

Санкт-Петербургский государственный университет

Программная инженерия  
Кафедра Системного программирования

Слесарев Илья Дмитриевич

# Интеграция модуля чата в проект CrossCon

Курсовая работа

Научный руководитель:  
ст. преп. Немешев М. Х.

Рецензент:  
Разработчик "Ланит-Терком" Федотовский П. В.

Санкт-Петербург  
2017

SAINT-PETERSBURG STATE UNIVERSITY

Software Engineering  
Software Engineering department

Slesarev Ilya Dmitrievich

# Integration of the chat module into the project CrossCon

Course Work

Scientific supervisor:  
senior lecturer Nemeshev M. H.

Reviewer:  
Developer "Lanit-Terkom" Fedotovskiy P. V.

Saint-Petersburg  
2017

# Оглавление

<b>Введение</b>	<b>4</b>
<b>1. Обзор</b>	<b>6</b>
1.1. IRC . . . . .	6
1.2. SIP . . . . .	6
1.3. XMPP . . . . .	7
1.4. Matrix . . . . .	7
<b>2. Постановка задачи</b>	<b>10</b>
<b>3. Реализация</b>	<b>11</b>
3.1. Выбор языка для реализации . . . . .	11
3.2. Реализация библиотеки . . . . .	11
<b>Заключение</b>	<b>12</b>
<b>Список литературы</b>	<b>13</b>

# Введение

Организация конференций - это сложный, длительный и достаточно ресурсозатратный процесс. Логистика при работе с размещением участников, расположением помещений, потоками, регистрациями, возвратом денег – количество задач бесконечно. Необходимо предусмотреть все детали, чтобы конференция была подготовлена и проведена на высшем уровне.

Помимо подготовки к конференции, которая заключается в поиске места проведения мероприятия, рассылке приглашений слушателям и докладчикам и других организационных моментов, сложности существуют и непосредственно в процессе проведения. Например, зачастую организаторам необходимо связаться со слушателями и сообщить какие-либо изменения времени, аудитории и т.п., или слушателям хочется обсудить тему, близкую к докладу, скооперироваться для посещения определенного доклада.

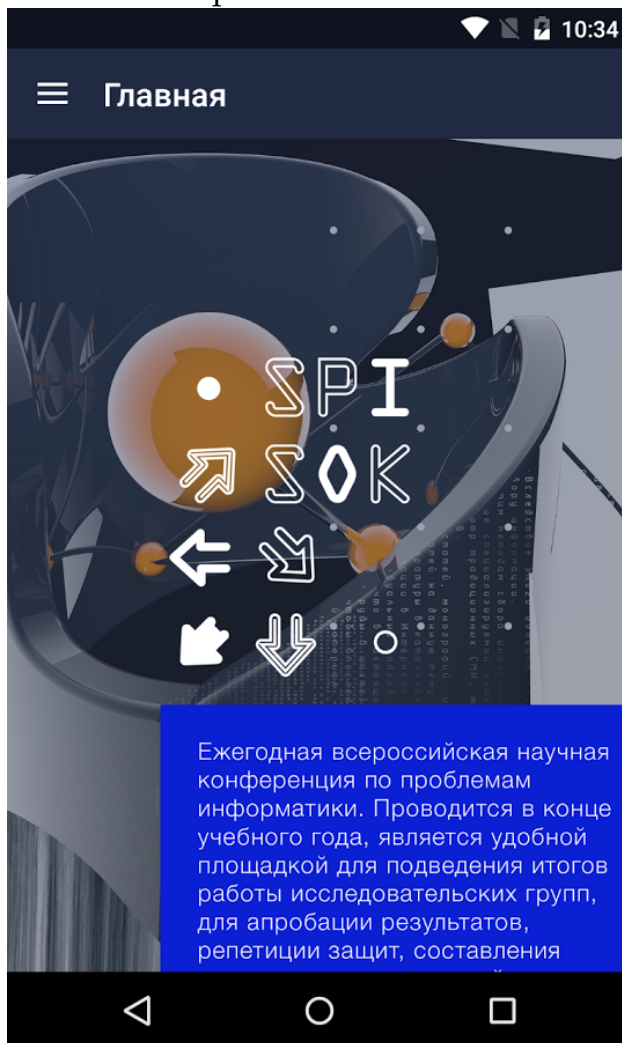
Организаторы конференций по-разному решают данные проблемы. У некоторых есть веб сервисы для отслеживания расписания и мессенджеры/страницы в социальных сетях для связи внутри конференции, у других для того, чтобы узнать расписание докладов, нужно заглянуть в листовку. Для решения подобных задач в организации конференций в компании ЛАНИТ - Терком было разработано кроссплатформенное приложение CrossCon.

