

требованиями к системе, что делает невозможным использование готовых решений и требует разработки собственной CRM.

Разработка системы включает в себя основные этапы характерные для любого программного обеспечения. При этом на каждом этапе существуют нюансы, свойственные для разработки только данного типа систем. Первый и один из самых важных этапов, который проводится до проектирования будущей системы - это определение целей внедрения данной системы, а также описание текущих бизнес-процессов организации. После описания и оптимизации структуры бизнес-процессов организации наступает этап проектирования будущей системы. Дальнейшим этапом является непосредственно разработка CRM-системы. При реализации модулей системы разработчикам необходимо обратить внимание на безопасность и отказоустойчивость системы. Далее производится подготовка инфраструктуры будущей системы. Последующим этапом является этап внедрения.

Создание CRM системы сопряжено с многими трудностями и требует довольно больших затрат со стороны организации. При этом, если у бизнеса есть специфика, которая не позволяет использовать готовые решения, разработка данной системы оправдана и окупает затраты на ее реализацию, внедрение и поддержку за счет повышения уровня продаж, улучшения обслуживания клиентов и тем самым повышения их лояльности, а также сокращения издержек на управление процессами взаимодействия с клиентами, предоставляя возможность для дальнейшего роста и развития организации.

Список используемых источников:

1. Гринберг, Пол. CRM со скоростью света. Привлечение и удержание клиентов в реальном времени через Интернет/Пол Гринберг – СПб.: «Символ-Плюс» 2006. – 528с.
2. Эдриан Пейн Руководство по CRM. Путь к совершенствованию менеджмента клиентов/Эриан Пэйн – СПб.: «Гревцов Паблишер» 2007. – 384с.
3. Трофимов, Сергей. CRM для практиков/Сергей Трофимов – М.: «АвтоКод» 2006. – 304 с.

МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ УПРАВЛЕНИЯ ПЕРСОНАЛЬНЫМИ ФИНАНСАМИ

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

Чернянин Р.И.

*Пачинин В. И. – зав. кафедрой ИСиТ, к.т.н., доцент
Иванков А. С. – зам. нач. отдела разработки*

Обсуждаются результаты разработки мобильного приложения для управления персональными финансами, позволяющего вести учет и статистику расходов пользователя. Приложение накапливает информацию в виде, позволяющем анализировать персональные финансовые операции за любой промежуток времени.

В процессе проектирования был проведен анализ существующих аналогов и выявлены их преимущества и недостатки. На основе этого анализа было определено, какие функции необходимо реализовать в данном программном средстве. В экономическом разделе был проведен расчет экономической целесообразности разработки данного программного средства, из которого следует что рентабельность инвестиций проекта составит 175%.

В работе с использованием всех возможностей языка программирования Java были реализованы следующие функции:

- добавление/изменение/удаление финансовых операций;
- добавление/изменение/удаление расписания;
- изменение настроек приложения и системных данных (категории и валюты);
- просмотр отчетов о финансовых операциях;
- экспорт данных о финансовых операциях в файл.

При проектировании баз данных было использовано приложение DB Browser for SQLite. Браузер баз данных для SQLite представляет собой визуальный инструмент с открытым исходным кодом для создания, разработка и редактирование файлов баз данных, совместимых с SQLite. Он использует привычный табличный интерфейс, поэтому не нужно изучать сложные SQL-команды.

Для разработки самого приложения использовалась среда разработки Android Studio. Это интегрированная среда разработки для работы с платформой Android. Данная среда имеет ряд преимуществ над конкурентами:

- разработчиком данной IDE является компания Google, что означает более удобную интеграцию всех необходимых функций для новых версий Android;
- встроенный SDK (software development kit);
- удобный конструктор интерфейсов;
- информативные логи.

Таким образом, при помощи современных средств разработки было создано мобильное приложение для управления персональными финансами, которое успешно выполняет поставленные задачи.

Список использованных источников.

1. SqLitebrowser [Электронный ресурс]. – Режим доступа: <http://sqlitebrowser.org/>. – Дата доступа: 17.01.2018.

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.