Вестник Сыктывкарского университета.
Серия 1: Математика. Механика. Информатика.
Выпуск 3 (40). 2021

УДК 004.42 DOI: 10.34130/1992-2752_2021_3_41

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ЧАТ-БОТА ДЛЯ ПРЕДСТАВЛЕНИЯ РАСПИСАНИЯ В СОЦИАЛЬНОЙ СЕТИ

Ю. В. Гольчевский, А. В. Непеин

В данной работе представлено исследование проблемы доставки расписания участникам образовательного процесса на базе проектирования и разработки чат-бота для социальной сети. В ходе работы проведено моделирование бизнес-процесса составления расписания, выполнены анализ платформ для реализации диалоговых интерфейсов (чат-ботов), проектирование и разработка программной архитектуры и базы данных, а также представлены некоторые аспекты реализации интерфейса межсистемного взаимодействия.

Ключевые слова: чат-бот, расписание занятий, учебное заведение, программная архитектура, социальная сеть.

Введение

Поведение современного потребителя информационных сервисов характеризуется желанием быстро и качественно получать информацию, прилагая для этого минимум усилий.

© Гольчевский Ю. В., Непеин А. В., 2021.

На смену чтению информационных текстов, телефонным разговорам или личным встречам с консультантами приходит обмен короткими сообщениями, что способствует росту популярности использования чат-ботов. Например, согласно исследованию, приведенному в журнале «ПЛАС», объем рынка чат-ботов, поддерживающих технологию понимания естественного языка, в России к 2023 году достигнет 33 миллиардов рублей [1].

В широком смысле чат-бот — это программа, которая интерактивно имитирует человеческую речь (устную или письменную) и позволяет общаться с цифровыми устройствами так, как с людьми [2]. Чат-бот может быть простой программой, отвечающей на простой запрос или нажатие кнопки, или достаточно сложной системой, обучающейся и развивающейся по мере накопления информации. Он также может быть реализован для некоторой социальной сети в виде скрипта, получающего уведомления о новых событиях и обрабатывающего их определенным образом.

Спектр использования чат-ботов весьма широк. Например, бот для заказа пиццы (https://telegram.me/papajohns_namebot), боты, помогающие найти маршрут или выбрать подарок (https://dev.by/news/5-uspeshnyy-keysov-primeneniya-chat-botov-ili-kak-sekonomit-million). Чат-боты могут оптимизировать повторяющиеся бизнес-задачи, например первоначальный подбор персонала или интеграция информационных систем (https://www.cossa.ru/trends/190984/), или просто оказывать положительное терапевтическое воздействие на людей, общаясь с ними (https://woebothealth.com/).

Исследования проблем разработки и использования чат-ботов различной сложности достаточно широко представлены в научных изданиях. Например, в работе [3] приведены ре-

зультаты исследования проблемы разработки функционального и надежного новостного чат-бота и использования его в качестве альтернативного средства доступа к информации в кризисной ситуации, когда потребности аудитории в своевременной информации быстро возрастают. В [4] авторами представлен процесс проектирования, функционального описания разработанной архитектуры и деталей реализации системы поддержки клиентов для аэропорта Венеции. Целью являлась разработка ядра, способного работать на базе использования различных парадигм, от речи до сенсорных экранов и через различные пользовательские интерфейсы (мобильные телефоны, стационарные установки и др.) В работах [5; 6] представлены исследования эффективного применения чат-ботов для бизнеса, а в [7] – для проверки безопасности зданий.

Можно констатировать, что в последнее время чат-боты начали активно применяться в сфере образования. Например, для дополнения существующих услуг академического консультирования, ответов студентам на вопросы по материалам и логистике учебных курсов, что отражено в работах [8; 9]. Исследователи отмечают большой потенциал использования чат-ботов для улучшения процесса обучения и результатов обучения [10].

