



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИИТ)
Кафедра математического обеспечения и стандартизации
информационных технологий (МОСИТ)

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ
по дисциплине «Тестирование и верификация программного обеспечения»
Команда «Три кота»

Практическое занятие № 1

Студенты группы *ИКБО-50-23, Ерхова В.А., Павлов Н. С.,
Хохряков А. Ю.*

(подпись)

Преподаватель *Ильичев Г.П*

(подпись)

Отчет представлен «19» сентября 2025 г.

Москва 2025 г.

Введение

Программа "Конвертер величин" представляет собой desktop-приложение для быстрого и удобного преобразования различных физических величин между единицами измерения. Приложение предназначено для использования в образовательных целях, в быту, а также в профессиональной деятельности, где требуется оперативный пересчет значений между различными системами измерений.

Области применения:

- Учебный процесс в школах и вузах
- Инженерные расчеты
- Научно-исследовательская деятельность
- Повседневное использование для бытовых расчетов

Основания для разработки

Разработка программы осуществляется в рамках учебного проекта для демонстрации практических навыков программирования на Python с использованием библиотеки tkinter для создания графического интерфейса.

Исходные документы:

- Задание на практическую работу по программированию
- Рекомендации по разработке пользовательских интерфейсов

Назначение разработки

Программа предназначена для автоматизации процесса конвертации физических величин, что позволит:

- Сократить время расчета по сравнению с ручными вычислениями
- Исключить ошибки при пересчете значений между системами измерений
- Обеспечить наглядное представление результатов преобразования
- Предоставить удобный интерфейс для работы с различными категориями величин

Требования к программе

Функциональные требования

- Конвертация величин в категориях: длина, масса, температура, время, объем, площадь
- Возможность выбора конкретных единиц измерения внутри каждой категории
- Функция обмена выбранных единиц измерения
- Отображение результатов преобразования с точностью до 8 знаков после запятой
- Обработка ошибок ввода и некорректных данных
- Валидация вводимых значений (проверка на отрицательные значения для определенных категорий)
- Очистка всех полей ввода по запросу пользователя

Требования к надежности

- Обработка исключительных ситуаций без аварийного завершения программы
- Корректное восстановление после ошибок ввода
- Защита от некорректных операций (деление на ноль, нечисловые значения)
- Стабильная работа при непрерывном использовании

Условия эксплуатации

- Операционная система: Windows 7/8/10/11, Linux, MacOS
- Минимальные требования: 512 МБ ОЗУ, 50 МБ свободного места на диске
- Обязательное наличие интерпретатора Python версии 3.6 и выше
- Наличие библиотеки tkinter в составе Python

Требования к совместимости

- Совместимость с операционными системами семейства Windows, Linux, MacOS
- Поддержка кодировки UTF-8 для корректного отображения символов
- Независимость от дополнительного программного обеспечения (кроме Python)

Требования к интерфейсу

Основные элементы интерфейса

- Выпадающий список для выбора категории величин
- Поле ввода исходного значения с возможностью использования точки или запятой как разделителя
- Два выпадающих списка для выбора единиц измерения (исходная и целевая)
- Кнопка обмена единицами измерения (\leftrightarrow)
- Кнопка "Конвертировать" зеленого цвета с белым текстом
- Поле вывода результата с зеленым текстом на светлом фоне
- Область дополнительной информации о результате преобразования
- Кнопка "Очистить все" красного цвета с белым текстом

Особенности интерфейса

- Все элементы управления должны быть подписаны на русском языке
- Размер окна фиксированный (500x650 пикселей)
- Шрифты: Arial различных размеров (9-16) и начертаний
- Цветовая схема: сочетание зеленых, серых и белых тонов
- Кнопка конвертации должна быть заметным акцентным элементом
- Поле результата должно визуально отличаться от поля ввода

Критерии приемки

- Успешное выполнение 100% тестов
- Отсутствие аварийных завершений программы при любых сценариях использования
- Соответствие времени отклика интерфейса требованиям (не более 0.5 сек)
- Полное соответствие функциональности заявленным требованиям

Требования к документации

Обязательная документация

- Запрос пользователя с описанием требуемой системы
- Техническое описание программы
- Техническое задание
- Исходный код с соответствующими комментариями

Порядок контроля и приемки

Методы тестирования

- Тестирование методом чёрного ящика
- Функциональное тестирование всех элементов интерфейса
- Тестирование корректности преобразования величин
- Тестирование обработки ошибок и исключительных ситуаций
- Тестирование пользовательского интерфейса на удобство использования

Приемочные испытания

- Проверка на соответствие техническому заданию
- Тестирование на различных операционных системах

- Проверка работы при различных разрешениях экрана
- Оценка удобства интерфейса целевой аудиторией

Этапы и сроки разработки

1. Проектирование архитектуры приложения - 1 день
2. Разработка графического интерфейса - 2 дня
3. Реализация логики преобразования величин - 3 дня
4. Реализация обработки ошибок и исключений - 1 день
5. Тестирование и отладка программы - 2 дня
6. Написание документации - 1 день

Общий срок разработки: 10 рабочих дней

Дополнительная документация на программный продукт

"Конвертер величин"

1. Руководство пользователя

Обзор приложения

"Конвертер величин" — это графическое приложение для конвертации различных физических единиц измерения. Оно поддерживает категории: Длина, Масса, Температура, Время, Объем и Площадь. Приложение позволяет вводить значение, выбирать исходную и целевую единицы, выполнять конвертацию и отображать результат.

