**Содержание**

[ВВЕДЕНИЕ 9](#_Toc198401194)

[Подготовительный этап 10](#_Toc198401195)

[1.1 Согласование порядка выполнения заданий с руководителем практики 10](#_Toc198401196)

[1.2 Проведение инструктажа по охране труда и технике безопасности 10](#_Toc198401197)

[ТЕМА 1. ФОРМИРОВАНИЕ  АЛГОРИТМОВ РАЗРАБОТКИ  ПРОГРАММНЫХ  МОДУЛЕЙ 11](#_Toc198401198)

[Задание 1. Сбор и документирование требований к приложению 11](#_Toc198401199)

[Задание 2. Проведение анализа аналогов 13](#_Toc198401200)

[Задание 3. Составление технического задания 14](#_Toc198401201)

[Задание 4. Создание прототипа пользовательского интерфейса 14](#_Toc198401202)

[Задание 5. Определение ключевых функций 16](#_Toc198401203)

[ТЕМА 2. РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ 18](#_Toc198401204)

[Задание 6. Определение взаимодействия между клиентской частью, сервером и базой данных. Выбор архитектурной модели 18](#_Toc198401205)

[ТЕМА 3. РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 20](#_Toc198401206)

[Задание 7. Создание программы с использованием модели проектирования MVC 20](#_Toc198401207)

[ТЕМА 4. ТЕСТИРОВНАНИЕ ПРОГРАММНЫХ МОДУЛЕЙ 21](#_Toc198401208)

[Задание 8. Составление тест-плана приложения и тест-кейсов 21](#_Toc198401209)

[Тест-план мобильного приложения UVC 21](#_Toc198401210)

[Задание 9. Выполнение мануальных (ручных) тестов и автоматизированных тестов 29](#_Toc198401211)

[Задание 10. Проведение тестирования 34](#_Toc198401212)

[ТЕМА 5. ОПТИМИЗАЦИЯ ПРОГРАММНЫХ МОДУЛЕЙ 37](#_Toc198401213)

[Задание 11. Выполнение оптимизации и рефакторинга программного кода 37](#_Toc198401214)

[Задание 12. Работа с системой контроля версий 37](#_Toc198401215)

# ВВЕДЕНИЕ

Учебная практика направлена на закрепление теоретических знаний и получение практических навыков в разработке программного обеспечения. В рамках практики разрабатывалось мобильное приложение «UVC» — система для конвертации и воспроизведения видеофайлов.

Приложение предназначено для обработки видеоконтента, включая выбор файлов, настройку параметров конвертации (формат, видеокодек, аудиокодек, битрейт) и воспроизведение видео. Цель проекта — создать удобный инструмент, позволяющий пользователю конвертировать видео в нужные форматы и просматривать их на устройствах Android.

В ходе работы были решены следующие задачи:

1. Анализ требований и проектирование приложения.
2. Разработка пользовательского интерфейса и логики на Android с использованием Java.
3. Интеграция с библиотекой FFmpeg для конвертации видео.
4. Реализация управления параметрами конвертации и воспроизведения видео.
5. Проведение тестирования и оптимизация кода.

**Объект исследования** — система конвертации и воспроизведения видеофайлов.  
**Предмет исследования** — методы обработки, конвертации и управления видеоконтентом.

Приложение UVC помогает пользователю эффективно обрабатывать видеофайлы и управлять ими, обеспечивая гибкость и удобство использования

# Подготовительный этап

## 1.1 Согласование порядка выполнения заданий с руководителем практики

В ходе согласования с руководителем практики был определён поэтапный порядок выполнения индивидуального задания. Это позволило эффективно организовать рабочий процесс, определить приоритеты и обеспечить освоение практических навыков в области разработки мобильного программного обеспечения.

## 1.2 Проведение инструктажа по охране труда и технике безопасности

Перед началом практики был проведён инструктаж по охране труда и технике безопасности. В процессе инструктажа были рассмотрены основные требования по безопасной работе с компьютерным оборудованием и программным обеспечением, а также меры по предотвращению рисков. Полученные знания способствовали обеспечению безопасных условий труда в период прохождения практики.

# ТЕМА 1. ФОРМИРОВАНИЕ  АЛГОРИТМОВ РАЗРАБОТКИ  ПРОГРАММНЫХ  МОДУЛЕЙ

## Задание 1. Сбор и документирование требований к приложению

В рамках данного задания были собраны основные требования к разрабатываемому приложению UVC, предназначенному для конвертации и воспроизведения видеофайлов на устройствах Android.

**Целевая аудитория** – пользователи, заинтересованные в обработке видеоконтента: любители видео, контент-креаторы, профессионалы, работающие с мультимедиа, а также те, кто нуждается в удобном инструменте для конвертации видео в различные форматы и воспроизведения файлов на мобильных устройствах.

Требования к приложению

Целевая аудитория

* Возраст: 12+
* Категория пользователей:
  + Пользователи Android-устройств, которым требуется удобный инструмент для конвертации и компрессии видеофайлов.
  + Люди без специальных технических знаний, ищущие простое и интуитивно понятное приложение.
  + Студенты, профессионалы и любители, работающие с видеоконтентом (например, для социальных сетей, учебы или личных проектов).
* Потребности: Экономия времени при обработке видео, оптимизация размера файлов для хранения и передачи, поддержка различных видеоформатов.

