|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА − Российский технологический университет»**  **РТУ МИРЭА** |

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ**

по дисциплине «Тестирование и верификация программного обеспечения»

**Практическая работа №1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студенты группы | *ИКБО-50-23, Иващенко А.В.*  *Галкин М. В* | (подпись) | |  |
| Преподаватель | *Ильичев Г. П.* | (подпись) | |  |
| Отчет представлен | «\_\_\_»\_\_\_\_\_\_\_\_202\_\_г. | |  | |

Москва 2025 г.

# Введение

Поступил заказ для создания эмулятора Shell для языка оболочки UNIX-подобной операционной системы. Продукт должен реализовывать команды, содержать уже готовую конфигурацию и работать в режиме графического редактора.

# Основания для разработки

Данный заказ поступил для проверки возможностей программистов в сфере разработки программ, на примере Эмулятора Shell. Проект должен продемонстрировать способности программистов и их компетентность в реализации сложных программ с использованием языка программирования Python. Для разработки предоставлены:

* Документация для работы в Python;
* Методические материалы для реализации заказа;
* ГОСТ 34.602-2020 для составления технического задания.

# Назначение разработки

Целью разработки является создание простого Эмулятора Shell с конфигурацией в соответствии с заказом, а также определенными стандартными функциями для оболочки ОС. В итоге ожидается готовый безотказный продукт, способный продемонстрировать ожидаемый функционал.

# Требования к программе

# 4.1 Функциональные требования

Эмулятор должен работать в режиме GUI и принимать образ виртуальной файловой системы в виде файла формата zip.

Должен быть создан конфигурационный файл в формате csv который будет содержать:

* Имя пользователя для показа в приглашении к вводу;
* Путь к архиву виртуальной файловой системы;
* Путь к старому архиву.

Эмулятор должен поддерживать команды:

* ls – отображение файлов и директорий внутри заданной;
* cd – перемещение по директориям;
* exit – выход из эмулятора;
* find – нахождение по шаблону файла/директории;
* chown – изменение владельца файла;
* tail – вывод последних строк текстового файла.

Запуск должен осуществляться через консоль (python main.py config.csv).

# 4.2 Требования к надежности

Эмулятор должен обрабатывать только команды, описанные в функциональных требованиях. Попытки ввести иные команды или случайный набор символов вне и внутри готовых должны сопровождаться сообщением об ошибке цвета, отличного от стандартного темного.

# 4.3 Условия эксплуатации

Операционная система Windows 10 или выше;

Минимальные требования персонального компьютера: современный процессор (любой), 256 Мб оперативной памяти, 1 Мб свободного места на диске.

# 4.4 Требования к совместимости

Программа требует предустановки языка программирования Python, так как реализована на данном языке (.py).

# Требования к интерфейсу

Эмулятор должен имитировать оболочку UNIX-подобной ОС, где основными цветами являются: розовый (фон терминала), темно-синий/близкий к черному (текст), красный (сообщения об ошибках) и белый/custom (внешняя составляющая терминала).

Текст должен быть разборчивым, достаточно большим. Формат вывода консоли: <пользователь>:~<текущая директория> “команда”. (С новой строки – результат выполнения команды, если предусмотрен).

# Критерии приемки

Продукт считается соответствующим настоящему ТЗ и готовым к приемке, если:

* Успешно пройдены все тест-кейсы, составленные на основе функциональных требований, указанных в разделе 4.1;
* Интерфейс программы соответствует требованиям раздела 5;
* Программа запускается и функционирует на целевой операционной системе, указанной в п. 4.3.

# Требования к документации

В состав поставки программного продукта должна входить следующая документация:

* Краткое руководство пользователя (в формате README.md).

# Порядок контроля и приемки

Тестирование программы будет проводиться методом "черного ящика" на основе требований, изложенных в настоящем техническом задании. Приемочные испытания включают в себя:

* Функциональное тестирование всех элементов интерфейса;
* Тестирование корректности вывода при выполнении команд;
* Тестирование удобства использования.

# Этапы и сроки разработки

1. Проектирование архитектуры эмулятора – 1 дня;
2. Разработка кода программы и конфигурации – 4 дня;
3. Написание сопроводительной документации – 1 день;
4. Внутреннее тестирование – 1 день.

Общий срок разработки – 7 дней.