Установка и запуск

- **Требования:** Python 3.x с установленной библиотекой Tkinter (обычно входит в стандартную поставку Python).
- **Запуск:** Сохраните код в файл `main.py` и выполните команду `python main.py` в терминале. Приложение откроется в отдельном окне.

Интерфейс пользователя

- **Категория:** Выберите категорию из выпадающего списка (например, "Длина").
- **Исходное значение:** Введите числовое значение в поле ввода. Поддерживаются целые и дробные числа (используйте точку или запятую как разделитель).
- **Из:** Выберите исходную единицу измерения.
- **В:** Выберите целевую единицу измерения.
- **Кнопка "↔":** Обменяет исходную и целевую единицы.
- **Кнопка "Конвертировать":** Выполнит расчет и отобразит результат.
- **Раздел результата:** Показывает конвертированное значение и дополнительную информацию (например, "10 метры = 1000 сантиметры").
- **Кнопка "Очистить все":** Сбросит все поля.

Примеры использования

1. **Конвертация длины:** Выберите "Длина", введите "1" в поле значения,

"километры" в "Из", "метры" в "В". Нажмите "Конвертировать".
Результат: "1000".

2. **Конвертация температуры:** Выберите "Температура", введите "0", "°C" в "Из", "°F" в "В". Результат: "32".
3. **Обмен единиц:** Нажмите "↔" для быстрого переключения между "Из" и "В".
4. **Обработка ошибок:** Если значение не число, приложение покажет сообщение "Введите корректное число". Для отрицательных значений в "Длина", "Масса", "Температура", "Время", "Объем" и "Площадь" — "Значение не может быть отрицательным".

Советы

- Используйте клавишу Enter для конвертации после ввода значения.
- Результат форматируется для удобства: с разным количеством знаков после запятой в зависимости от величины.
- Для температуры учитывается абсолютный ноль (в Кельвинах результат не может быть отрицательным).

2. Описание архитектуры системы

Общая структура

Приложение построено на основе объектно-ориентированного подхода. Основной класс UnitConverter инкапсулирует всю логику: инициализацию данных, настройку UI и обработку конвертации. Главная функция main() создает корневое окно Tkinter и экземпляр класса.

- **Модули и библиотеки:**
 - tkinter и ttk для графического интерфейса.
 - messagebox для вывода ошибок.
- **Данные:**
 - Словарь units хранит списки единиц по категориям.
 - Словарь conversion_factors содержит коэффициенты конвертации (относительно базовой единицы, первой в списке).
- **Методы:**

- `__init__`: Инициализирует окно, данные и UI.
- `setup_ui`: Настраивает элементы интерфейса (фреймы, метки, комбобоксы, кнопки).
- `update_units`: Обновляет списки единиц при смене категории.
- `swar_units`: Обменивает единицы "Из" и "В".
- `clear_fields`: Очищает поля ввода и результата.
- `convert_temperature`: Специальная функция для температуры (конвертация через Цельсий).
- `convert`: Основная логика конвертации, включая проверки и форматирование.

Архитектурные принципы

- **Модульность:** UI отделен от логики конвертации.
- **Событийно-ориентированный подход:** Используются привязки событий (e.g., `<<ComboboxSelected>>`, `<Return>`).
- **Обработка исключений:** Ловит `ValueError` (некорректное число), `ZeroDivisionError` и другие.
- **Базовая единица:** Для большинства категорий конвертация идет через базовую единицу (e.g., метры для длины).
- **Особенности:** Температура обрабатывается отдельно из-за нелинейных формул. Для объема реализована специальная логика (см. ошибки ниже).

3. Схемы и диаграммы

Диаграмма классов (UML-в текстовом формате)

UnitConverter
Атрибуты: - root: Tk - units: dict {категория: список единиц} - conversion_factors: dict {категория: коэффициенты} - category_var: StringVar - from_var: StringVar - to_var: StringVar - input_var: StringVar - result_var: StringVar - result_info_var: StringVar
Методы: + <code>__init__(root)</code> : Инициализация окна и UI + <code>setup_ui()</code> : Создание интерфейса

- + `update_units(event)`: Обновление единиц
- + `swap_units()`: Обмен единиц "Из" и "В"
- + `clear_fields()`: Очистка полей ввода/вывода
- + `convert_temperature(value, from, to)`: Конвертация температуры
- + `convert()`: Выполнение конвертации

Схема интерфейса (ASCII-арт)

Рисунок 1 - Интерфейс

Диаграмма потока данных для конвертации

1. Пользователь выбирает категорию → `update_units` обновляет комбобоксы.
2. Ввод значения + единицы → Нажатие "Конвертировать" → `convert`:
 - Проверки (значение, категория, единицы).
 - Если Температура: `convert_temperature`.
 - Иначе: $\text{value} * \text{factor_from} / \text{factor_to}$.

- Форматирование → Отображение в `result_var` и `result_info_var`.

4. Описание внесённых ошибок в собственное ПО

В ПО преднамеренно добавлены ошибки для демонстрации уязвимостей и тестирования. Ниже описаны их типы (логические, синтаксические и т.д.), причины и способы обнаружения. Эти ошибки не влияют на запуск, но проявляются при определенных сценариях.