CrossCon – мобильное приложение, разработанное для научной конференции Список, с поддержкой платформ Android, iOS и WinPhone. Однако также реально сделать его адаптивным под другие конференции в таком виде, чтобы для применения требовались минимальные усилия.

CrossCon предоставляет следующие возможности:

- Лента новостей
- Просмотр расписания на каждый день
- Описаний секций и пленарных докладов

Рис. 1: Приложение СПИСОК



Реализовано приложение на языке C# с использованием Xamarin.Forms и облачных технологий Microsoft Azure.

Таким образом большая часть задач, которые ставятся при организации конференции, была решена. Однако остались те пункты, которые связаны с контактом между участниками и организаторами. Зачастую внутри конференции необходима оперативная связь, если же мессенджеры установлены не у всех, то приложения для конференции (в част. СПИСОК) скорее всего скачаны у большинства посетителей мероприятия.

Так появилась идея реализации чата внутри приложения для конференций.

# 1. Обзор

На данный момент существует большое количество протоколов для передачи текстовых данных, которые можно было бы использовать для реализации необходимого инструмента:

- IRC (Internet Relay Chat)
- SIP (Session Initiation Protocol)
- (XMPP) - мессенджер Jabber
- Matrix

Каждый из них обладает своими плюсами и минусами.

## 1.1. IRC

IRC - протокол прикладного уровня для обмена сообщениями между пользователями в режиме реального времени. Является клиент-серверной системой.

Протокол достаточно прост, однако излишне наполнен различными функциями, которые не нужны для простой связи в приложении. Впервые описан в RFC 1459. [3] Согласно спецификации протокола, IRC-сеть - это группа серверов, соединенных между собой. Таким образом, если произойдет обрыв связи между IRC - серверами, то в результате некоторое количество пользователей отключится от канала.

## 1.2. SIP

SIP - протокол установления сеанса. Это протокол передачи данных, который описывает способы установления и завершения обмена мультимедийным контентом (в част. мгновенными сообщениями).

SIP - один из протоколов, лежащих в основе Voice over IP. Описан в RFC 3261. [4] SIP очень похож на протокол HTTP, используемый для Web приложений, или на SMTP (обмен почтовыми сообщениями).

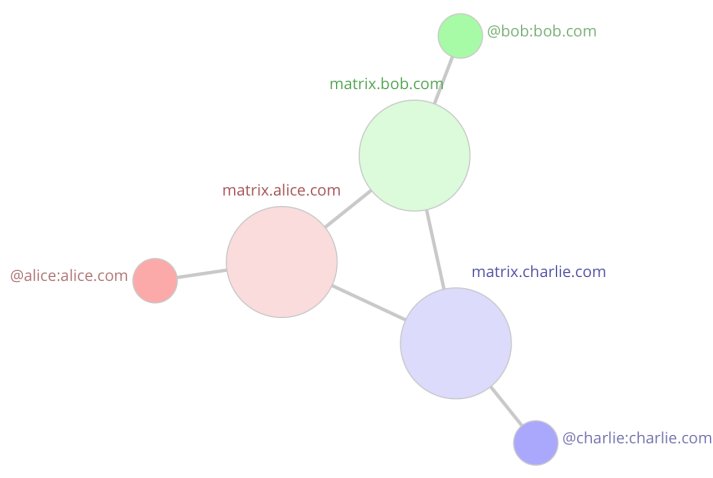


сообщения, так и сообщения других форматов (картинки, голосовые сообщения, файлы, видео). Осуществляется поддержка как групповых, так и частных чатов (в том числе поддерживается модерация и разделение прав). Протокол распространяется в открытом виде, то есть открыт исходный код как серверов (Synapse, Pallium, Rume и других), так и клиентов.

Основными плюсами протокола Matrix является особенная архитектура (клиент-сервер-сервер-клиент) и федеративность. Архитектура данного протокола особенно важна для мобильных устройств, потому что в отличие от полностью децентрализованных систем, на клиента не ложится вся нагрузка по построению и поддержанию связей с близлежащими узлами, что экономит их трафик и заряд батареи, ведь необходимо поддерживать связь только со своим сервером.