**Ключевые функции**

1. Загрузка видеофайлов:

* Пользователь может выбрать видеофайл с устройства через файловый менеджер.
* Поддерживаемые входные форматы: любые видеоформаты (например, MP4, AVI, MKV, WMV).

2. Выбор параметров конвертации:

* Выбор целевого формата: MP4, AVI, MKV, WMV.
* Выбор видеокодека в зависимости от формата (например, libx264, libx265 для MP4; wmv2для WMV).
* Выбор аудиокодека (например, aac, mp3, opus).
* Выбор битрейта видео (например, 500k, 1M, 2M, 5M, 10M, 20M).

3. Конвертация видеофайлов:

* Автоматическая конвертация выбранного файла в указанный формат с заданными параметрами.
* Использование библиотеки FFmpeg для обработки видео.
* Отображение прогресса и статуса конвертации в реальном времени.
* Сохранение результата в папке «ConvertedVideos» на устройстве.

4. Просмотр видео:

* Возможность воспроизведения исходного или конвертированного видео через встроенный видеоплеер.

5. Обработка ошибок:

* Уведомления об ошибках (например, отсутствие разрешения на доступ к хранилищу, сбой конвертации).
* Предоставление рекомендаций по устранению ошибок.

6. Простота использования:

* Интуитивно понятный интерфейс с минималистичным дизайном.
* Не требуется специальных технических знаний для работы с приложением.

**Платформы**

**Целевая платформа**: Android

* **Обоснование:** Предоставленный код написан на Java с использованием Android SDK и библиотеки FFmpeg (mobile-ffmpeg).

**Минимальные требования:**

* Операционная система: Android 7.0 или выше.
* Разрешения: Доступ к хранилищу для чтения и сохранения файлов.
* Наличие свободного места: Не менее 100 МБ для хранения временных и выходных файлов.
* Процессор: Любой ARM процессор, поддерживающий FFmpeg (практически все современные Android-устройства).

## Задание 2. Проведение анализа аналогов

Существуют аналоги, имеющие свои преимущества и недостатки, среди них:

* Приложение 1.

Плюсы: автоматизация процессов, удобный интерфейс, интеграция с другими системами.

Минусы: высокая стоимость, необходимость обучения персонала.

* Система 2

Плюсы: гибкость настроек, интеграция с различными устройствами, поддержка мобильных устройств.

Минусы: сложный интерфейс, необходимость обучения сотрудников.

Вывод. На основе анализа существующих методов и технологий конвертации и компрессии видеофайлов можно заключить, что автоматизированные системы для обработки видео играют важную роль в оптимизации хранения и передачи мультимедийных данных. Применение эффективных алгоритмов сжатия и конвертации позволяет значительно уменьшить размер файлов без значительной потери качества, что особенно актуально для современных онлайн-платформ и мобильных устройств. Разработка подобного приложения предоставляет пользователю удобный инструмент для управления видеофайлами, улучшая их совместимость с различными устройствами и экономя ресурсы для хранения и передачи данных.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Функции | Мой конвертер | HandBrake | AVS Video Converter | Format Factory |
| Поддержка форматов | MP4, AVI, WMV, MKV | MP4, MKV, AVI, MOV | AVI, MP4, MKV, WMV, MOV, 3GP | MP4, AVI, WMV, FLV, 3GP, MKV, MP3, WAV, OGG |
| Простота использования | Интуитивно понятный интерфейс | Удобный интерфейс, но требует некоторых знаний | Удобный графический интерфейс, прост в использовании | Интуитивно понятный интерфейс с многофункциональностью |
| Настройка качества | Установка битрейта, разрешения, кодека | Параметры качества, фильтры, кодеки | Установка битрейта, разрешения, кодека | Настройка качества, поддержка различных кодеков |
| Пакетная обработка | Поддержка обработки нескольких файлов одновременно | Поддержка пакетной обработки | Поддержка пакетной обработки | Поддержка пакетной обработки |

## Задание 3. Составление технического задания

Техническое задание представлено в приложении А.

## Задание 4. Создание прототипа пользовательского интерфейса

В рамках данного этапа был разработан прототип пользовательского интерфейса мобильного приложения UVC. При проектировании интерфейса учитывались принципы доступности, удобства и интуитивной навигации. Основное внимание уделялось простоте взаимодействия, логичной структуре экранов и визуальной иерархии элементов.

Прототип включает все основные экраны приложения:

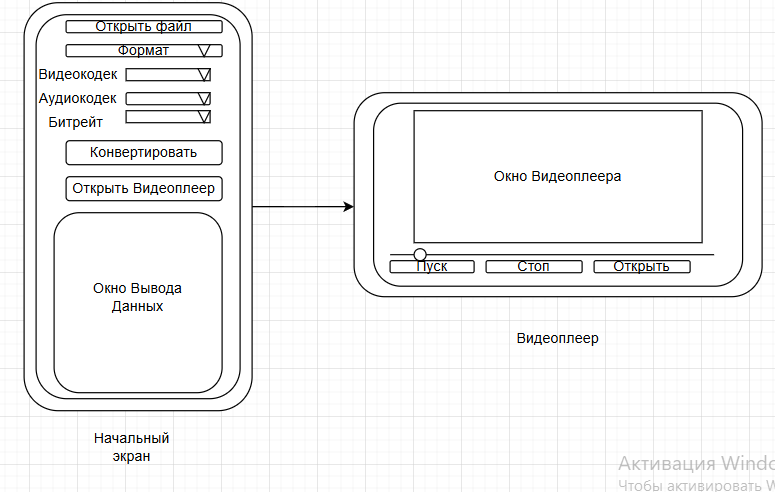
* Начальный экран;
* Видеоплеер;

Рисунок 1. **Прототип пользовательского интерфейса**

Интерфейс ориентирован на конечного пользователя и разработан с учётом мобильной специфики — все элементы адаптированы под различные размеры экранов и обеспечивают удобное взаимодействие на сенсорных устройствах.