Одним из направлений использования чат-ботов для учебных заведений может являться решение задачи представления расписания учебных занятий. Это и определило цель данной работы – представить исследование проблемы доставки расписания участникам образовательного процесса на базе проектирования и разработки чат-бота для социальной сети «ВКонтакте». Данная работа является продолжением исследований о способах представления расписания, опубликованных в работах [11; 12]. Актуальность выбранной темы про-

диктована необходимостью в улучшении пользовательского опыта при использовании сервисов учебного заведения пользователями мобильных устройств, а также повышении понимания проблемы взаимодействия людей и чат-ботов, на актуальность которой исследователи обращают пристальное внимание в последнее время, в частности в работе [13].

Анализ бизнес-процесса и выбор инструментария

Бизнес-процесс составления расписания занятий может сильно варьироваться в зависимости от конкретного учебного заведения, однако можно смоделировать наиболее общую модель, которая представлена на рис. 1.

Крайне важной частью этого процесса является подпроцесс «Публикация расписания». В нем определяется то, какими способами составленное расписание доставляется участникам образовательного процесса. Существует множество различных способов, каждый из которых обладает своими достоинствами и недостатками, что обсуждается в работе [12]. Распространенными способами являются загрузка PDF-файлов на официальный сайт и размещение электронной таблицы в информационной системе учебного заведения. Частыми проблемами таких способов являются сложная структура расписания, смешение различных типов событий (лекций, зачетов и т. д.), в таблице может появляться много пустых строк, из-за которых приходится дольше искать нужную информацию, что особенно актуально для устройств с небольшими экранами. Как правило, расписание выводится сразу на всю неделю, хотя зачастую необходимо посмотреть его на конкретный день, например на сегодня или завтра. Визуальная разрозненность элементов расписания (часто номер и время начала занятия находятся слишком далеко от описания предмета), что может потребовать совершать много лишних движений.

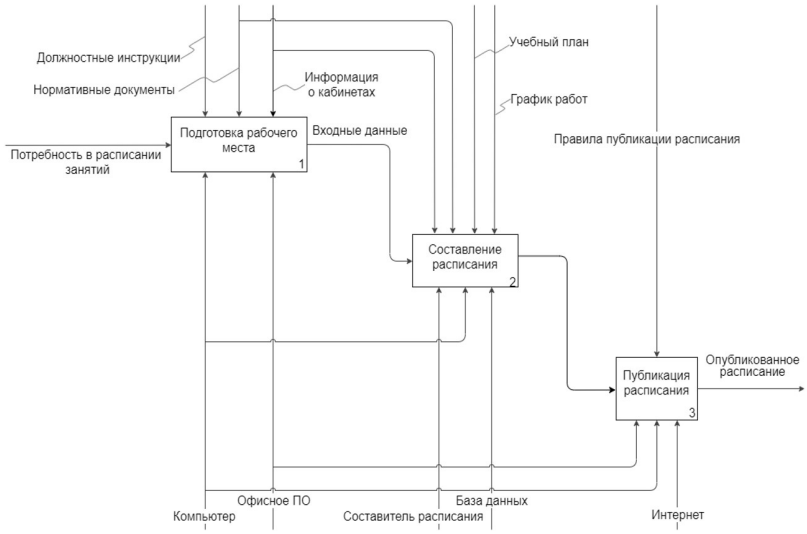

Рис. 1. SADT-диаграмма бизнес-процесса составления расписания занятий

Лучшим решением было бы расположение всех столбцов по вертикальной оси в виде упорядоченного по времени списка. Кроме того, актуальной проблемой может быть не всегда достаточная поддержка мобильных устройств и небольшие размеры экрана самого устройства, что затрудняет просмотр больших таблиц. Все это в совокупности может сделать опыт для пользователей мобильных устройств малоприятным. В результате можно выделить несколько основных требований, которые необходимо учитывать при реализации системы доставки расписания. Кроме стандартных требований постоянной актуальности и доступности, можно дополнительно выделить следующие:

1. Получение расписания должно быть максимально простым и быстрым; интерфейс должен быть доступен и удобен для пользователей, использующих мобильные устройства.

