

Котов Д.С.

*Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: dmitrijkotov634@mail.ru*

Создание мобильного приложения для просмотра расписания

В преддверии нового учебного года вопрос удобного доступа к информации о расписании занятий остро встает у студентов, преподавателей. Существующий сайт mivlgu.ru, предоставляющий данную информацию, обладает рядом недостатков:

- Неудобный интерфейс для получения информации.
- Отсутствие мобильного приложения.
- Отсутствует возможность офлайн-доступа к расписанию.

В связи с этим, целью работы является разработка мобильного приложения, которое позволит студентам:

- Легко и быстро получать информацию о расписании занятий.
- Избегать необходимости авторизации на сайте.
- Использовать приложение в офлайн-режиме.

Приложение представляет собой мобильный клиент, который взаимодействует с API сайта mivlgu.ru. Пользователь выбирает факультет, к которому он относится, а затем указывает свою группу. Пользователь может выбрать просмотр расписания на четные или нечетные недели с помощью соответствующего фильтра. Также он может добавить ярлык с приложением на главный экран своего устройства для быстрого доступа к расписанию занятий. Расписание занятий обновляется через API сайта mivlgu.ru, что обеспечивает актуальность данных для пользователей приложения.

Приложение разработано нативно для платформы Android на языке Kotlin с использованием Android XML для создания пользовательского интерфейса. Kotlin был выбран для проекта из-за своей полной поддержки Android, а также модернизированных возможностей программирования, а Android XML был выбран для описания пользовательского интерфейса из-за его официальной поддержки Android, декларативного подхода, разделения логики и представления, удобства визуального проектирования и поддержки различных устройств.

Архитектура приложения построена на паттерне MVVM, что обеспечивает эффективное управление данными и бизнес-логикой приложения. Оно ориентировано на удобное отображение расписания занятий, позволяет пользователям выбирать факультет и группу. Центральным принципом работы является связь между данными (Model), представлением (View) и моделью-представлением (ViewModel), обеспечивающая четкое разделение бизнес-логики и пользовательского интерфейса. Это позволило создать структурированное приложение, где каждая компонента выполняет свою функцию без лишней зависимости. Навигация в приложении осуществляется с помощью стандартного компонента, что обеспечивает простоту и интуитивность перемещения по различным экранам. В результате чего пользователь легко может переключаться между основными разделами приложения.

Стек технологий приложения включает:

- AndroidX для расширенного набора библиотек поддержки Android.
- Material Design Components для создания пользовательского интерфейса.
- Ktor для создания клиентских HTTP-запросов к API сай-та mivlgu.ru.
- Kotlin Serialization для работы с данными в формате JSON.

Каждое открытое расписание кэшируется в памяти приложения, обеспечивая доступ к нему в офлайн-режиме. Это позволяет пользователям просматривать свое расписание даже без подключения к Интернету.

API сайта mivlgu.ru предоставляет REST API HTTP с параметрами текущего семестра и года, а также возможностью выбора группы или преподавателя. Это позволяет приложению

эффективно обращаться к серверу и получать актуальную информацию о расписании занятий в зависимости от выбранных параметров.

Приложение для расписания занятий успешно загружено и доступно в RuStore – российском сетевом магазине приложений от VK и Министерства цифрового развития. Ссылка на страницу магазина приложений доступна на официальном сайте МИВлГУ.

Разработанное приложение является удобным и функциональным инструментом для студентов. Оно позволяет легко и быстро получать информацию о расписании занятий, что значительно упрощает процесс обучения.

Миханов А.В.

Научный руководитель: к.т.н., доцент кафедры ИС А.Д. Варламов
Муромский институт (филиал) федерального государственного бюджетного
образовательного учреждения высшего образования «Владимирский государственный
университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
artem.mikhanov@mail.ru

Проблемы тестирования адаптеров для подключения к СУБД

Адаптер (другими словами - расширение) для подключения к системе управления базами данных (СУБД) – посредник между СУБД и программным кодом. Адаптер содержит набор функций или объектов, упрощающих работу с базой данных [1]. Для того чтобы подобрать адаптер для использования его в информационной системе, необходимо учитывать целый ряд факторов:

- используемый в системе язык программирования;
- СУБД;
- версия языка программирования;
- версия СУБД;
- тип и версия операционной системы.

Как правило, любой новый адаптер для СУБД поддерживает целый ряд возможных значений как минимум трех последних параметров приведенного списка. А для того, чтобы протестировать адаптер для строго заданного окружения (определенной версии СУБД, языка программирования и операционной системы) требуется выполнить десятки, а то и сотни тестов (с учетом разнообразия типов запросов, типов таблиц, полей и так далее). Не трудно догадаться, что полноценное тестирование определенного адаптера будет включать в себя огромное число тестовых проверок, которое может исчисляться десятками тысяч.

Таким образом, возникает проблема сверхбольшой трудоемкости ручного тестирования. Чтобы снизить ресурсные затраты на тестирование, необходимо автоматизировать этот процесс. По этой причине необходимо разработать систему для автоматического тестирования адаптеров СУБД.