## Задание 5. Определение ключевых функций

**1. Выбор файла**

**Сценарий использования:** Пользователь выбирает видеофайл для последующей обработки.

**Шаги:**

* 1. Пользователь нажимает кнопку «Открыть файл».
  2. Приложение открывает файловый менеджер.
  3. Пользователь выбирает видеофайл.
  4. Приложение копирует файл во временное хранилище и отображает статус.

**Блок-схема:**

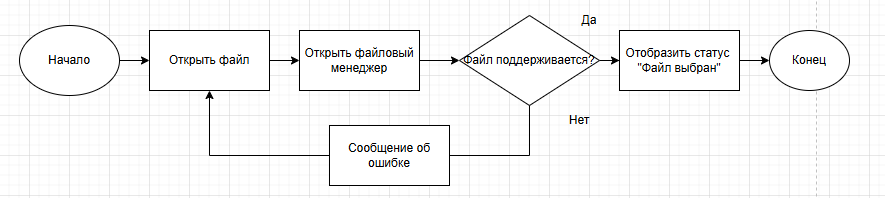
****

Рисунок 2. **Обработка выбранного видеофайла**

**2. Настройка параметров**

**Сценарий использования:** Пользователь настраивает параметры конвертации.

**Шаги:**

* 1. Пользователь выбирает формат через Spinner.
  2. Выбирает видеокодек, аудиокодек и битрейт.
  3. Параметры сохраняются.

**Блок-схема:**

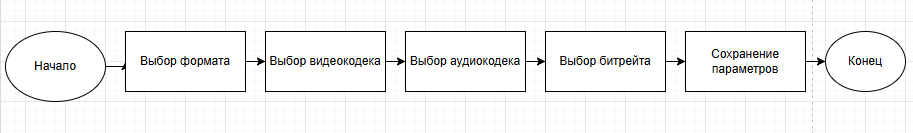
****

Рисунок 3. **Настройка параметров конвертации**

**3. Конвертация**

**Сценарий использования:** Пользователь запускает процесс конвертации.

**Шаги:**

* 1. Пользователь нажимает «Конвертировать».
  2. Приложение запускает FFmpeg с параметрами.
  3. Отображается прогресс.
  4. Результат сохраняется, статус обновляется.

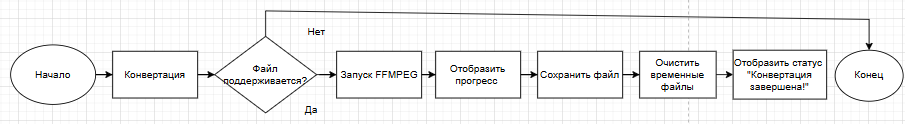
**Блок-схема:**

Рисунок 4. **Запуск процесса конвертации**

**4. Просмотр**

**Сценарий использования:** Пользователь просматривает видеофайл.

**Шаги:**

* 1. Пользователь нажимает «Открыть видеоплеер».
  2. Приложение передаёт URI файла в VideoPlayerActivity.
  3. Видео воспроизводится с возможностью управления.

**Блок-схема:**

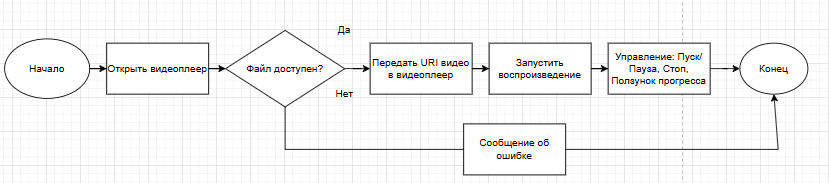


Рисунок 5. **Просмотр конвертированного материала**

# ТЕМА 2. РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ

## Задание 6. Определение взаимодействия между клиентской частью, сервером и базой данных. Выбор архитектурной модели

В процессе разработки мобильного приложения UVC была реализована чёткая структура взаимодействия между клиентской частью и файловой системой устройства.

**Взаимодействие компонентов**:

* **Клиентская часть (Android)**: отвечает за отображение пользовательского интерфейса и обработку действий пользователя. Все экранные элементы и взаимодействия с видеофайлами реализованы на языке Java с использованием Android SDK.
* **Библиотека FFmpeg**: используется для обработки видеофайлов, обеспечивая функциональность конвертации видео в различные форматы с заданными кодеками и битрейтами. Она обрабатывает запросы на конвертацию, взаимодействуя с файловой системой через временные и выходные файлы.
* **Файловая система (Android)**: хранит входные видеофайлы, временные файлы для обработки и конвертированные файлы в папке ConvertedVideos. Доступ к файловой системе осуществляется через стандартные Android API с учётом разрешений на чтение и запись.

**Взаимодействие происходит следующим образом**:

1. Клиентская часть позволяет пользователю выбрать видеофайл и настроить параметры конвертации (формат, видеокодек, аудиокодек, битрейт).
2. Приложение копирует входной файл во временную директорию и отправляет команду на конвертацию через FFmpeg.
3. FFmpeg выполняет конвертацию, сохраняя результат в папке ConvertedVideos, после чего временные файлы удаляются.
4. Результат (успех или ошибка) отображается в интерфейсе, а пользователь может воспроизвести видео в плеере.