1. Отсутствие проверки на отрицательные значения для категорий Время, Объем, Площадь

- **Тип:** Логическая ошибка (неполная валидация ввода).
- **Описание:** Проверка на отрицательные значения реализована только для Длины и Массы, но не для других категорий, где отрицательные значения также бессмысленны (отрицательное время).
- **Способ обнаружения:** Введите отрицательное значение (-1) в категориях Время, Объем или Площадь и выполните конвертацию. Приложение вернет отрицательный результат без ошибки, что логически неверно.

2. Ошибка знака при конвертации из Кельвинов в Цельсий

- **Тип:** Логическая ошибка (математическая).
- **Описание:** В функции `convert_temperature` при конвертации из К в С используется плюс вместо минуса: `celsius = value + 273.15` (должно быть `-`).
- **Способ обнаружения:** Конвертируйте 273.15 К в °С — результат будет 546.3 вместо 0. Обнаруживается при тестировании стандартных значений (абсолютный ноль).

3. Неверная константа для недель в категории Время

- **Тип:** Логическая ошибка (фактическая неточность в данных).
- **Описание:** В `conversion_factors['Время']` для 'недели' указано 691200 секунд (8 дней), вместо 604800 (7 дней).
- **Способ обнаружения:** Конвертируйте 1 неделю в секунды —

результат 691200 вместо 604800. Обнаруживается при сравнении с известными значениями или ручном расчете.

4. Отрицательная константа для сантиметров в категории Длина

- **Тип:** Логическая ошибка (математическая).
- **Описание:** В `conversion_factors['Длина']` для 'сантиметры' указано -0.01 вместо 0.01.
- **Способ обнаружения:** Конвертируйте положительное значение из/в сантиметры (e.g., 100 см в метры) — результат будет отрицательным. Обнаруживается при любой конвертации с сантиметрами.

5. Специальная обработка для категории Объем

- **Тип:** Интерфейсная/логическая ошибка (условное поведение).
- **Описание:** В методе `convert` для Объемов вместо расчета выводится "Я не знаю" в результате и "А если бы и знал, то все равно не сказал бы" в `info`.
- **Способ обнаружения:** Выберите "Объем" и выполните любую конвертацию — приложение откажется от расчета. Обнаруживается только в этой категории.

6. Неверная константа для граммов в категории Масса

- **Тип:** Логическая ошибка (фактическая неточность в данных).
- **Описание:** В `conversion_factors['Масса']` для 'граммы' указано 0.01 вместо 0.001 (относительно килограмма).
- **Способ обнаружения:** Конвертируйте 1000 граммов в килограммы — результат 10 вместо 1. Обнаруживается при тестировании с круглыми числами.

Техническое задание программного продукта другой команды.

1. Введение

Mente Nova — это настольное приложение, разработанное для помощи студентам в эффективной организации учебных материалов. Оно предоставляет удобный интерфейс для структурирования предметов и загрузки различных типов файлов, включая лекции, практические задания, лабораторные работы, отчеты, фотографии и многое другое. Приложение позволяет организовывать данные по учебным семестрам и предлагает функции управления файлами, такие как переименование, удаление и открытие, с встроенной поддержкой чтения PDF-файлов.

Цели: структуризация материалов в одном приложении с возможностью чтения файлов без сторонних приложений.

Сфера: основной сферой использования является учеба.

2. Основания для разработки

- Проблема: разрозненное хранение материалов (мессенджеры, облака, локальные папки), отсутствие единого интерфейса и структуры.
- Необходимость: единое приложение с иерархией семестр → предмет → тип материала, с быстрым доступом и базовым просмотром.
- Технологические предпосылки: кроссплатформенность Java/JavaFX, объектное хранилище MinIO, библиотеки для чтения документов.

3. Назначение разработки

- Пользователи: студенты, преподаватели, учебные группы.
- Сценарии: импорт/загрузка материалов; организация по семестрам и предметам; быстрый поиск; базовый просмотр PDF; управление файлами (переименование/удаление/перемещение).
- Ожидаемый эффект: повышение организованности, снижение времени на поиск материалов, единое место хранения.

4. Требования к программе

Функциональные:

- Структура данных: семестры → предметы → категории материалов →

файлы.

- Операции с хранилищем:
загрузка/скачивание/перемещение/переименование/удаление файлов и папок; создание/удаление разделов.

- Просмотр: встроенное чтение PDF; отображение метаданных (размер, дата изменения).

- Поиск и фильтрация: по названию, типу, предмету, семестру.

- Импорт/экспорт: пакетная загрузка папок; скачивание папок.

Нефункциональные:

- Кроссплатформенность: Windows/macOS/Linux (JDK 23).

- Производительность: потоковые операции без “заморозки” UI для файлов ≥ 100 МБ.

- Надёжность: устойчивость к сетевым сбоям MinIO; возобновляемые операции (по возможности).

- Безопасность: хранение конфигурации доступа вне кода; маскирование секретов.

- Логирование: операции и ошибки через TinyLog.

Технологии:

- Java 23, Spring Boot, JavaFX (OpenJFX), MinIO (io.minio), Apache PDFBox, Apache POI/ODF Toolkit (по необходимости), Maven.

5. Требования к интерфейсу

- Главный экран: дерево/список семестров и предметов, панель файлов, строка поиска, панель действий (загрузить / скачать / переименовать / удалить / открыть).

- Темы: светлая и тёмная.

- Юзабилити: крупные интерактивные элементы, горячие клавиши для частых действий, подтверждения опасных операций.

- Адаптивность: корректная компоновка при изменении размеров окна; сохранение пользовательских предпочтений (последний семестр/предмет).