2. Необходима реализация учета различных особенностей расписания, например, системы числитель/знаме-

натель, блоковой системы проведения занятий, т. е. вести учет тип недели.

3. Интерфейс, предоставляющий расписание, должен предоставлять последнее состояние расписания на случай непредвиденного нарушения работоспособности сервера.

В качестве программной оболочки для реализации диалогового интерфейса было решено использовать платформу для чат-ботов социальной сети «ВКонтакте» и мессенджера «Телеграм». Основными причинами такого выбора явились отсутствие необходимости в проектировании собственного графического пользовательского интерфейса, ограничений в выборе технологий, простота распространения и обновления программного продукта, результаты анализа аудитории «ВКонтакте» по возрастным группам.

Анализ популярности платформ также проводился на базе информации ресурса MediaScope (https://mediascope.net). Кроме того, такой выбор позволяет охватить пользователей всех устройств, поскольку приложение доступно в браузере.

Хотелось бы отметить, что в данной работе не ставилась цель заменить традиционные способы доставки расписания. Предлагаемый чат-бот может служить полезным дополнением к традиционному табличному или иным способам представления расписания, ориентированным на поколение, привыкшее достаточно много времени проводить в социальных сетях и желающее получать от них не только развлечения, но и полезные сервисы.

При разработке использовался такой инструментарий, как платформа Miro (https://miro.com/) для проектирования диалогов, язык программирования TypeScript и программная платформа NodeJS. Для хранения данных применялись ре-

ляционная СУБД PostgreSQL и хранилище «ключ-значение» Redis.

Для легкого переноса и установки приложения на удаленном сервере, а также локальной разработки применялся инструмент по работе с контейнерами Docker (https://www.docker.com/). В качестве редактора кода использовался Visual Studio Code от Microsoft.

Выявление функциональных требований и проектирование архитектуры программной системы

Для выявления функциональных требований для каждого компонента системы были построены диаграммы прецедентов, представленные на рис. 2, 3.

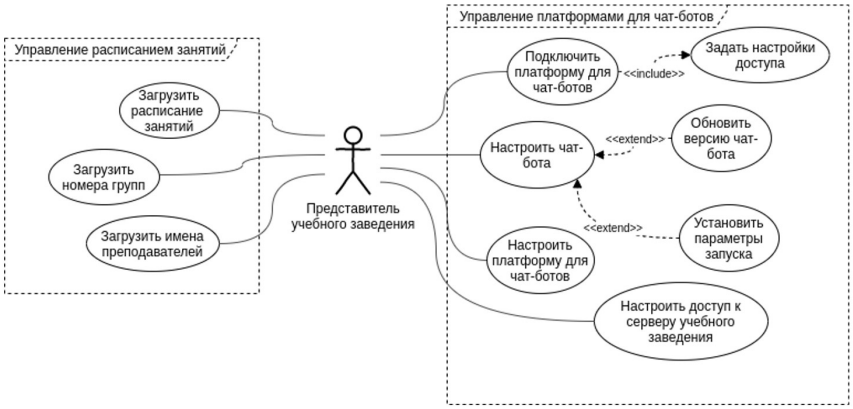

Рис. 2. Функциональность web-интерфейса управления чат-ботом

В дополнение к диаграммам прецедентов был составлен бэклог продукта, позволивший упорядочить перечень всех запросов, что дополнило результаты проведенного MOSCOW-анализа.

Одним из требований к проектируемой системе является возможность без особых усилий распределить компоненты системы по нескольким серверам для улучшения производительности всей системы в целом.

На рис. 4 представлена общая структура проектируемой системы. Можно выделить четыре подсистемы, каждая из которых выполняет свою собственную задачу: работа с каналами связи, управление и обработка событиями, обработка запросов к API от внешних пользователей.