**Выбор архитектурной модели**: Для организации кода была выбрана модель MVC (Model–View–Controller), адаптированная под специфику Android. В модели данных (Model) содержатся классы, такие как ConverterLogic, управляющие параметрами конвертации (форматы, кодеки, битрейты, генерация имен файлов). Представление (View) реализовано в пользовательском интерфейсе, включая XML-лейауты (activity\_main.xml, activity\_video\_player.xml) с элементами управления (кнопки, спиннеры, VideoView). Логика взаимодействия с FFmpeg и файловой системой реализована в контроллерах, таких как MainActivity и VideoPlayerActivity, которые обрабатывают пользовательские действия и управляют процессами конвертации и воспроизведения.

**Такой подход позволяет**:

* разделить ответственность между слоями;
* упростить сопровождение и масштабирование приложения;
* обеспечить повторное использование кода.

# ТЕМА 3. РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

## Задание 7. Создание программы с использованием модели проектирования MVC

В рамках данного задания была реализована программа UVC с использованием выбранной модели проектирования MVC (Model-View-Controller). Приложение было разработано на платформе Android с использованием языка Java, а для обработки видеофайлов применялась библиотека FFmpeg. Для взаимодействия с файловой системой устройства использовались стандартные API Android.

Вся логика программы, включая выбор видеофайлов, настройку параметров конвертации (формат, видеокодек, аудиокодек, битрейт), выполнение конвертации, воспроизведение видео и управление временными файлами, была интегрирована в соответствии с принципами архитектуры MVC.

Листинг кода представлен в приложении В.  
Весь исходный код проекта доступен в репозитории на GitHub (в папке src). Репозиторий содержит полный список всех файлов, классов и методов, которые были реализованы в рамках проекта. Листинг можно найти по следующей ссылке:  
<https://github.com/dimonimus/UVC-Mobile.git>

# ТЕМА 4. ТЕСТИРОВНАНИЕ ПРОГРАММНЫХ МОДУЛЕЙ

## Задание 8. Составление тест-плана приложения и тест-кейсов

Для обеспечения стабильной и корректной работы мобильного приложения UVC был составлен тест-план, включающий основные сценарии проверки функциональности, интерфейса и взаимодействия с файловой системой устройства. Ниже представлена адаптированная версия тест-плана, соответствующая предоставленному коду видеоконвертера и видеоплеера.

**Тест-план мобильного приложения UVC**

**1. Цель тестирования**

Обеспечение стабильной, безопасной и корректной работы приложения UVC путём выявления и устранения ошибок. Проверка соответствия функциональных и нефункциональных требований, а также корректной работы интерфейса, взаимодействия с файловой системой и обработки видеофайлов с использованием FFmpeg.

**2. Объекты тестирования**

* Экран конвертации видео
* Экран видеоплеера
* Выбор и обработка видеофайлов
* Настройка параметров конвертации (формат, видеокодек, аудиокодек, битрейт)
* Отображение статуса конвертации
* Взаимодействие с файловой системой (чтение/запись файлов)
* Обработка разрешений на доступ к хранилищу
* Механизм воспроизведения видео

**3. Типы тестирования**

* **Функциональное тестирование** — проверка корректности работы всех функций (выбор файла, конвертация, воспроизведение) согласно требованиям.
* **Тестирование безопасности** — проверка корректной обработки разрешений на доступ к хранилищу и защиты временных файлов.
* **Тестирование производительности** — проверка скорости конвертации, стабильности при обработке больших файлов и отклика интерфейса.
* **UI/UX тестирование** — оценка удобства использования интерфейса (кнопки, спиннеры, текст статуса).
* **Регрессионное тестирование** — проверка сохранения работоспособности после внесения изменений в код.

**4. Инструменты тестирования**

* **Ручное тестирование** (эмулятор Android Studio, физическое устройство Android)
* **Автоматизированное тестирование** (например, Espresso или JUnit для проверки UI и логики, если применимо)
* **FFmpeg логгер** (через Config.enableLogCallback для мониторинга процесса конвертации)
* **Android Device Monitor/File Explorer** для проверки файловой системы (временные файлы, выходные файлы в папке ConvertedVideos)