В докладе изложена постановка требований к системе тестирования адаптеров подключения к СУБД, предложены критерии тестирования. В заключении сделан вывод о целесообразности разработки такой системы и оценены перспективы использования подобных систем.

Литература

1. Подключение к базе данных в PHP через mysql, pdo и mysqli // realadmin.ru URL: <https://realadmin.ru/coding/connect-mysql-php.html?ysclid=ltu4j7pr7g418230730> (дата обращения: 16.03.2024).
2. Полное руководство по тестированию баз данных // QaRocks URL: <https://qarocks.ru/database-testing-process/?ysclid=ltu5x7vq6a254825946> (дата обращения: 16.03.2024).
3. DDL, DML, DCL и TCL в языке SQL // Skypro URL: <https://sky.pro/media/gruppy-operatorov-sql/?ysclid=ltekr7sgoi838436866> (дата обращения: 27.02.2024).

Никифорова Д.Е.

Научный руководитель: к.т.н. С.В. Комкова

*Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23*

Разработка автоматизированной системы комплексного учета теплоресурсов промышленного предприятия

Внедрение современных систем для контроля и учета энергоресурсов на всех этапах передачи ее до конечного потребителя в России является одним из возможных путей решения проблемы снижения потерь энергии. Основной задачей комплекса АСКУЭ по сбору и обработке данных является автоматизированный сбор, формирование базы данных и последующий анализ показаний всех устройств учета расхода энергоносителей. Данная система АСКУЭ направлена на обеспечение контроля работы всего энергетического оборудования, а также комплексный и одновременно технический учет энергии. Система разработана в целях применения на промышленных предприятиях. Стратегические цели содержания системы: целенаправленное ведение процесса энергоснабжения и обеспечения смежных и вышестоящих систем управления оперативной и достоверной информацией.

В работе рассматривается приложение учета тепловой энергии промышленного предприятия. Приложение .NET Core, которое осуществляет доступ к базе данных с помощью Entity Framework Core.

Система предоставляет возможность:

- просмотреть информацию об оборудовании;
- редактировать оборудование;
- просматривать информацию о проведенных ремонтах;
- просматривать информацию о поверках оборудования;
- просматривать изображение оборудования;
- просматривать информацию о комплектующих;
- просматривать информацию о помещениях;
- просматривать информацию о показаниях приборов.

Автоматизированная система комплексного учета теплоресурсов гарантирует оперативность и упорядочивание финансовых расчетов за потребление и отпуск энергоносителей при помощи своевременного выявления сверхнормативного их потребления, минимизацию непроизводительных и производственных энергопотерь, уменьшение размеров разбаланса потребления энергоресурсов за счет исключения возможности их хищения, повышения точности учета энергоресурсов, снижения риска неоплаты энергоносителей, за которые производится расчет по нормам потребления населением, эффективный оперативный контроль за рациональным использованием теплоресурсов за счет сокращения времени обработки и сбора данных автоматизированного учета.

Осинцев Д.Н.

Научный руководитель: старший преподаватель кафедры ИС А.В. Булаев
Муромский институт (филиал) федерального государственного бюджетного
образовательного учреждения высшего образования «Владимирский государственный
университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
osincevd@bk.ru

Разработка системы внутрисетевого информирования

Информирование коллектива сотрудников о деятельности компании, клиентов об оказываемых компанией услугах играет важную роль для формирования сплоченности сотрудников и заинтересованности клиентов в новых услугах. Распространение текстовой и графической информации не всегда позволяет достичь желаемых результатов. Использование аудио и видеoinформации позволяет повысить заинтересованность как сотрудников в работе компании, так и клиентов в оказываемых ею услугах.

Применение классических подходов к организации средств аудио и видео информирования требует использования специфического автономного аппаратного обеспечения, что не всегда позволяет синхронизировать распространяемую информацию между территориально распределенными представительствами организации. В этом контексте разработка систем внутреннего информирования, строящихся на базе корпоративных сетей, становится актуальной задачей для ИТ - специалистов и разработчиков программного обеспечения.

Разрабатываемая система (рис. 1) предназначена для распространения аудиоинформации посредством корпоративной сети организации и состоит из сервера ретрансляции, модуля управления (рис. 2) и принимающих устройств.

1. Модуль управления - отвечает за настройку системы и осуществление аудио вещания.
2. Сервер ретрансляции – осуществляет распространение аудио потока по сети, поддерживает большое количество клиентских подключение, ретрансляция осуществляется по протоколу Icescast (разрабатываемым под лицензией GPL, являющимся аналогом протокола Shoutcast компании Nullsoft).
3. Клиентские системы воспроизведения - это приложения или устройства, которые прослушивают канал и взаимодействуют с ним. Данные устройства могут быть:
 - специализированными аппаратными средствами (САС) - устройства на основе микроконтроллеров (например, ESP-32);
 - малобюджетными компьютерами - одноплатные компьютеры (например, OrangePi);
 - компьютеры – клиентские приложения под операционные системы Windows и Linux или современные версии web браузеров.

