

2. Awesomedevlop [Электронный ресурс]. – Режим доступа: <http://awesomedevlop.blogspot.com.by/2016/12/android-studio.html>. – Дата доступа: 17.01.2018.

ПРОГРАММНОЕ СРЕДСТВО ПО ОРГАНИЗАЦИИ ТРАНСЛЯЦИИ ПОТОКОВОГО ВИДЕО ПОСРЕДСТВОМ WOWZA STREAMING ENGINE

*Институт информационных технологий БГУИР,
г. Минск, Республика Беларусь*

Шилкин А.О.

*Пачинин В. И. – зав. кафедрой ИСиТ, к.т.н., доцент
Образцова О. Н. – доцент каф. ИСиТ, к.т.н., доцент*

В работе представлена разработка программного средства обеспечивающего организации трансляции потокового видео посредством wowza streaming engine.

Прямой эфир - это процесс передачи информации в реальном времени, трансляция видео и аудио сигналов с места проведения записи в эфир. Основной особенностью прямого эфира является то, что получаемые данные невозможно редактировать.

В настоящее время метод передачи данных используется повсеместно во всех видах электронных средствах массовой информации. Это различные развлекательные передачи, викторины, интервью, новости. На телевидении в прямом эфире сегодня выходит не так много программ. Чаще всего это новостные сюжеты в жанре прямого отчета, прямые трансляции футбольных матчей, спортивных соревнований, музыкальных конкурсов, важных политических событий.

Среди всех положительных черт прямого эфира, в первую очередь нужно выделить оперативность. Аудитория получает самую свежую информацию «из первых рук». Это подогревает интерес зрителя и вызывает у него желание узнать о случившемся первым.

Главная задача системы – показ потокового видео для пользователей, которые приобрели подписку на просмотр в системе.

В основе работы программного средства будет положена микросервисная архитектура. Это модульный подход к разработке программного обеспечения, основанный на использовании распределённых, слабо связанных заменяемых компонентов, оснащённых стандартизированными интерфейсами для взаимодействия по стандартизированным протоколам. Программные комплексы, разработанные в соответствии с сервис-ориентированной архитектурой, реализуются как набор веб-служб, взаимодействующих по протоколу REST. Интерфейсы компонентов в сервис-ориентированной архитектуре инкапсулируют детали реализации (операционную систему, платформу, язык программирования) от остальных компонентов, таким образом обеспечивая комбинирование и многократное использование компонентов для построения сложных распределённых программных комплексов, обеспечивая независимость от используемых платформ и инструментов разработки, способствуя масштабируемости и управляемости создаваемых систем.

Приложение состоит из следующих серверов:

- сервер Admin microservice – является интерфейсом для управления, контентом, пользователями, подписками;
- сервер Content microservice – предназначен для управления контентом, предоставления данных обычному пользователю, все данные представлены в json;
- сервер User microservice – предназначен для регистрации, авторизации пользователей, все данные представлены в json;
- сервер Subscription microservice – предназначен для оформления подписки, проверки наличия подписки для предоставления контента, все данные представлены в json;
- сервер Frontend – является интерфейсом для обычного пользователя;
- сервер Wowza Streaming Engine – предназначен для транслирования прямых трансляций;
- хранилище файлов – предназначен для хранения картинок и видео файлов;
- сервер Memcached – хранилище для кеша.

Для каждого микросервиса база данных располагается на отдельном сервере. Благодаря этому, нагрузка распределяется на несколько баз данных, вместо одной.

Передача данных осуществляется через протокол TCP/IP.

На рисунке 1 представлена диаграмма развертывания программного средства, демонстрирующая конфигурацию и связь узлов программной системы.

Для обмена данными между клиентом и сервером осуществляется по протоколу HTTPS с использованием сообщений в формате JSON.

