

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

на разработку настольного приложения «WallHub»
(Управление фоновыми изображениями рабочего стола)

Версия 1.0

Дата: 20.01.2026

1. Общие сведения

Наименование системы: Настольное приложение «WallHub»

Назначение системы: Локальное приложение для управления коллекцией статических обоев рабочего стола в среде WSL/Windows.

Заказчик: [Медведев Д.В. и Анисимов Н.С.]

Исполнитель: [Медведев Д.В. и Анисимов Н.С.]

2. Плановые сроки и бюджет

Сроки разработки: 4 недели

Бюджет: Учебный проект, без коммерческой стоимости

Этапы разработки:

Этап	Сроки
Анализ требований и проектирование	1 неделя
Разработка интерфейса	1 неделя
Реализация функций работы с файлами	1 неделя
Тестирование и отладка	1 неделя

3. Назначение и цели

Цели создания системы:

- Создать простой и удобный инструмент для управления коллекцией обоев в WSL-окружении
- Обеспечить быструю установку выбранных обоев на рабочий стол Windows
- Предоставить базовые операции управления файлами (добавление, удаление, обновление)
- Реализовать функцию скачивания обоев по URL

4. Требования к системе

4.1. Функциональные требования

Управление локальной коллекцией:

- Создание и поддержание папки /walls для хранения изображений
- Отображение коллекции в виде сетки с превью (320×180 пикселей)
- Автоматическое обновление списка при изменении содержимого папки
- Поддержка форматов: JPG, JPEG, PNG, BMP

Операции с файлами:

- Добавление файлов через диалог выбора (множественный выбор)
- Удаление выбранных файлов из коллекции с подтверждением
- Обновление списка файлов вручную
- Скачивание изображений по URL с использованием wget

Установка обоев:

- Установка выбранного изображения в качестве обоев Windows
- Поддержка двух методов установки:
 - Через PowerShell и SystemParametersInfo API
 - Через изменение реестра Windows
- Предоставление инструкций для ручной установки при неудаче

Интерфейс пользователя:

- Основное окно с кнопками управления
- Список/сетка файлов с превью

- Панель статуса с информацией о текущих операциях
- Стил: светлая тема с закругленными элементами

4.2. Нефункциональные требования

Требования к производительности:

- Загрузка превью не должна блокировать интерфейс
- Потребление памяти: до 50 МБ в активном режиме
- Время реакции на действия пользователя: < 500 мс

Требования к совместимости:

- Операционная система: Windows 10/11 с установленным WSL
- Требуемые компоненты: wslpath, PowerShell, wget

Требования к интерфейсу:

- Цветовая схема: светлая (#f8f9fa фон, #dee2e6 границы)
- Стил кнопок:
 - Зеленая кнопка "Установить" (#4CAF50)
 - Синяя кнопка "Скачать" (#2196F3)
- Отображение файлов: иконный режим с подписями
- Статусная панель: зеленая (#e8f5e8)

Требования к надежности:

- Проверка существования файлов перед операциями
- Обработка ошибок при копировании/удалении файлов
- Резервное копирование не требуется (файлы хранятся в исходных локациях)

5. Архитектура системы

5.1. Компоненты системы

Клиентское приложение:

- Язык: C++
- Фреймворк: Qt 5/6
- Компоненты:
 - QListWidget для отображения файлов
 - QPushButton для управления
 - QTimer для мониторинга папки
 - QProcess для вызова внешних команд

Внешние зависимости:

- wslpath: конвертация путей WSL → Windows
- PowerShell: установка обоев через Windows API
- wget: скачивание файлов по URL
- Командная строка Windows: работа с реестром

5.2. Структура данных

```
WallHub/
├── walls/           # Папка с обоями
│   ├── image1.jpg
│   ├── image2.png
│   └── ...
└── WallHub.exe     # Исполняемый файл
```

└─ (системные файлы Qt)

5.3. Алгоритмы работы

1. Инициализация:

- a. Создание папки walls при отсутствии
- b. Сканирование папки и загрузка превью
- c. Запуск таймера мониторинга (каждые 2 секунды)

2. Установка обоев:

Выбор файла → Конвертация пути → PowerShell команда → Проверка результата

3. Скачивание:

Ввод URL → Генерация имени → wget → Проверка размера → Добавление в коллекцию

6. Модель данных

Основные сущности:

WallpaperFile:

- filename: string
- path: string
- preview: QPixmap
- filesize: int
- lastModified: QDateTime

7. Интерфейс пользователя

Макет главного окна:



8. Тестирование

Тестовые сценарии:

1. Добавление файлов через диалог выбора
2. Установка обоев с различными путями
3. Удаление файлов с подтверждением
4. Скачивание по корректным и некорректным URL
5. Проверка автообновления при изменении папки

Критерии приемки:

- Все кнопки выполняют заявленные функции
- Интерфейс не блокируется при длительных операциях
- Обработка ошибок при отсутствии внешних утилит
- Корректная работа в WSL окружении

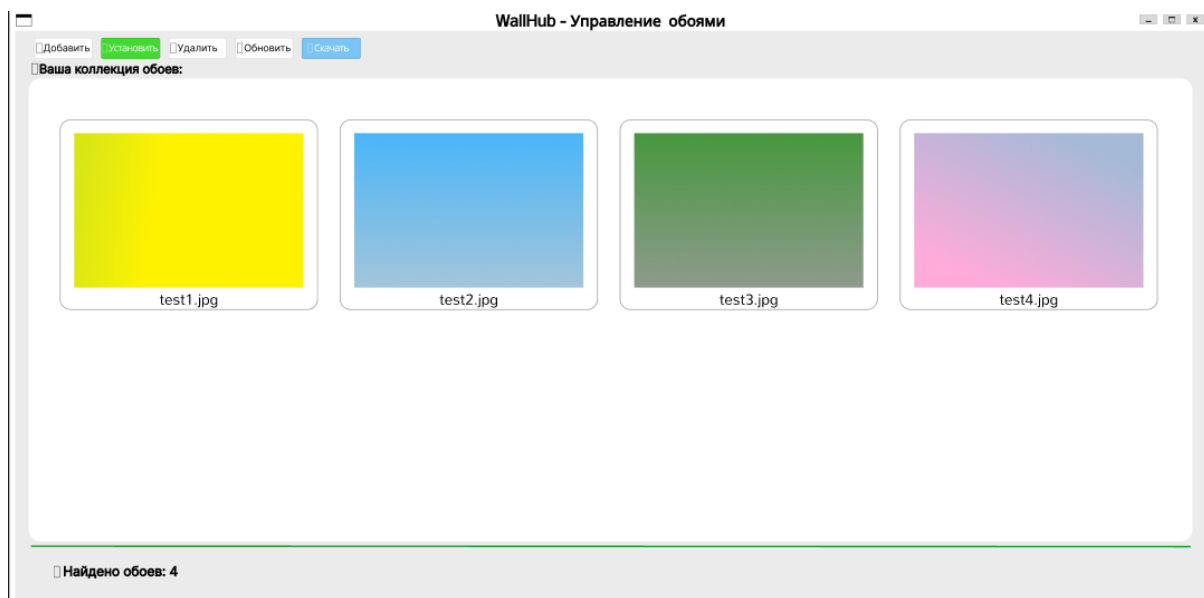
9. Ограничения

Известные ограничения:

- Работает только в WSL окружении
- Требуется наличия wslpath, PowerShell
- Не поддерживает динамические/видео обои
- Нет синхронизации между устройствами
- Локальное хранение без облачных функций

10. Приложения

Приложение А: Снимки экрана



Приложение В: Цветовая схема