- Просмотрщик: встроенное окно просмотра PDF с базовой навигацией

(страницы, масштаб).

6. Критерии приёмки

- Функциональность: все заявленные операции с файлами и папками выполняются корректно; структура семестр → предмет работает; PDF открывается во встроенном просмотрщике.
- Стабильность: приложение обрабатывает типовые сценарии без критических ошибок; корректная обработка недоступности MinIO (понятные сообщения).
- Производительность: UI не блокируется при загрузке/скачивании больших файлов; операции завершаются в разумные сроки.
- Удобство: понятная навигация, консистентные статусы и подтверждения; тёмная/светлая тема переключаются без артефактов.
- Сборка и поставка: исполняемый JAR собирается; приложение запускается на поддерживаемых ОС.

7. Требования к документации

- Пользовательская: краткое руководство (установка, подключение к MinIO, базовые операции, горячие клавиши).
- Техническая: описание архитектуры, модулей, зависимостей, профилей конфигурации; инструкции сборки и запуска.
- Администрирование: параметры `application.properties/yml`, переменные окружения, примеры конфигурации MinIO.

8. Порядок контроля и приёмки

- Тестирование: функциональные тест-кейсы по ключевым сценариям (CRUD файлов/папок, просмотр PDF, поиск); проверка отказоустойчивости (падение сети/MinIO).
- Демо-сессия: показ выполнения критериев приемки на эталонном окружении.
- Отчётность: чек-лист прохождения тестов, логи без критических ошибок; фиксация версий артефактов.
- Принятие: подписание протокола приемки при успешном прохождении

всех пунктов.

9. Этапы и сроки разработки

- Инициация (1 неделя): уточнение требований, проектирование структуры данных и UX-скетчи.
- Архитектура и настройка (1 неделя): каркас Spring Boot/JavaFX, конфигурация Maven, интеграция MinIO.
- Базовый функционал (2 недели): структура семестров/предметов, CRUD файлов/папок, метаданные, логирование.
- Просмотр PDF и поиск (1 неделя): встроенный ридер PDF, поиск/фильтрация.
- UI/UX и темы (1 неделя): адаптивность, светлая/тёмная тема, горячие клавиши.
- Тестирование и стабилизация (1 неделя): исправления, документация, релизный пакет.
- Итог: ориентировочно 6–7 недель, возможна корректировка по результатам анализа рисков и объёма.

Комментарий к ТЗ:

1. Следует дополнить введение кратким описанием целевой аудитории (например, студенты бакалавриата или магистратуры) и контекста использования (например, в университете или дома).
2. Рекомендуются добавить обоснование выбора технологий (например, почему JavaFX предпочтительнее Qt или Electron) и указать версии библиотек (например, MinIO 8.x).
3. Следует дополнить раздел описанием совместного доступа и указать, поддерживается ли многопользовательский режим.
4. Добавить конкретные метрики (время загрузки, поиска). Указать, как обрабатываются большие файлы (>1 ГБ). Указать версии технологий для избежания конфликтов.
5. Добавить конкретные размеры кнопок и минимальное разрешение.
6. Указать инструменты тестирования.

7. Добавить автоматизированные тесты (JUnit) и указать, кто подписывает протокол (преподаватель, заказчик).

Результаты тестирования программного продукта

1. Ошибка 1

Идентификатор: TC-001



Рисунок 2 – Визуализация ошибки

Название: Проверка поведения кнопки при наведении и нажатии

Описание: Убедиться, что при наведении на кнопку она остается видимой и при нажатии выполняется переход в соответствующую папку.

Тип ошибки: Функциональная

Предварительные условия:

Программа запущена.

Открыт интерфейс файлового менеджера.

Папка с кнопкой для перехода отображается.

Шаги выполнения:

Навести курсор на кнопку перехода в папку.

Нажать на кнопку.

Ожидаемый результат:

При наведении кнопка остается видимой.

При нажатии происходит переход в соответствующую папку, отображается её содержимое.

Фактический результат:

При наведении кнопка пропадает.

При нажатии переход в папку не происходит.

Статус: Failed

2. Ошибка 2

Идентификатор: TC-002

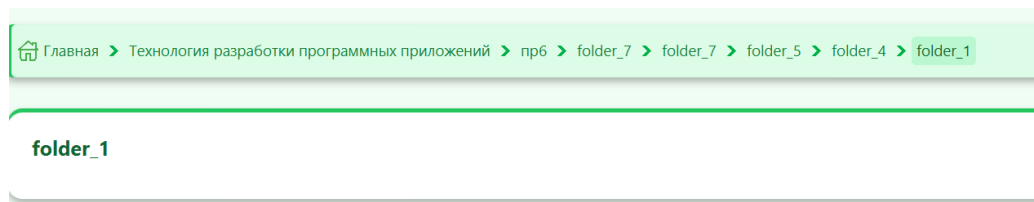


Рисунок 3 – Визуализация ошибки

Название: Проверка работы кнопки «Назад»

Описание: Убедиться, что при нажатии кнопки «Назад» пользователь возвращается в предыдущую папку без появления новых папок.

Тип ошибки: Функциональная

Предварительные условия:

Программа запущена.

Пользователь находится в папке, доступной через навигацию.

Кнопка «Назад» активна.

Шаги выполнения:

Нажать на кнопку «Назад».

Ожидаемый результат:

Пользователь возвращается в предыдущую папку.

Список папок остается неизменным, новые папки не появляются.