Полезность федеративности покажу на следующей ситуации. Пользователь с мобильного клиента подключился к каналу. А сервер, через который "ходил" пользователь, отказал в работе, тогда система продолжит работать, так как нет центрального сервера. На данный момент пользователь регистрируется на конкретном сервере (Однако разработчиками протокола планируется и эту часть сделать непривязанно к серверу), а при дальнейшем использовании "ходит" через другие сервера.

Рис. 3: Архитектура на примере одной комнаты



На рисунке (Рис. 3) изображено то, как каждый клиент связан с



определенным сервером и отправляет сообщения через него.

На основе анализа приведенных выше протоколов была составлена следующая таблица (Табл. 1), где плюсами и минусами указаны положительные и отрицательные стороны того или иного протокола соответственно. В следствие чего, для реализации чата в приложении для работы с конференциями был выбран протокол Matrix.

Таблица 1: Представленные протоколы

Протокол	Групповые чаты	Простота реализации	Федеративность	Архитектура
IRC	+	+	-	клиент-сервер
SIP	+	-	-	клиент (он же сервер)
XMPP	+	-	+	клиент-сервер
Matrix	+	+	+	клиент-сервер- сервер-клиент

## 2. Постановка задачи

Целью данной работы является реализация инструмента для связи с докладчиками и слушателями внутри приложения. Для ее достижения поставлены следующие задачи:

- выбор протокола
- реализация библиотеки для работы с протоколом
- интеграция написанной библиотеки в приложение
  - реализация одной общей комнаты конференции
  - реализация чата для каждой секции

## 3. Реализация

### 3.1. Выбор языка для реализации

Приложение CrossCon реализовано на языке C# [5] с использованием платформы Xamarin.Forms. Это кроссплатформенное приложение, и поэтому для поддержания платформ Android, iOS и WinPhone необходимо было реализовать библиотеку для работы с протоколом чата также в кроссплатформенном виде. В виду этого было принято решение разрабатывать библиотеку на языке C# с использованием платформы .Net Core.

.NET Core — это универсальная платформа разработки, которая поддерживается корпорацией Майкрософт и сообществом .NET на сайте GitHub. Она является кроссплатформенной, поддерживает Windows, Mac OS и Linux и может использоваться на устройствах, в облаке, во внедренных системах и в сценариях Интернета вещей.[2]

### 3.2. Реализация библиотеки

Выбор протокола Matrix был удачным с точки зрения написания библиотеки. Открытое и полное API позволяет полностью написать свой клиент для работы с протоколом. [1] Несмотря на относительную "молодость" протокола, уже было реализовано несколько библиотек на других языках программирования, но протокол достаточно динамичен и в некоторых реализациях, уже сейчас есть методы, которые не поддерживаются в виду устаревания их в официальном API разработчиков Matrix.

## Заключение

В ходе выполнения работы были получены следующие результаты. Написана библиотека для работы с протоколом, поддерживается:

- отправка и чтение сообщений
- некоторые модераторские возможности
  - создание комнаты
    - \* получение параметров и участников комнаты
    - \* общее состояние комнаты
  - запрет на вход в комнату
  - приглашение в комнату и присоединение к комнате
  - принудительное завершение сессии пользователя
  - возможность ”забыть” комнату

Расширение функций библиотеки было упрощено тем, что при текущем проектировании имеется возможность без изменения написанного кода, добавляя минимум дополнительного кода, добавлять новые функции для работы с комнатами и пользователями.

Также реализован прототип работы с библиотекой на платформе .Net Core.

## Список литературы

- [1] Matrix. Client-Server API. — URL: [http://matrix.org/docs/spec/client\\_server/r0.2.0.html](http://matrix.org/docs/spec/client_server/r0.2.0.html).
- [2] Microsoft. Документация по .Net Core // Документация Microsoft. — URL: <https://docs.microsoft.com/ru-ru/dotnet/articles/core/> (дата обращения: 24.05.2017).
- [3] RFC. Internet Relay Chat Protocol. — URL: <https://www.ietf.org/rfc/rfc1459.txt>.
- [4] RFC. SIP: Session Initiation Protocol. — URL: <https://www.ietf.org/rfc/rfc3261.txt>.
- [5] Richter Jeffrey. CLR via C# (4th Edition). — 2006.
- [6] Wikipedia. XMPP // Extensible Messaging and Presence Protocol. — URL: <https://en.wikipedia.org/wiki/XMPP>.