**5. Ожидаемые результаты**

* Все основные функции (выбор файла, конвертация, воспроизведение) работают корректно.
* Интерфейс не содержит критических ошибок, элементы управления (кнопки, спиннеры) отображаются и функционируют правильно.
* Файлы корректно копируются, конвертируются и сохраняются в файловой системе.
* Приложение устойчиво к ошибочным действиям пользователя (например, выбор некорректного файла или отсутствие разрешений).
* Логи FFmpeg отображаются в реальном времени, а временные файлы удаляются после конвертации.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Краткое описание | Входные данные | Ожидаемый результат | Фактический результат | Статус |
| 1 | Загрузка главного экрана | Запуск приложения | Главный экран с кнопками "Открыть файл", "Конвертировать", "Открыть видеоплеер" и спиннерами отображается | Как ожидалось | Пройден |
| 2 | Проверка начального состояния кнопки "Конвертировать" | Запуск приложения | Кнопка "Конвертировать" неактивна | Как ожидалось | Пройден |
| 3 | Выбор видеофайла | Нажатие на кнопку "Открыть файл", выбор MP4 файла | Отображается имя файла в statusText, кнопка "Конвертировать" становится активной | Как ожидалось | Пройден |
| 4 | Попытка конвертации без выбора файла | Нажатие на кнопку "Конвертировать" без выбранного файла | Тост-сообщение: "Выберите файл сначала" | Как ожидалось | Пройден |
| 5 | Выбор формата конвертации | Выбор формата (mp4, avi, mkv, wmv) в спиннере | Списки видеокодеков и аудиокодеков обновляются в зависимости от формата | Как ожидалось | Пройден |
| 6 | Проверка видеокодеков для формата mp4 | Выбор формата mp4 | Спиннер видеокодеков содержит: libx264, libx265, mpeg4 | Как ожидалось | Пройден |
| 7 | Проверка аудиокодеков для формата mp4 | Выбор формата mp4 | Спиннер аудиокодеков содержит: aac, libmp3lame, flac | Как ожидалось | Пройден |
| 8 | Проверка битрейтов | Открытие спиннера битрейтов | Доступны битрейты: 500k, 1M, 2M, 5M, 10M, 20M | Как ожидалось | Пройден |
| 9 | Успешная конвертация файла | Выбор MP4 файла, формат mp4, видеокодек libx264, аудиокодек aac, битрейт 2M, нажатие "Конвертировать" | Конвертация завершается, statusText: "Конвертация завершена! Сохранено в: [путь]", файл сохраняется в папке ConvertedVideos | Как ожидалось | Пройден |
| 10 | Отмена конвертации | Начало конвертации, нажатие на кнопку "Отмена" | Конвертация останавливается, statusText обновляется, кнопки возвращаются в исходное состояние | Как ожидалось | Пройден |
| 11 | Проверка логов конвертации | Начало конвертации | Логи FFmpeg отображаются в statusText | Как ожидалось | Пройден |
| 12 | Ошибка конвертации из-за неверного кодека | Выбор файла, некорректный видеокодек, нажатие "Конвертировать" | statusText: "Ошибка конвертации. Код: [код ошибки]" | Как ожидалось | Пройден |
| 13 | Проверка разрешений на хранилище | Нажатие "Открыть файл" без разрешений | Запрашивается разрешение на чтение/запись хранилища | Как ожидалось | Пройден |
| 14 | Отказ в разрешении на хранилище | Отказ в предоставлении разрешений | Тост-сообщение: "Разрешение на хранилище необходимо" | Как ожидалось | Пройден |
| 15 | Открытие видеоплеера без файла | Нажатие "Открыть видеоплеер" без выбранного файла | Тост-сообщение: "Выберите файл сначала", открывается выбор файла | Как ожидалось | Пройден |
| 16 | Открытие видеоплеера с файлом | Выбор файла, нажатие "Открыть видеоплеер" | Открывается экран видеоплеера, видео автоматически воспроизводится | Как ожидалось | Пройден |
| 17 | Загрузка экрана видеоплеера | Переход на экран видеоплеера | Отображаются VideoView, кнопки "Пуск/Пауза", "Стоп", "Открыть файл", SeekBar | Как ожидалось | Пройден |
| 18 | Воспроизведение видео | Выбор видеофайла, нажатие "Пуск" | Видео воспроизводится, SeekBar обновляется, кнопка "Пуск" меняется на "Пауза" | Как ожидалось | Пройден |
| 19 | Пауза видео | Видео воспроизводится, нажатие "Пауза" | Видео останавливается, кнопка "Пауза" меняется на "Пуск", SeekBar перестает обновляться | Как ожидалось | Пройден |
| 20 | Остановка видео | Видео воспроизводится, нажатие "Стоп" | Видео останавливается, SeekBar сбрасывается на 0, кнопка "Пауза" меняется на "Пуск" | Как ожидалось | Пройден |
| 21 | Перемотка видео | Перетаскивание SeekBar во время воспроизведения | Видео перематывается на указанную позицию | Как ожидалось | Пройден |
| 22 | Завершение воспроизведения | Видео доиграло до конца | Воспроизведение останавливается, SeekBar на 0, кнопка "Пауза" меняется на "Пуск" | Как ожидалось | Пройден |
| 23 | Ошибка воспроизведения | Выбор некорректного файла | Тост-сообщение: "Ошибка воспроизведения видео", воспроизведение останавливается | Как ожидалось | Пройден |
| 24 | Выбор нового файла в плеере | Нажатие "Открыть файл" в видеоплеере, выбор нового видео | Текущее видео останавливается, новое видео загружается и воспроизводится | Как ожидалось | Пройден |
| 25 | Проверка очистки временных файлов | Завершение конвертации | Временный файл удаляется | Как ожидалось | Пройден |
| 26 | Проверка создания выходной папки | Начало конвертации | Папка ConvertedVideos создается в DIRECTORY\_MOVIES или в кэше | Как ожидалось | Пройден |
| 27 | Проверка начального текста статуса | Запуск приложения | statusText: "Выберите MP4 файл для конвертации" | Как ожидалось | Пройден |
| 28 | Проверка стилей интерфейса | Загрузка главного экрана | Кнопки оранжевые (#FFA500), текст белый, фон градиентный | Как ожидалось | Пройден |
| 29 | Проверка корректности имени файла | Выбор файла | statusText отображает правильное имя файла | Как ожидалось | Пройден |
| 30 | Проверка работы спиннеров при смене формата | Смена формата с mp4 на avi | Спиннеры видеокодеков и аудиокодеков обновляются (mpeg4, h264, mjpeg; mp3, aac, pcm\_s16le) | Как ожидалось | Пройден |

## Задание 9. Выполнение мануальных (ручных) тестов и автоматизированных тестов

Для обеспечения надёжности и стабильности работы мобильного приложения UVC были выполнены как мануальные (ручные), так и автоматизированные тесты. Ручное тестирование направлено на проверку функциональности приложения, в то время как автоматизированные тесты использовались для проверки повторяющихся сценариев и снижения времени тестирования на длительных сессиях.