Фактический результат:

При нажатии кнопки «Назад» отображаются новые папки, которых ранее не было.

Статус: Failed

3. Ошибка 3

Идентификатор: ТС-003

Название: Проверка удаления папки

Описание: Убедиться, что при нажатии на кнопку «Удалить» выбранная папка удаляется из файлового менеджера.

Тип ошибки: Функциональная

Предварительные условия:

Программа запущена.

В файловом менеджере есть хотя бы одна папка.

Кнопка «Удалить» активна.

Шаги выполнения:

Выбрать папку для удаления.

Нажать кнопку «Удалить».

Ожидаемый результат:

Выбранная папка удаляется.

В интерфейсе отображается обновленный список без удаленной папки.

Фактический результат:

При нажатии на кнопку «Удалить» папка не удаляется и остается в списке.

Статус: Failed

4. Ошибка 4

Идентификатор: ТС-004

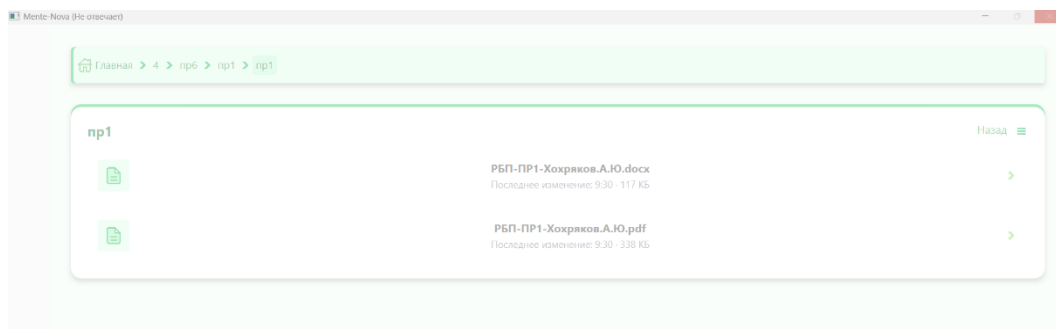


Рисунок 4 – Визуализация ошибки

Название: Проверка загрузки больших папок

Описание: Убедиться, что программа корректно обрабатывает загрузку больших папок и остается отзывчивой.

Тип ошибки: Производительность

Предварительные условия:

Программа запущена.

Доступна большая папка (содержащая множество файлов или большой объем данных).

Шаги выполнения:

Попытаться открыть большую папку.

Перезапустить программу и повторить попытку.

Ожидаемый результат:

Папка загружается, интерфейс остается отзывчивым.

После перезапуска программа работает корректно.

Фактический результат:

Программа перестает отвечать при загрузке большой папки.

После перезапуска программа продолжает не отвечать.

Статус: Failed

5. Ошибка 5

Идентификатор: ТС-005

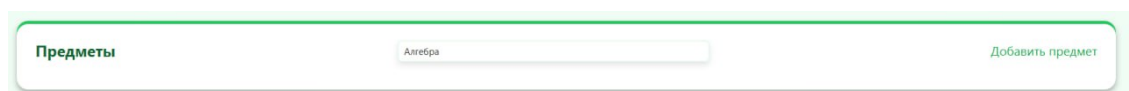


Рисунок 5 – Визуализация ошибки

Название: Проверка добавления предметов и выхода из семестра

Описание: Убедиться, что пользователь может добавлять предметы в семестр и выходить из интерфейса семестра.

Тип ошибки: Функциональная

Предварительные условия:

Программа запущена.

Открыт семестр, в котором нет предметов.

Шаги выполнения:

Попытаться добавить предмет в семестр.

Попытаться выйти из интерфейса семестра.

Ожидаемый результат:

Предмет успешно добавляется в семестр.

Пользователь может выйти из интерфейса семестра.

Фактический результат:

Предметы не добавляются.

Невозможно выйти из интерфейса семестра.

Статус: Failed

6. Ошибка 6

Идентификатор: ТС-006

Название: Проверка наличия функции редактирования файлов и папок

Описание: Убедиться, что пользователь может редактировать файлы и папки в программе, как указано в документации.

Тип ошибки: Функциональная

Предварительные условия:

Программа запущена.

В файловом менеджере есть файлы и папки.

Шаги выполнения:

Выбрать файл или папку.

Попытаться выполнить редактирование (например, переименовать или изменить содержимое).

Ожидаемый результат:

Пользователь может редактировать файлы и папки (например, изменять названия или содержимое).

Изменения сохраняются и отображаются в интерфейсе.

Фактический результат:

Функция редактирования файлов и папок отсутствует.

Статус: Failed

7. Ошибка 7

Идентификатор: ТС-007

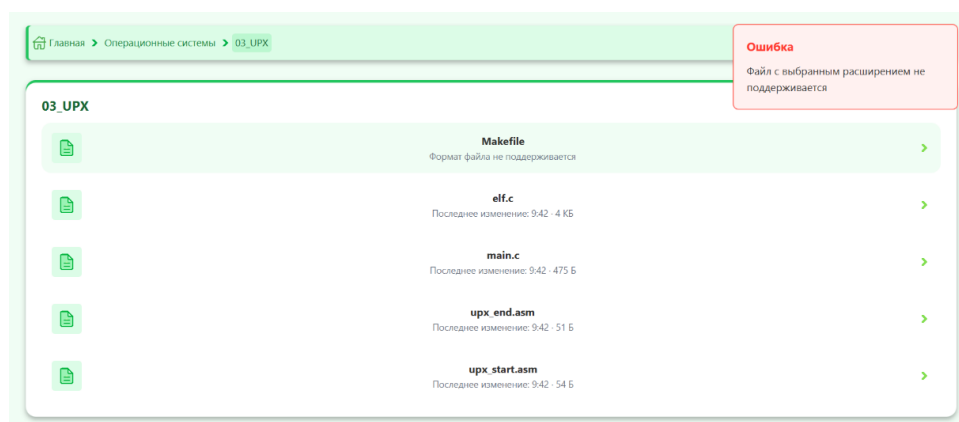


Рисунок 6 – Визуализация ошибки

Название: Проверка поддержки MakeFile

Описание: Убедиться, что программа корректно открывает MakeFile, как и другие типы файлов, в сторонних приложениях.