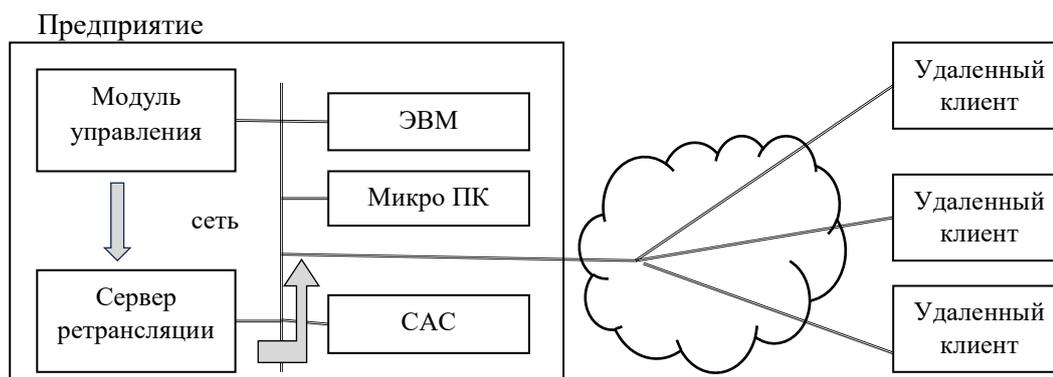


Рис. 1 - Архитектура системы внутрисетевого информирования

В качестве сервера ретрансляции используется приложение icestcast, которое принимает аудиопоток от модуля управления и осуществляет его трансляцию клиентам по протоколу воспринимаемому большим спектром устройств и программ.

Связь сервера ретрансляции и модуля управления идет по защищенному каналу, что препятствует подключению сторонних источников трансляции.

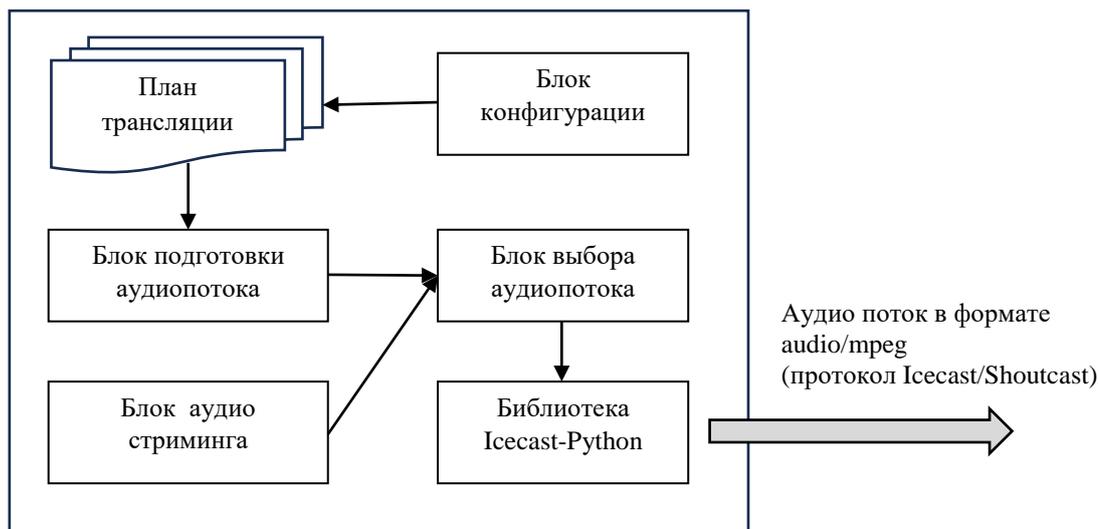


Рис. 2 - Модуль управления

Блок конфигурации обеспечивает задание параметров соединения модуля управления с сервером ретрансляции (IP адрес и порт подключения), позволяет сформировать план трансляции на определенную дату и выполнять его корректировку в реальном масштабе времени.

Блок аудио стриминга обеспечивает взаимодействие модуля управления с микрофоном для проведения трансляции.

Блок подготовки аудиопотока предназначен для преобразования аудиофайлов формата mp3 в структуру данных, принимаемую библиотекой Icast-Python.

Блок выбора аудиопотока позволяет переключаться между аудиопотоком с микрофона и музыкальный поток.

План трансляции представляется файлом, хранящим информацию о времени запуска каждого элемента аудиотрансляции (воспроизведение аудиофайла, голосовой стрим, режим молчания) и ссылку на аудиофайл. Именованное файла плана трансляции хранит дату применения плана, что позволяет осуществлять трансляции в автоматическом режиме.

Разработанная система имеет клиент-серверную архитектуру. Программный модуль трансляции реализован с использованием языка программирования Python; функции ретрансляции аудиопотока реализуются приложением Icast; клиентами, принимающими аудиопоток, выступают: на персональных компьютерах под управлением Windows/Linux – приложение, реализованное на языках Java и Python; специализированные аппаратные платформы – микроконтроллер ESP-32.

При реализации компонентов системы использовался объектно-ориентированный подход программирования, многопоточное программирование, взаимодействие компонентов было организовано посредством стандартизированных сетевых протоколов обмена данными, применение различных сред программирования для реализации элементов системы обусловлено их оптимальностью для выполнения требуемых функций.