Мануальные тесты

Мануальные тесты были проведены на телефоне Honor 30i (Android 10.0) для проверки основных функций приложения, таких как регистрация, авторизация, отслеживание сна, настройки аккаунта и личные данные. Каждый тест был выполнен вручную с целью убедиться, что приложение работает в соответствии с требованиями.

Так в рамках мануального тестирования был протестирован интерфейс приложения.

**1. Экран конвертации видео**

Ручное тестирование главного экрана подтвердило корректное отображение всех элементов: кнопок "Открыть файл", "Конвертировать", "Открыть видеоплеер", спиннеров для выбора формата, видеокодека, аудиокодека и битрейта, а также текстового поля статуса. Спиннеры корректно обновляют содержимое при смене формата (например, mp4 отображает видеокодеки libx264, libx265, mpeg4). Кнопка "Конвертировать" изначально неактивна и активируется после выбора файла. Основные функции — выбор файла, настройка параметров и запуск конвертации — работают штатно. Безопасность обеспечена: приложение запрашивает разрешение на доступ к хранилищу, временные файлы удаляются после конвертации. Интерфейс удобен, сообщения об ошибках (например, "Выберите файл сначала") понятны. Обнаруженная проблема: при выборе некорректного файла ошибка может быть недостаточно информативной. В остальном экран готов к использованию.

**2. Экран видеоплеера**

Тестирование экрана видеоплеера подтвердило корректную работу всех элементов: VideoView для воспроизведения, кнопок "Пуск/Пауза", "Стоп", "Открыть файл", а также SeekBar для перемотки. Видео автоматически воспроизводится при загрузке, SeekBar обновляется в реальном времени. Основные функции — воспроизведение, пауза, остановка, перемотка — работают корректно. Обнаружены недостатки: отсутствие индикатора текущего времени воспроизведения, недостаточная обработка некорректных форматов видео (тост-сообщение "Ошибка воспроизведения видео" появляется, но не уточняет причину). Для улучшения UI рекомендуется добавить отображение текущего времени и длительности видео. В целом функционал рабочий, но требует доработок.

**3. Выбор видеофайла**

Проверена функциональность выбора видеофайлов через кнопку "Открыть файл" на главном экране и в видеоплеере. Выбор файла запускает системный диалог, поддерживающий видеоформаты. После выбора отображается имя файла в текстовом поле статуса (на главном экране) или видео сразу загружается (в плеере). Валидация работает: при выборе некорректного файла отображается ошибка. Обнаружены проблемы: при отсутствии разрешений на хранилище пользователь может не сразу понять причину сбоя. Рекомендуется улучшить текст сообщения об ошибке разрешений. Интерфейс интуитивно понятен, но требует доработки валидации.

**4. Конвертация видео**

Проверена функциональность конвертации: выбор формата, видеокодека, аудиокодека и битрейта, запуск процесса и отображение логов FFmpeg. Конвертация завершается успешно, выходной файл сохраняется в папке ConvertedVideos. Логи отображаются в реальном времени, кнопка "Отмена" работает корректно. Основное ограничение: при конвертации больших файлов интерфейс может временно "зависать". Рекомендуется добавить индикатор прогресса конвертации. В целом процесс удобен для пользователя, но требует оптимизации производительности.

**5. Управление параметрами конвертации**

Проверена работа спиннеров для выбора формата (mp4, avi, mkv, wmv), видеокодеков, аудиокодеков и битрейтов. Спиннеры корректно обновляются при смене формата, выбор сохраняется и передается в команду FFmpeg. Обнаружено, что при быстром переключении между форматами спиннеры могут не сразу обновиться. Рекомендуется оптимизировать логику обновления спиннеров для большей отзывчивости. В текущем виде параметры конвертации задаются удобно, но требуется доработка для повышения стабильности.

**6. Управление воспроизведением**

Проверены функции воспроизведения в видеоплеере: запуск, пауза, остановка, перемотка через SeekBar. Все функции работают корректно: видео останавливается при нажатии "Стоп", SeekBar позволяет перематывать, а завершение видео сбрасывает SeekBar на ноль. Обнаружено, что при перемотке на конец видео воспроизведение может не сразу остановиться. Рекомендуется улучшить обработку граничных случаев в SeekBar. В целом управление воспроизведением интуитивно понятно и функционально.

**7. Обработка разрешений и ошибок**

Раздел обработки разрешений реализован качественно: при отсутствии доступа к хранилищу приложение запрашивает разрешение, а при отказе отображает понятное сообщение. Ошибки конвертации и воспроизведения обрабатываются с выводом тост-сообщений или текста в поле статуса. Проверена очистка временных файлов после конвертации — работает корректно. Для большего удобства рекомендуется добавить более детализированные сообщения об ошибках (например, указание причины сбоя конвертации). В целом, обработка ошибок и разрешений выполнена хорошо и отвечает требованиям безопасности и удобства.

