

Смирнов Максим Алексеевич, магистрант, Уфимский университет науки и технологий, г. Уфа

## АРХИТЕКТУРА РЕГИОНАЛЬНОГО АГРЕГАТОРА НОВОСТЕЙ В TELEGRAM

**Аннотация.** В данной статье рассматривается архитектура Telegram-бота, который предназначен предоставления пользователям актуальных данных по выбранному ими региону. Приложение построено на клиент-серверной архитектуре, где клиентская часть (Telegram-интерфейс) взаимодействует с сервером Telegram Api который взаимодействует с сервером чат-бота, который обрабатывает запросы, и предоставляет необходимые данные. Не осталось без внимания распределение нагрузки между компонентами, безопасности взаимодействия через Telegram API и механизмами получения актуальных данных из внешних источников. Описанная архитектура демонстрирует подход к созданию региональных агрегаторов в мессенджерах.

**Annotation.** This article deals with the architecture of Telegram-bot, which is designed to provide users with up-to-date data on the region of their choice. The application is built on a client-server architecture, where the client part (Telegram interface) interacts with the Telegram Api server, which interacts with the chatbot server, which processes requests and provides the necessary data. Load distribution between components, security of interaction via Telegram API and mechanisms for obtaining actual data from external sources were not left unattended. The described architecture demonstrates an approach to creating regional aggregators in messengers.

**Ключевые слова:** Telegram-bot, агрегатор, архитектура, клиент, сервер, API.

**Keywords:** Telegram-bot, aggregator, architecture, client, server, API.

Чат-бот, обеспечивающий потребителей своевременными сведениями по заданному региону, функционирует на клиент-серверной архитектуре (Рисунок 1).

Клиент-серверная архитектура – это механизм взаимодействия компьютерных систем, в котором для получения различных данных, услуг и информации пользовательские устройства делают запрос к серверам.

В такой архитектуре серверы представляют собой специальные эффективные устройства, преподносящие определенные виды услуг или ресурсы, наподобие базы данных, различных онлайн-сервисов, электронной почты и т.п.

Потребительские устройства, к примеру, могут быть как компьютерами, смартфонами, так и различными другими устройствами, которые в процессе работы обращаются к серверам для получения необходимой информации или какой-либо услуги.

Данная архитектура помогает равномерно разделить нагрузку между разными устройствами и гарантировать более результативное управление ресурсами и правильную обработку запросов.



Рисунок 1 – Клиент-серверная архитектура telegram бота

Сервер API Telegram:

1. Отображает ресурсы и различные услуги, используемые клиентами. Например, файлы, базы данных, приложения и т.д.;
2. Обрабатывает всевозможные запросы, приходящие от пользователей, с помощью проведения различных операции и предоставляет нужные данные и ресурсы клиентам;

3. Может также гарантировать безопасность, проверку подлинности пользователя и управлять полноценным доступом клиентов к нужным ресурсам.

Сервер чат-бота:

1. Изначально получает обращение от сервера Telegram API, далее обрабатывает полученный запрос и отправляет ответ.

Клиент (приложение Telegram):

1. Формирует запросы к серверу для открытого входа к ресурсам и услугам, которые предоставляет сервер;

2. Отправляет полученные запросы на серверы Telegram API для получения и отправки сообщений;

3. Осуществляет обработку информации, валидирует ее и предоставляет результат для пользователя;

4. Отображает интерфейс для сотрудничества с пользователем.

База данных:

1. Используется для хранения различной информации, редактирования и извлечения данных, нужных для работы чат-бота.

Подводя итоги, данный сервер API Telegram дает нужные ресурсы и услуги, а пользователь использует их, обращаясь к этому же серверу для обеспечения высокой функциональности и нужной информации.

Рассмотрим подробную архитектуру приложения, приведенную на Рисунок 2.