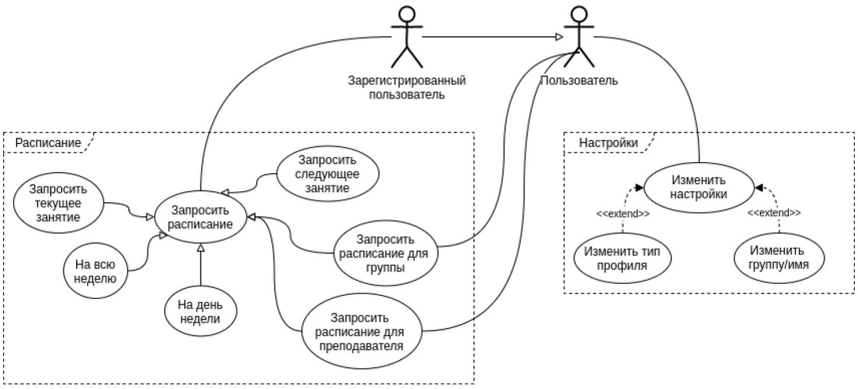

Рис. 3. Спецификация функциональности чат-бота

Задача первой подсистемы заключается в преобразовании входящих и исходящих событий в общий и специфичный вид соответственно, что позволяет выполнять обработку события вне зависимости от типа канала связи. Для этого были разработаны специальные «коннекторы», которые позволяют выполнять преобразования в обе стороны для конкретного канала связи. После преобразования событие попадает в подсистему управления событиями, которая добавляет его в очередь типа FIFO на последующую обработку.

Очереди реализуются с помощью библиотеки Bull, которая использует Redis для хранения данных событий, что дает возможность горизонтального масштабирования системы, поскольку доступ к Redis может осуществляться с любого сервера.

Существует также очередь исходящих событий, обрабатываемых коннекторами. Здесь происходит преобразование

события из общего вида в специфичный для определенного канала связи.

Подсистема, обрабатывающая входящие события, представляет собой набор обработчиков, каждый из которых привязан к конкретному учебному заведению, и сервис, предоставляющий API для работы с источниками данных. Среди таких источников могут быть: Redis, выступающий в качестве кеша, внутренняя база данных или внешнее API для работы с базой данных учебного заведения.