Тип ошибки: Функциональная

Предварительные условия:

Программа запущена.

В файловом менеджере присутствует MakeFile.

Стороннее приложение для открытия MakeFile установлено.

Шаги выполнения:

Выбрать MakeFile.

Попытаться открыть его.

Ожидаемый результат:

MakeFile открывается в соответствующем стороннем приложении, как и другие типы файлов.

Фактический результат:

MakeFile не поддерживается и не открывается, хотя другие файлы открываются в сторонних приложениях без проблем.

Статус: Failed

8. Ошибка 8

Идентификатор: TC-008

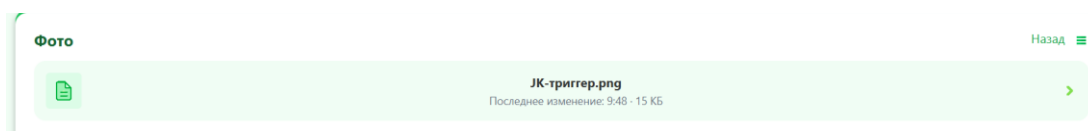


Рисунок 7 – Визуализация ошибки

Название: Проверка открытия PNG-файлов

Описание: Убедиться, что программа корректно открывает PNG-файлы, как заявлено в поддерживаемых форматах.

Тип ошибки: Функциональная

Предварительные условия:

Программа запущена.

В файловом менеджере присутствует PNG-файл.

Шаги выполнения:

Выбрать PNG-файл.

Попытаться открыть его.

Ожидаемый результат:

PNG-файл открывается в программе или в стороннем приложении.

Фактический результат:

PNG-файл не открывается, несмотря на заявленную поддержку.

Статус: Failed

9. Ошибка 9

Идентификатор: ТС-009

Название: Проверка наличия темной темы

Описание: Убедиться, что в программе доступна темная тема для интерфейса.

Тип ошибки: Интерфейс

Предварительные условия:

Программа запущена.

Открыт раздел настроек интерфейса.

Шаги выполнения:

Попытаться найти и включить темную тему в настройках.

Ожидаемый результат:

В настройках доступна опция переключения на темную тему.

После активации интерфейс переключается на темную тему.

Фактический результат:

Темная тема отсутствует в настройках и не может быть включена.

Статус: Failed

10. Ошибка 10

Идентификатор: ТС-010

Название: Проверка поддержки горячих клавиш

Описание: Убедиться, что в программе реализованы горячие клавиши для основных функций.

Тип ошибки: Интерфейс

Предварительные условия:

Программа запущена.

Пользователь находится в интерфейсе файлового менеджера или другом функциональном разделе.

Шаги выполнения:

Попытаться использовать стандартные горячие клавиши (например, Ctrl+C, Ctrl+V, Delete).

Ожидаемый результат:

Горячие клавиши выполняют соответствующие действия (копирование, вставка, удаление и т.д.).

Фактический результат:

Горячие клавиши отсутствуют и не работают.

Статус: Failed

11. Ошибка 11

Идентификатор: TC-011

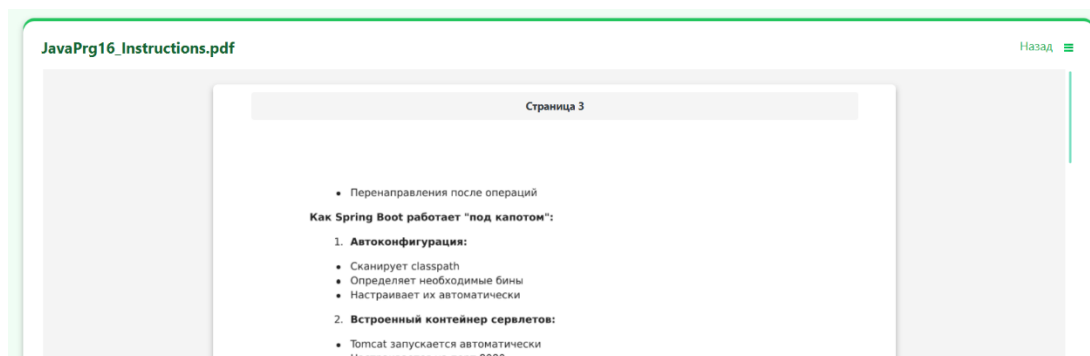


Рисунок 10 – Визуализация ошибки

Название: Проверка отображения PDF-файлов

Описание: Убедиться, что PDF-файлы отображаются корректно, начиная с первой страницы.

Тип ошибки: Функциональная

Предварительные условия:

Программа запущена.

В файловом менеджере присутствует PDF-файл.

Шаги выполнения:

Открыть PDF-файл.