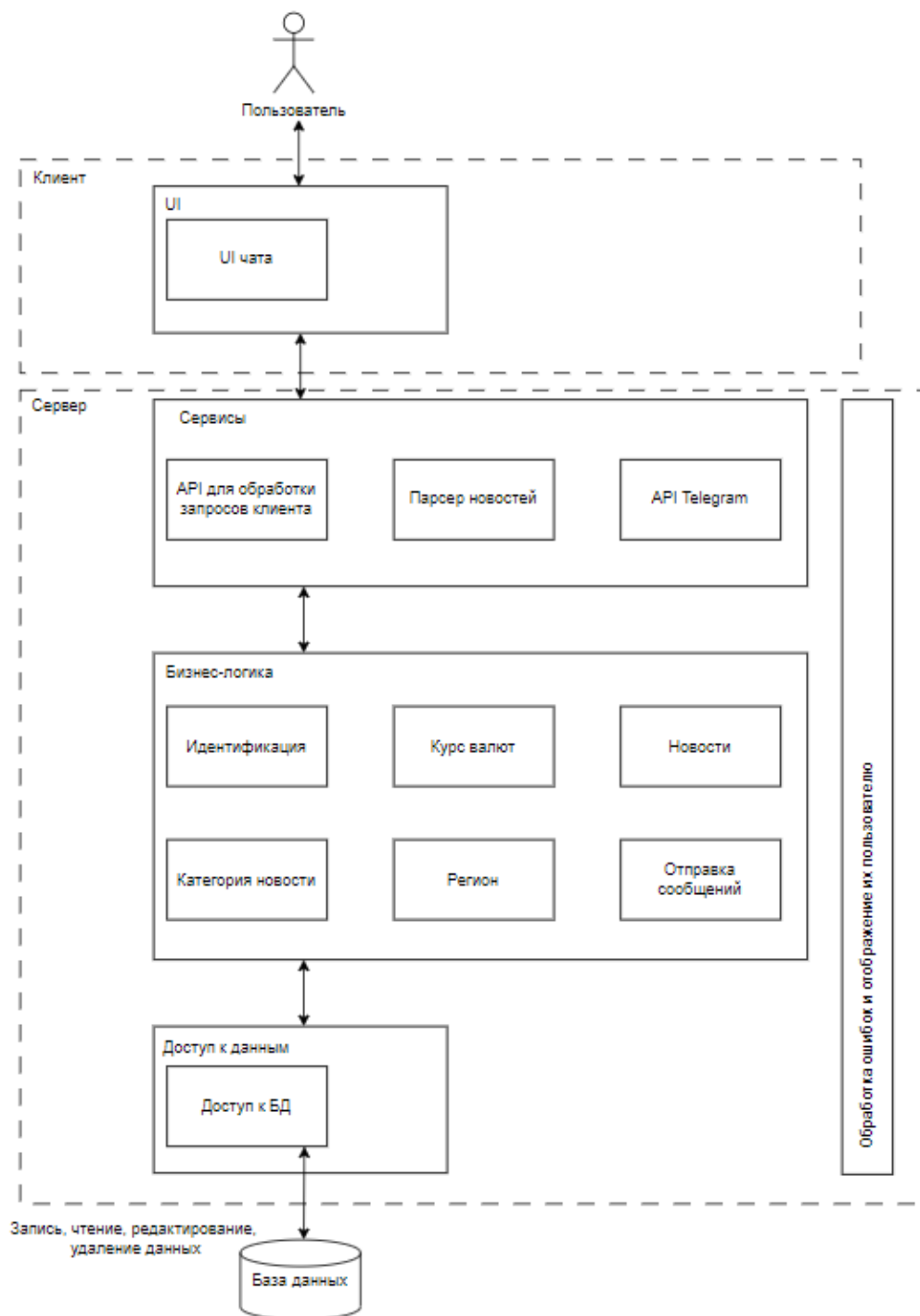


Рисунок 2 – Детальная архитектура приложения

Архитектура состоит из таких важных составляющих, как:

Клиентская часть:

1. User Interface чата:

Интерфейс пользователя может быть в виде различных элементов управления: кнопки, поля ввода, меню и прочие, помогающие клиенту легко и удобно взаимодействовать с ботом и создавать нужные запросы;

Взаимодействие с ботом через чат-интерфейс Telegram.

Серверная часть:

1. Обработчик запросов (Request Handler):

Несет ответственность за прием и оценку поступающих сообщений от пользователя;

Распознавание запроса от пользователя, связанного с актуальными данными о запрашиваемом регионе.

2. Модуль для работы с данными по региону:

Интеграция с геолокационными сервисами:

Для получения пользователями своевременной информацией о регионе, чат-бот может использовать API геолокационных сервисов для получения данных о настоящем местоположении пользователя

3. Модуль интеграции внешних источников:

API для получения данных:

Взаимодействует с внешними службами, такие как курсы валют, чтобы получать актуальные данные на настоящее время;

Парсер новостей:

Взаимодействует с внешними источниками информации, такими как новостные агентства и сервисы курса валют, чтобы получать своевременную и точную информацию по региону;

Взаимодействует с внешними источниками данных для того, чтобы получать свежие новости по данному местоположению пользователя.

Обработка данных:

Отвечает за обработку и группировку всех полученных данных для дальнейшего предоставления услуг пользователю.

