнение тз

9.1. Данное ТЗ может быть изменено (дополнено) по согласованию сторон в установленном порядке.

|  |
| --- |
| Руководитель разработки Зам. Начальника |
| И.И. Иванов П.П. Петров |
| \_\_\_\_\_\_\_\_\_\_\_Хххххххх Х.Х. \_\_\_\_\_\_\_\_\_\_Хххххххх Х.Х. |
| “\_\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_200\_\_ “\_\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_200\_\_ |