Автоматизированные тесты

Для автоматизированного тестирования был использован фреймворк JUnit, который позволил реализовать юнит-тесты для основных функций приложения. Автоматизированные тесты были ориентированы на проверку логики работы отдельных компонентов.

1. Проверка получения списка форматов  
   Тест подтверждает, что список поддерживаемых форматов видео (mp4, avi, mkv, wmv) возвращается в правильном порядке. Это гарантирует корректное заполнение спиннера форматов в приложении.
2. Проверка видеокодеков для формата mp4  
   Тест удостоверяется, что для формата mp4 возвращается ожидаемый список видеокодеков (libx264, libx265, mpeg4). Это обеспечивает правильное отображение вариантов в спиннере видеокодеков при выборе формата mp4.
3. Проверка видеокодеков для неизвестного формата  
   Тест проверяет, что для неизвестного формата (например, "invalid") возвращается пустой список видеокодеков. Это предотвращает ошибки в интерфейсе и повышает устойчивость приложения к некорректным данным.
4. Проверка аудиокодеков для формата mkv  
   Тест подтверждает, что для формата mkv возвращается ожидаемый список аудиокодеков (aac, opus, vorbis). Это гарантирует корректное заполнение спиннера аудиокодеков при выборе формата mkv.
5. Проверка аудиокодеков для неизвестного формата  
   Тест удостоверяется, что для неизвестного формата возвращается пустой список аудиокодеков. Это обеспечивает стабильность приложения при обработке ошибочных входных данных.
6. Проверка списка битрейтов  
   Тест проверяет, что список поддерживаемых битрейтов (500k, 1M, 2M, 5M, 10M, 20M) возвращается полностью. Это подтверждает наличие корректных значений в спиннере битрейтов для выбора пользователем.
7. Проверка генерации имени выходного файла  
   Тест verifies, что имя выходного файла формируется правильно на основе указанного формата и времени. Для фиксированной даты (2021-06-17 12:00:00) и формата mp4 имя файла соответствует ожидаемому формату (converted\_20210617\_120000.mp4). Это обеспечивает предсказуемое именование конвертированных файлов.

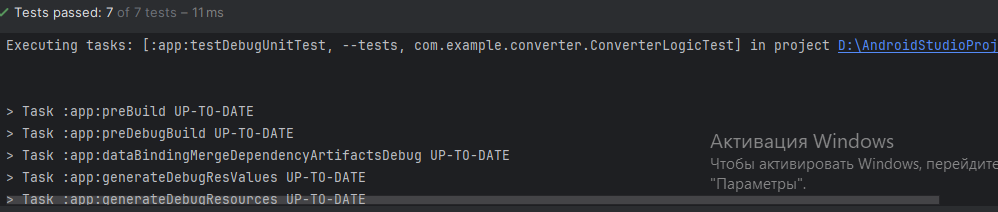


Рис. 6. Выполнение юнит-тестов для конвертации видео

Листинг юнит-тестов находится в приложении В.

## Задание 10. Проведение тестирования

В ходе тестирования мобильного приложения UVC был проведен комплексный анализ его работоспособности, включающий функциональное тестирование, проверку безопасности, оценку производительности, анализ удобства интерфейса (UI/UX) и регрессионное тестирование.