Проверить отображение страниц.

Ожидаемый результат:

PDF-файл открывается, начиная с первой страницы.

Все страницы отображаются корректно.

Фактический результат:

PDF-файл открывается с третьей страницы, первые две страницы не отображаются.

Статус: Failed

12. Ошибка 12

Идентификатор: ТС-012

Название: Проверка наличия строки поиска

Описание: Убедиться, что в программе доступна строка поиска для быстрого нахождения файлов и папок.

Тип ошибки: Интерфейс

Предварительные условия:

Программа запущена.

Открыт интерфейс файлового менеджера.

Шаги выполнения:

Попытаться найти строку поиска в интерфейсе.

Ввести запрос для поиска файла или папки.

Ожидаемый результат:

В интерфейсе присутствует строка поиска.

При вводе запроса отображаются соответствующие файлы или папки.

Фактический результат:

Строка поиска отсутствует в интерфейсе.

Статус: Failed

13. Ошибка 13

Идентификатор: ТС-013

Название: Проверка стабильности программы

Описание: Убедиться, что программа работает стабильно и не зависает при выполнении операций.

Тип ошибки: Производительность

Предварительные условия:

Программа запущена.

Пользователь выполняет стандартные операции (например, открытие папок, просмотр файлов).

Шаги выполнения:

Выполнить несколько операций в программе (например, открыть папку, просмотреть файл).

Продолжить использование программы в течение нескольких минут.

Ожидаемый результат:

Программа работает стабильно, без зависаний.

Все операции выполняются корректно.

Фактический результат:

Программа зависает во время выполнения операций.

Статус: Failed

Анализ документации

Mente Nova - Организатор учебных материалов

Содержание

- Описание
- Функции
- Скриншоты
- Установка
- Использование
- Технологический стек
- Конфигурация
- Troubleshooting
- Лицензия и вклад
- Дата обновления

Описание

Mente Nova — это кроссплатформенное настольное приложение на Java, предназначенное для студентов и преподавателей для эффективной организации учебных материалов. Оно позволяет структурировать данные по семестрам, предметам и категориям (лекции, практики, лабораторные работы, отчеты, фото и другие файлы). Поддерживаются операции с файлами: загрузка, переименование, удаление, открытие и встроенный просмотр PDF. Приложение использует объектное хранилище MinIO для хранения, обеспечивая удобный доступ без сторонних сервисов. Целевая аудитория: студенты вузов, преподаватели. Версия приложения: 1.0 (на сентябрь 2025).

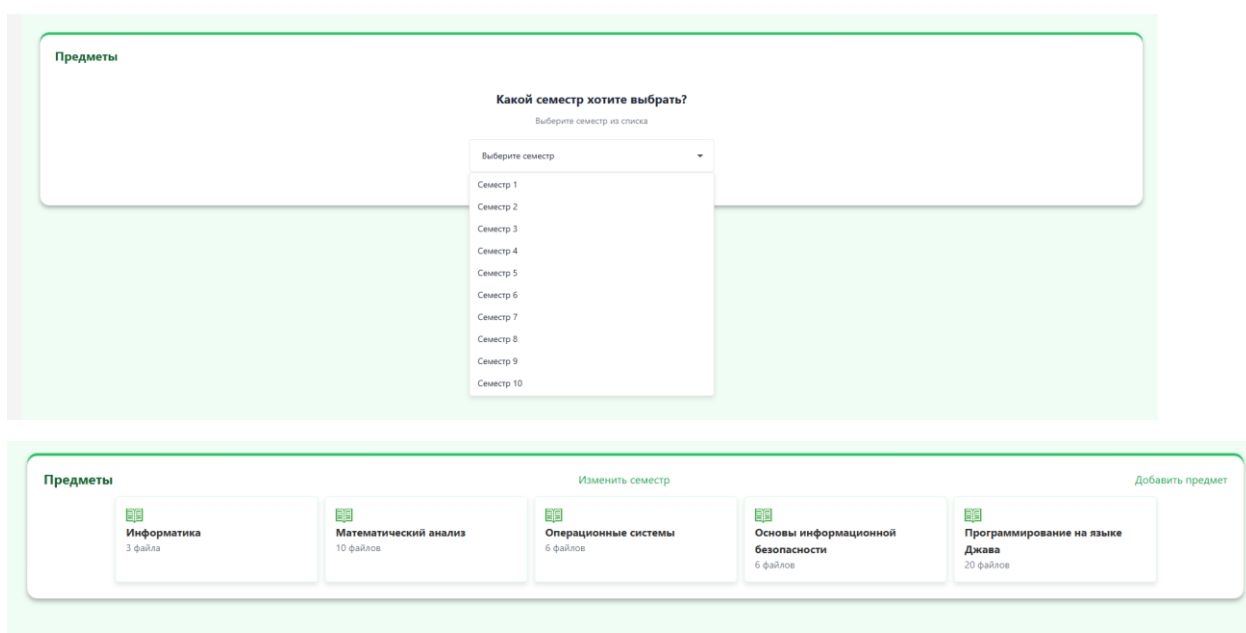
Функции

- **Управление предметами:** Легко добавляйте, редактируйте и удаляйте предметы.
- **Загрузка файлов:** Загружайте несколько файлов через диалоговое окно или перетаскиванием папок.
- **Управление файлами:** Переименовывайте, удаляйте и открывайте

файлы непосредственно в приложении.

