

ОСНОВНЫЕ ПРИНЦИПЫ И ПОДХОДЫ К ПРОЕКТИРОВАНИЮ ПРОГРАММНОГО КОМПЛЕКСА VOIP ТЕЛЕФОНИИ

Вершицкий К. А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Половения С. И. – к.т.н., доцент

В работе рассматриваются основные принципы и подходы к реализации программного комплекса VoIP телефонии.

В современном мире передача информации посредством голоса и видео имеет важное значение. Рассмотрим следующие случаи из практики:

- передача информации между пользователями определенного приложения;
- общение между оператором и клиентом (заказ продукции, консультация пользователей).

Конкретными примерами использования комплекса могут являться: электронный консьерж, сервис по предоставлению медицинской помощи, служба технической поддержки, приложение для внутренней связи между предприятием.

Для качественного предоставления сервиса пользователям программный комплекс должен соответствовать следующим требованиям:

- масштабируемость (способность работать с большей нагрузкой при улучшении параметров дополнительных ресурсов);
- защищенность (сохранение конфиденциальных данных от третьего лица);
- возможность работы комплекса без привлечения 3-их сторон (серверное оборудование с данными пользователей может располагаться на собственных ресурсах);
- работа на широком диапазоне существующих устройств;
- модульность и возможность переиспользования решения (способность предоставления одного общего решения для нескольких заказчиков);
- расширяемость (возможность предоставления API (внешний интерфейс) для реализации бизнес-логики заказчика и возможности ее расширения);
- возможность подключения к телефонной сети общего пользования и другим сетям.

Архитектура программного комплекса представлена на рисунке 1.

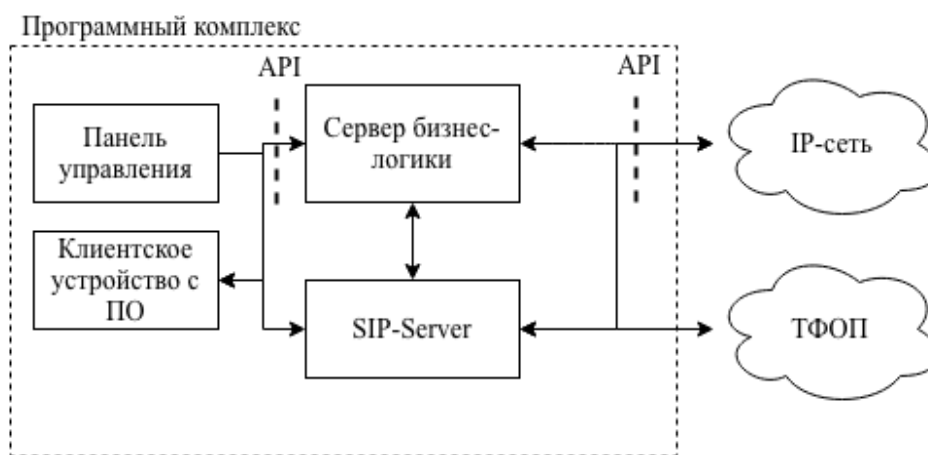


Рисунок 1 – Принципиальная схема программного комплекса

Клиентское устройство с ПО – мобильное устройство на операционной системе Android. В будущем планируется реализация для Web и IOS. ПО представляет собой приложение заказчика, которое использует функционал реализованной библиотеки.

Основной функционал библиотеки:

- низкоуровневое взаимодействие с микрофоном, динамиком, камерой;
- кодирование/декодирование данных;
- шифрование;
- представление информации (звучание звука, отображение видео);
- реализация транспортной функции на основе SIP-протокола [1].

Таким образом библиотека решает проблемы модульности и может быть максимально переиспользована для различных требований заказчика.

SIP-Server осуществляет следующие функции:

- выполнение проксирования и регистрации клиентов;
- настройка и управление соединением между двумя и более клиентами;

- согласование параметров и спецификаций для соединения каждой конечной точки;
- взаимодействие с внешними провайдерами для подключения к ТФОП.

Сервер бизнес-логики выполняет следующие функции:

- регистрация, авторизация пользователей на SIP-сервере;
- ограничение доступа пользователей, функционирование ролей;
- расчет основных характеристик и истории для каждого пользователя, создание отчетов;
- биллинг (расчёт длительности, стоимости звонков, отключение связи);
- предоставление программного интерфейса (API) для взаимодействия с внешними программными комплексами.

Панель управления – Web-приложение, предназначенное для управления функционалом сервера бизнес-логики и сервером SIP, мониторинга нагрузки, выгрузки отчетов, конфигурирование сервера.

Анализируя требования и архитектурное решение были выбраны технологии для каждого элемента комплекса.

1. Клиентское устройство – мобильное устройство на операционной системе Android. Покрывает 88% мобильных устройств. Язык разработки – Kotlin. Основная библиотека для работы с SIP- протоколом, кодирование/декодирования данных – PJSIP [2], шифрование – SSL, библиотека для шифрования – OpenSSL.

2. Сервер бизнес логики. Написан на платформе NodeJS на языке JavaScript. В качестве хранилища данных – нереляционная база данных MongoDB. Данный стек технологий выбран из-за простоты и высокой скорости разработки, при этом соответствуя всем рекомендациям по быстродействию. В качестве архитектурного стиля для API выбран REST, являющийся стандартом для распределенных систем. API необходимо для панели управления, клиентского ПО, а также предоставляется внешним системам заказчика.

3. SIP-Server. Используется Asterisk [3] – свободное решение компьютерной телефонии с открытым исходным кодом. Приложение работает на операционной системе Linux. Обладает всеми возможностями классической АТС, поддерживает множество VoIP-протоколов и предоставляет

4. Панель управление – Web-приложение, взаимодействующее с сервером бизнес- логики посредством REST-API. Основной фреймворк для написания панели – ReactJS.

Список использованных источников:

1. SIP-протокол [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Протокол_установления_сеанса – Дата доступа: 11.03.2020.
2. PJSIP [Электронный ресурс]. – Режим доступа: <http://www.pjsip.org> – Дата доступа: 10.09.2018.
3. Asterisk [Электронный ресурс]. – Режим доступа: <https://www.asterisk.org> – Дата доступа: 11.03.2020.