

РАЗРАБОТКА ИНТЕРАКТИВНОГО ПРИЛОЖЕНИЯ ДЛЯ ПОИСКА И КАТАЛОГИЗАЦИИ МЕДИАКОНТЕНТА

Введение. В современной цифровой среде наблюдается стремительный рост объемов медиаконтента, что создает значительные сложности для пользователей в организации эффективного доступа к информации. Разрозненность платформ распространения контента и отсутствие унифицированных инструментов управления приводят к фрагментации пользовательского опыта. Существующие решения часто ограничены узкой специализацией и не обеспечивают комплексного подхода к каталогизации разнородных медиаресурсов.

Возникает объективная потребность в создании интегрированных систем, способных объединять контент из различных источников через стандартизированные интерфейсы API. Такие системы должны обеспечивать персонализированную работу с медиабibliothекми, предлагая гибкие инструменты для систематизации и поиска. Особую значимость приобретают решения, сочетающие производительность настольных приложений с широкими возможностями веб-интеграции [1].

Ключевой целью исследования является разработка программного обеспечения, обеспечивающего унифицированный доступ и эффективную систематизацию медиаконтента. Достижение этой цели предполагает интеграцию разнородных источников данных через стандартизированные API-интерфейсы, что позволит реализовать персонализированные подходы к управлению медиабibliothекми.

В качестве объекта исследования выступают сложные процессы интеграции, каталогизации и визуального представления медиаконтента, поступающего из распределенных источников. Особое внимание уделяется особенностям реализации этих процессов в рамках desktop-приложений, что обусловлено требованиями к производительности и функциональности.

Предмет исследования охватывает комплекс методов и алгоритмов, направленных на решение ключевых задач. В первую очередь, это методы объединения разнородных медиаресурсов через API-интерфейсы, обеспечивающие согласованную работу с различными форматами данных.

Основная часть. Для достижения поставленной цели было разработано специализированное desktop-приложение, архитектура которого построена на принципах модульности и четкого разделения ответственности между компонентами. В качестве технологического стека выбраны платформа Visual Studio и язык программирования C++, что обусловлено требованиями к высокой производительности при рендеринге графических элементов интерфейса и работе с памятью в условиях отображения большого массива медиаданных [2].

Для получения актуальной информации об аниме-контенте приложение взаимодействует с REST API платформы Shikimori.one. Обмен данными осуществляется через защищенные HTTPS-запросы, обеспечивающие целостность и конфиденциальность передаваемой информации. Полученные данные в формате JSON подвергаются парсингу с последующей десериализацией во внутренние структуры приложения, что позволяет унифицировать обработку разнородных данных, поступающих из распределенного источника.

Модуль пользовательского интерфейса реализует трехкомпонентную архитектуру представления:

1. Верхний блок управления содержит элементы для первичной навигации и управления состоянием системы: поле ввода для контекстного поиска по названию, элемент отображения идентификатора авторизованного пользователя и кнопку сброса интерфейса к исходному состоянию.

2. Центральный блок навигации обеспечивает переключение между двумя основными режимами просмотра: полным каталогом и персонализированной коллекцией «Избранное». Для оптимизации нагрузки на систему и улучшения юзабилити реализован механизм постраничного вывода контента (пагинация), управляемый соответствующими кнопками навигации. Общий вид главного окна приложения представлен на рисунке 1.

На рисунке 2 представлен общий вид панели «Избранное».

3. Блок визуализации контента предназначен для отображения массива данных в виде сетки карточек. Каждая карточка представляет собой комплексный элемент интерфейса, агрегирующий графический постер, текстовое название и интерактивный элемент управления (кнопку добавления/удаления из избранного).

Такой подход к организации интерфейса позволяет эффективно решить задачу управления крупными медиатеками, обеспечивая пользователю интуитивно понятный и производительный инструмент для взаимодействия с контентом.

Реализация функциональности приложения основана на взаимодействии двух ключевых модулей: модуля работы с данными и модуля представления.

Модуль работы с данными отвечает за асинхронное выполнение HTTP-запросов к API Shikimori.one. Полученный ответ в формате JSON обрабатывается с использованием алгоритмов синтаксического анализа, которые извлекают необходимые атрибуты (название, URL изображения постера) и преобразуют их в объекты предметной области. Графические данные постеров загружаются и кэшируются локально для минимизации сетевых запросов и повышения скорости отклика интерфейса.



Рисунок 1 — Общий вид главного окна приложения



Рисунок 2 — Вид панели «Избранное»

Модуль представления динамически формирует интерфейс на основе полученных данных. Основной элемент — карточка контента — генерируется программно.

Алгоритм пагинации рассчитывает подмножество карточек для отображения на текущей странице, исходя из общего количества элементов и заданного лимита на страницу. Это позволяет оптимизировать потребление ресурсов системы, избегая создания чрезмерного количества графических элементов управления одновременно.

В результате, разработанное приложение представляет собой решение, которое на практике реализует описанные во введении принципы интеграции, каталогизации и персонализированного управления медиаконтентом.

Заключение. Проведенное исследование демонстрирует практическую реализацию эффективного подхода к решению проблемы фрагментации медиаконтента. Разработанное приложение позволяет пользователю централизовать работу с аниме-каталогом, обеспечивая быстрый доступ к актуальной информации через интеграцию с внешним API. Ключевым преимуществом системы является возможность персонализированного управления медиатекой через механизм "Избранного", что создает удобную среду для систематизации просматриваемого контента.

Архитектурные решения, включая трехкомпонентную организацию интерфейса и механизм пагинации, направлены на оптимизацию производительности при работе с большими объемами данных. Сочетание возможностей desktop-платформы с веб-технологиями позволяет достичь высокой отзывчивости интерфейса, что выгодно отличает решение от браузерных аналогов.

Таким образом, приложение представляет собой законченный инструмент, который на практике реализует принципы унифицированного доступа и систематизации медиаконтента, предлагая пользователю удобный и производительный способ взаимодействия с персональной медиатекой.

Список цитируемых источников

1. Алпатов, А. Н. Интерфейсы прикладного программирования : учебное пособие / А. Н. Алпатов. — М. : РТУ МИРЭА, 2024. — 157 с.
2. Рысин, М. Л. Основы программирования на языке C++ : учебное пособие / М. Л. Рысин, М. В. Сартаков, О. В. Макеева. — М. : РТУ МИРЭА, 2022. — 118 с.