- **Просмотр PDF:** Встроенный просмотрщик PDF для удобного доступа к документам.
- **Организация по семестрам:** Структурируйте данные по учебным семестрам.
- **Импорт/Экспорт:** Импортируйте существующие структуры папок или экспортируйте организованные данные для резервного копирования или переноса.

Скриншоты



Установка

Для пользователей

Чтобы установить Mente Nova, выполните следующие шаги:

1. **Предварительные требования:** Убедитесь, что на вашем компьютере установлена Java 11 или выше.
2. **Скачивание:** Загрузите последнюю версию из GitHub Releases.
3. **Распаковка:** Разархивируйте загруженный файл в нужное место.
4. **Запуск:** Запустите приложение, дважды щелкнув по исполняемому jar-файлу или используя предоставленный скрипт запуска.

Для разработчиков (сборка из исходного кода)

1. Клонировать репозиторий: `git clone`

https://github.com/YG5126/Mente_Nova.

2. Перейдите в директорию: `cd Mente_Nova` (исправлено на правильное имя репозитория).
3. Установите зависимости: Убедитесь в Maven 3.9+.
4. Соберите: `mvn clean package`.
5. Запустите: `java -jar target/mente-nova-1.0.jar`.

Использование

Начало работы

1. Запустите Mente Nova.
2. Создайте новый семестр, выбрав в начальном меню "Новый семестр", или выберите существующий.
3. В рамках семестра добавьте предметы, щелкнув правой кнопкой мыши и выбрав "Добавить предмет".
4. Загружайте файлы в предмет, выбрав "Загрузить файлы" или перетаскивая файлы/папки на предмет

Управление файлами

- Чтобы переименовать файл, щелкните по нему правой кнопкой мыши и выберите "Переименовать".
- Чтобы удалить файл, щелкните правой кнопкой мыши и выберите "Удалить".
- Чтобы сохранить файл на устройстве, щелкните правой кнопкой мыши и выберите "Сохранить файл".
- Чтобы открыть файл, щелкните по нему. PDF-файлы откроются во встроенном просмотрщике.

Организация по семестрам

- Управляйте семестрами через главное меню.
- Все файлы и предметы автоматически организуются в соответствующих семестрах.

Технологический стек

- **Фреймворк:** Spring Boot 3.3.x

- **Инструмент сборки:** Maven 3.9.x
- **Интерфейс:** JavaFX (OpenJFX 25)
- **Дополнительно:** MinIO, PDFBox, ODFDOM, Apache POI

Комментарий к документации

Фраза "включая лекции, практические задания... и многое другое" была расплывчатой — неясно, какие "другие" типы файлов поддерживаются (только документы или видео?). Нет упоминания о целевой аудитории, версии приложения или интеграции с MinIO. Пропущены нефункциональные аспекты, такие как кроссплатформенность (Windows/macOS/Linux).

Нет списка поддерживаемых форматов файлов, лимитов размера, метаданных, тем, горячих клавиш или поиска — все это из ТЗ, но пропущено.

Java 11 устарела (на 2025 год актуальна 25). Нет системных требований (ОС, RAM), версий (Maven), исправления имени директории (было "mentepova" с дефисом, но репозиторий без). Нет инструкций по зависимостям.

Заключение.

Сильные стороны

- **Полнота и структура:** ТЗ охватывает все ключевые аспекты (от введения до этапов разработки), с чётким разделением на функциональные/нефункциональные требования и критерии приёмки. README логично структурирован (описание, функции, установка), с практическими инструкциями и скриншотами, что облегчает быстрый старт для пользователей.
- **Логичность и соответствие:** Оба документа последовательны: ТЗ следует циклу разработки (проблема → требования → реализация), а README — пользовательскому пути (установка → использование). Они соответствуют базовым стандартам (например, ГОСТ для ТЗ) и ТЗ дисциплины "Тестирование и верификация ПО", фокусируясь на верификации (тест-кейсы, приёмка).
- **Практическая ценность:** Указаны релевантные технологии (JavaFX, MinIO, PDFBox), сценарии использования и функции, что позволяет

понять назначение приложения. README ориентирован на пользователей/разработчиков, с кодовыми примерами.

Слабые стороны (на основе выявленных недочётов)

- **Неактуальная информация:** В ТЗ версии технологий (например, Java 23) устарели к 2025 году (актуальна Java 25); в README — Java 11 и отсутствие релизов на GitHub, что делает инструкции по установке неработоспособными. Скриншоты частично недоступны.
- **Двусмысленные формулировки:** В ТЗ метрики размыты ("разумные сроки" без чисел); в README неясны детали операций (например, рекурсивная ли загрузка папок, как открываются не-PDF файлы), что может привести к путанице.
- **Отсутствие деталей:** ТЗ не включает визуалы (диаграммы, mockup'ы), риски, traceability matrix или детальные test cases. README пропускает конфигурацию MinIO, troubleshooting, лицензию, системные требования, поиск/фильтрацию и обработку ошибок. Оба документа не учитывают maintenance, scalability или доступность (accessibility).

Рекомендации

- **Обновление:** Добавить версии технологий, риски в ТЗ.
- **Улучшение полноты:** Дополнить визуалами (UML, скетчи), метриками (время операций), примерами (use cases, test plans) и разделами (риски, лицензия).
- **Тестирование документов:** Провести peer review или usability-тестирование (например, дать новому пользователю и собрать фидбек).
- **Дальнейшие шаги:** Автоматизированные тесты и версионировать документы (Git).