4. Бизнес-логика (Business Logic):

Обработка запросов, полученных от клиента и создание ответов на основе исходных данных;

Принятие решений по поводу того, какие именно данные предоставить пользователю.

#### 5. База данных:

Для хранения важных данных, таких как: имя пользователя, регион, актуальные новости, все это необходимо для обеспечения пользователей своевременной информацией по запросу и региону.

В описании архитектуры клиент-серверного приложения тщательно изучим главные модули приложения в первом приближении:

#### 1. Интерфейс пользователя (UI):

Включает в себя все составные части, нужные для взаимодействия пользователя с приложением, это могут быть экраны, поля ввода и другие элементы интерфейса;

#### 2. Бизнес-логика:

Этот модуль предоставляет обработку данных, логику данного приложения и взаимодействие с внешними источниками информации, различными службами или базами данных;

#### 3. Сетевой слой:

Гарантирует и поддерживает взаимосвязь между клиентской частью и серверной;

#### 4. База данных:

Модуль, отвечающий за хранение информации, которая используется в приложении;

#### 5. Модуль обработки команд (Command Processing Module):

Данный модуль обеспечивает анализ различных команд от потребителей и выполнение соответствующих действий.

Содержит логику обработки команд, таких как распознавание, выполнение операций совместно с базой данных и формирование требуемых ответов.

Таким образом, в чат-боте, созданном для получения пользователями актуальной информации по заданному ими региону, используется клиент-серверная архитектура. Клиент включает в себя пользовательский интерфейс, тем временем сервер использует разные API для получения данных, бизнес логику.

## Литература

1. Воробьев А.В. Интерактивная компьютерная модель для прогноза и анализа полярных сияний / А. В. Воробьев, А. А. Соловьев, В. А. Пилипенко, Г. Р. Воробьева // Солнечно-земная физика. - 2022. - Т. 8, № 2. - С. 93-100. - DOI 10.12737/szf-82202213. - EDN YLVVDW.
2. Воробьев, А. В. Индуктивный метод восстановления временных рядов геомагнитных данных / А. В. Воробьев, Г. Р. Воробьева // Труды СПИИРАН. - 2018. - № 2(57). - С. 104-133. - DOI 10.15622/sp.57.5. - EDN YVGDKY.
3. Воробьев, А. В. Автоматизированный анализ невозмущенного геомагнитного поля на основе технологии картографических веб-сервисов / А. В. Воробьев, Г. Р. Шакирова // Вестник Уфимского государственного авиационного технического университета. - 2013. - Т. 17, № 5(58). - С. 177-187. - EDN RPWYQF.
4. Vorobev, A. V. Web-Based Information System for Modeling and Analysis of Parameters of Geomagnetic Field / A. V. Vorobev, G. R. Shakirova // Procedia Computer Science : 1st, Angrek Campus, Jakarta, 24-26 августа 2015 года. - Angrek Campus, Jakarta, 2015. - P. 73-82. - DOI 10.1016/j.procs.2015.07.339. - EDN VAOAJZ.
5. Клиент-серверная архитектура в картинках[Электронный ресурс]: Хабр.– URL: <https://habr.com/ru/articles/495698/>

## Literature

1. Vorobyev, A.V. Interactive computer model for forecasting and analysis of polar auroras / A.V. Vorobyev, A.A. Solovyov, V.A. Pilipenko, G.R. Vorobyeva // Solar-Terrestrial Physics. - 2022. - T. 8, № 2. - C. 93-100. - DOI 10.12737/szf-82202213. - EDN YLVVDW.
2. Vorobyev, A. V. Inductive method of recovery of geomagnetic data time series / A. V. Vorobyev, G. R. Vorobyeva // Proceedings of SPIIRAN. - 2018. - № 2(57). - C. 104-133. - DOI 10.15622/sp.57.5. - EDN YVGDKY.
3. Vorobyev, A. V. Automated analysis of the unperturbed geomagnetic field based on the technology of cartographic web services / A. V. Vorobyev, G. R. Shakirova // Bulletin of Ufa State Aviation Technical University. - 2013. - T. 17, № 5(58). - C. 177-187. - EDN RPWYQF.
4. Vorobev, A. V. Web-Based Information System for Modeling and Analysis of Parameters of Geomagnetic Field / A. V. V. Vorobev, G. R. Shakirova // Procedia Computer Science : 1st, Anggrek Campus, Jakarta, August 24-26, 2015. - Anggrek Campus, Jakarta, 2015. - P. 73-82. - DOI 10.1016/j.procs.2015.07.339. - EDN VAOAJZ.
5. Client-server architecture in pictures[Electronic resource]: Hubr.- URL: <https://habr.com/ru/articles/495698/>