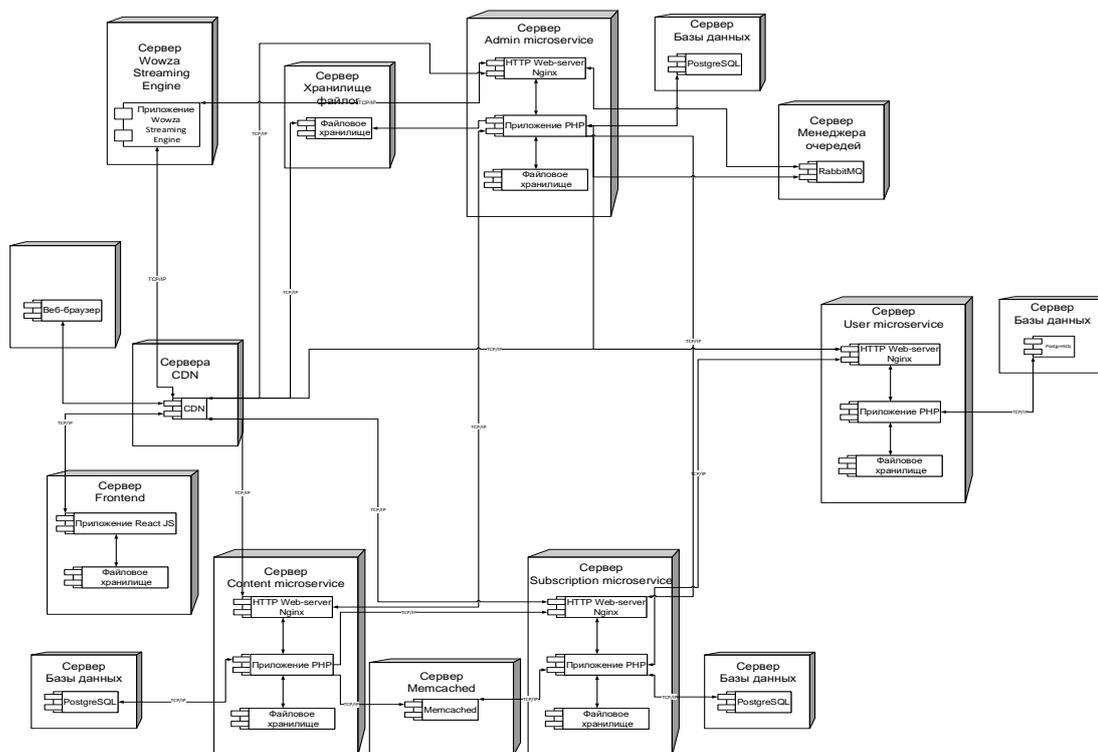


Рисунок 1 - Диаграмма развертывания

В качестве СУБД для данного программного средства была выбрана PostgreSQL. PostgreSQL не просто реляционная, а объектно-реляционная СУБД. Это даёт ему некоторые преимущества над другими SQL базами данных с открытым исходным кодом, такими как MySQL, MariaDB и Firebird.

Фундаментальная характеристика объектно-реляционной базы данных — это поддержка пользовательских объектов и их поведения, включая типы данных, функции, операции, домены и индексы. Это делает PostgreSQL невероятно гибким и надежным. Среди прочего, он умеет создавать, хранить и извлекать сложные структуры данных. В некоторых примерах ниже вы увидите вложенные и составные конструкции, которые не поддерживаются стандартными СУБД.

Существует обширный список типов данных, которые поддерживает PostgreSQL. Кроме числовых, с плавающей точкой, текстовых, булевых и других ожидаемых типов данных (а также множества их вариаций), PostgreSQL может похвастаться поддержкой uuid, денежного, перечисляемого, геометрического, бинарного типов, сетевых адресов, битовых строк, текстового поиска, xml, json, массивов, композитных типов и диапазонов, а также некоторых внутренних типов для идентификации объектов и место-положения логов. Справедливости ради стоит сказать, что MySQL, MariaDB и Firebird тоже имеют некоторые из этих типов данных, но только PostgreSQL поддерживает их все:

- сетевые адреса;
- многомерные массивы;
- геометрические данные;
- данные в формате JSON.

– Веб-сервер программного средства разработан с применением архитектурного стиля REST, который представляет собой согласованный набор ограничений, учитываемых при проектировании распределённой системы.

– Для реализации веб-клиента был выбран подход SPA (Single Page Application – одностраничное веб-приложение) и один из лидирующих фреймворков ReactJs. Среди преимуществ ReactJs можно выделить: декларативный стиль кода, высокая скорость разработки, модульность, связывание данных, развитое сообщество.

Разработанное программное средство выполняет следующие функции:

- управление пользователями: добавление, редактирование, удаление, блокирование, смена пароля;
- управление группами: для каждой группы можно выставлять права доступа и привязывать пользователя к группе;
- управление подписками: отмена подписки, просмотр статуса, просмотр транзакций;
- добавление, редактирование, удаление каналов, программ, сезонов, эпизодов;
- начинать и заканчивать запись прямых трансляций;
- видео должно быть зашифровано технологией AES-128;
- трансляция при завершении должна конвертироваться в видео по запросу.

Список использованных источников:

1. Ньюмен, С. Создание микросервисов. / Сэм Ньюмен – СПб.: Питер, 2015. – 304 с.
2. Мэттью, Нейл. PostgreSQL. Основы / Нейл Мэттью, Ричард Стоунз – Москва: Символ, 2014. – 640 с.