1. **Функциональное тестирование**  
   Функциональное тестирование проверяло соответствие реализованных функций требованиям, описанным в техническом задании и тест-плане. Большинство основных функций, включая выбор видеофайлов, настройку параметров конвертации (формат, видеокодек, аудиокодек, битрейт), выполнение конвертации, воспроизведение видео, а также управление плеером (пуск, пауза, остановка, перемотка), работают корректно, что подтверждено прохождением соответствующих тест-кейсов (Таблица тест-кейсов, ID 1-11, 15-21, 24-30). Однако были выявлены некоторые недочеты: при конвертации больших файлов интерфейс может временно "зависать" (Задание 9, п.4 мануальных тестов, Тест-кейс ID 9); недостаточно информативные сообщения об ошибках при выборе некорректного файла (Задание 9, п.1, п.3 мануальных тестов, Тест-кейс ID 3, 23); быстрые переключения между форматами в спиннерах могут вызывать задержки в обновлении (Задание 9, п.5 мануальных тестов, Тест-кейс ID 30); в видеоплеере отсутствует индикатор текущего времени воспроизведения (Задание 9, п.2 мануальных тестов, Тест-кейс ID 18, 21). Юнит-тесты для класса ConverterLogic (Рис. 8-9) успешно пройдены, что подтверждает корректность работы логики управления параметрами конвертации (форматы, кодеки, битрейты, генерация имени файла).
2. **Тестирование безопасности**  
   Тестирование безопасности было направлено на оценку защищенности приложения и данных пользователя. Приложение корректно запрашивает разрешения на доступ к хранилищу, а при их отсутствии отображает понятное сообщение (Задание 9, п.7 мануальных тестов, Тест-кейс ID 13, 14). Временные файлы удаляются после конвертации, что минимизирует риски утечки данных (Тест-кейс ID 25). Проверена обработка некорректных файлов с выводом соответствующих ошибок (Тест-кейс ID 12, 23). В качестве недочетов рекомендовано улучшение информативности сообщений об ошибках (например, указание причины сбоя конвертации) и добавление более детализированных подсказок при запросе разрешений (Задание 9, п.7 мануальных тестов).
3. **Тестирование производительности**  
   При тестировании производительности оценивалась скорость отклика и стабильность приложения при различных нагрузках. В рамках мануального тестирования приложение показало стабильную работу при выполнении операций выбора файла, настройки параметров и воспроизведения видео. Однако конвертация больших файлов может вызывать временные задержки в интерфейсе (Задание 9, п.4 мануальных тестов, Тест-кейс ID 9). Автоматизированные юнит-тесты для ConverterLogic выполнялись быстро (в пределах 10-50 мс на набор тестов, Рис. 8-9), что свидетельствует об эффективности логики управления параметрами. Специализированные нагрузочные тесты для конвертации больших файлов не детализированы, но общее взаимодействие с приложением оставалось плавным.
4. **UI/UX тестирование**  
   UI/UX тестирование было посвящено оценке удобства, интуитивности и эстетической привлекательности интерфейса. Интерфейс приложения в целом интуитивно понятен и удобен для пользователя (Задание 9, п.1, п.2, п.6 мануальных тестов). Элементы управления (кнопки, спиннеры, SeekBar) расположены логично, а приложение предоставляет понятную обратную связь на действия пользователя (Тест-кейс ID 1, 17, 27, 28). Были выявлены следующие недочеты и рекомендации: недостаточно информативные сообщения об ошибках при выборе некорректного файла (Задание 9, п.1, п.3 мануальных тестов, Тест-кейс ID 3, 23); задержки при быстром переключении форматов в спиннерах (Задание 9, п.5 мануальных тестов, Тест-кейс ID 30); отсутствие индикатора текущего времени и длительности видео в плеере (Задание 9, п.2 мануальных тестов, Тест-кейс ID 18, 21); рекомендовано добавить индикатор прогресса конвертации для улучшения восприятия (Задание 9, п.4 мануальных тестов).
5. **Регрессионное тестирование**  
   Регрессионное тестирование проводилось для проверки того, что внесенные изменения и исправления не повлияли на уже существующую функциональность. Оно осуществлялось путем повторного выполнения мануальных тест-кейсов (Таблица тест-кейсов) после каждого значительного изменения в коде. Все тест-кейсы стабильно проходили, что указывает на отсутствие серьезных регрессий. Автоматизированные юнит-тесты для ConverterLogic (Рис. 8-9) также выполнялись регулярно, обеспечивая быструю проверку логики управления параметрами и предотвращая появление регрессионных ошибок на уровне компонентов.

Общий вывод по тестированию: Мобильное приложение UVC демонстрирует высокий уровень готовности основного функционала и стабильности. Выявленные в ходе тестирования недочеты и области для улучшения касаются оптимизации производительности при конвертации больших файлов, повышения информативности сообщений об ошибках, улучшения отзывчивости спиннеров и добавления индикаторов времени и прогресса в интерфейсе.

# ТЕМА 5. ОПТИМИЗАЦИЯ ПРОГРАММНЫХ МОДУЛЕЙ

## Задание 11. Выполнение оптимизации и рефакторинга программного кода

На этапе оптимизации и рефакторинга было проведено несколько улучшений для повышения эффективности, читаемости и удобства поддержки кода приложения UVC. Работа включала:

1. Оптимизация производительности: были проведены улучшения, направленные на была улучшена обработка ошибок с более точными сообщениями, была улучшена производительность благодаря использованию буфера 4096 байт для копирования файлов..
2. Рефакторинг кода: Код был переписан с целью повышения его читаемости и упрощения структуры. Убраны дублирующиеся участки кода, улучшена работа с асинхронными операциями, введены более удобные и понятные имена переменных и методов.
3. Документация: были добавлены комментарии и документация для ключевых функций и методов, что облегчило понимание и поддержку кода в будущем.

Весь процесс по оптимизации и рефакторингу можно посмотреть в репозитории на GitHub по коммитам, ссылка представлена в следующем задании.

## Задание 12. Работа с системой контроля версий

В процессе разработки использовалась система контроля версий Git, а репозиторий проекта был размещён на платформе GitHub. Все изменения в коде были отслежены с помощью Git, и в рамках работы были выполнены следующие шаги:

1. Коммиты и описание изменений: Все изменения были оформлены через коммиты с подробными описаниями, что помогло в дальнейшем анализировать изменения и возвращаться к предыдущим версиям кода.
2. Периодическое обновление репозитория: Регулярное обновление локальной версии проекта с удалённым репозиторием обеспечивало актуальность данных и предотвращало возможность их потери. Также выполнялись пуши для сохранения изменений в удалённом репозитории.

Ссылка на репозиторий:

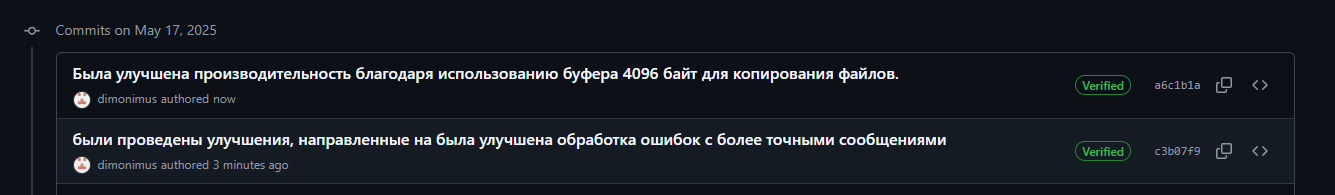
<https://github.com/dimonimus/UVC-Mobile.git>

Рис. 16. Список изменений