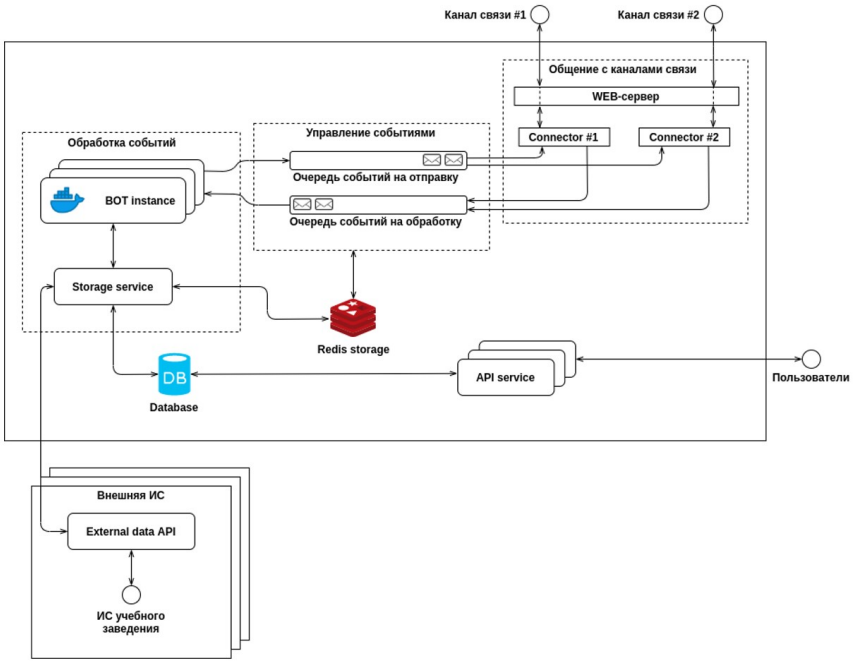

Рис. 4. Программная архитектура системы

Обработчик события представляет собой экземпляр чат-бота, настроенного под конкретное учебное заведение. Под настройкой понимается передача в него параметров, связанных с интеграцией с источниками данных, каналами связи и т. д. Сам чат-бот – это несложная система, которая занимается обработкой входящего сообщения с целью понять его смысл, после чего совершает связанные с ним действия, на-

пример записывает данные пользователя в хранилище, отправляет ответное сообщение в виде текста, кнопок и т. д. Для этого был разработан фреймворк, общая схема работы которого представлена на рис. 5 в виде диаграммы последовательностей.

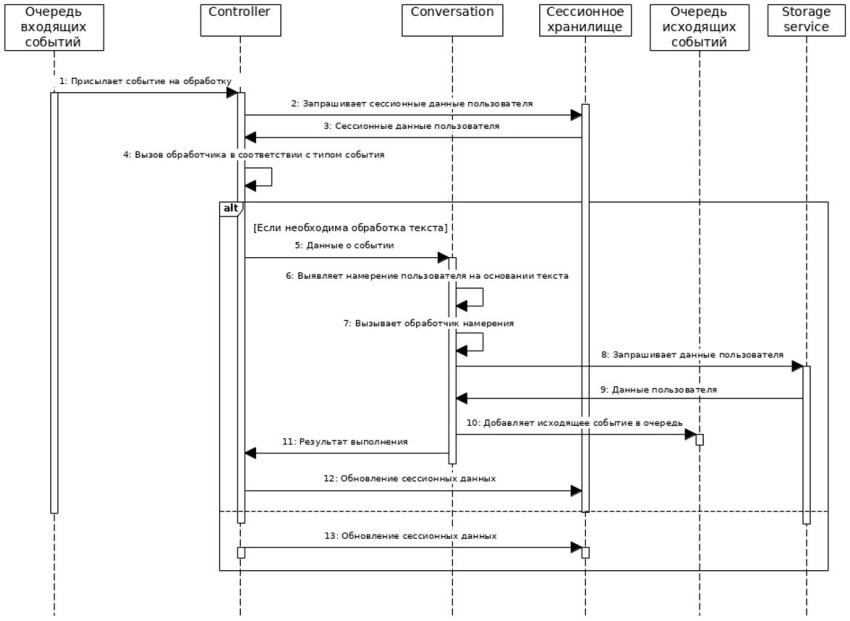

Рис. 5. Схема работы фреймворка для обработки события

Подсистема обработки запросов к API от внешних пользователей является стандартным серверным приложением, предоставляющим API к внутренним данным. Это может использоваться для получения данных со стороны web-интерфейса для управления чат-ботом, а также для интеграции с информационной системой учебного заведения. На схеме присутствует также внешняя система, связанная с сервисом, предоставляющим данные для чат-бота, которой является информационная система учебного заведения и с которой имеется возможность обмениваться данными через специальный программный интерфейс.

Для решения проблемы интеграции данных между проектируемой системой и информационной системой учебного заведения используется программный интерфейс, позволяющий системе получать актуальные данные прямо из базы данных учебного заведения. Его проектирование велось в соответствии со спецификацией OpenAPI 3.0, которая позволяет описать все доступные конечные точки и операции на них, параметры на вход и выход, методы аутентификации, а также некоторую дополнительную информацию.

Получившийся интерфейс состоит из пяти методов:

1. /check/teacher/teacher_name – позволяет проверить существование определённого преподавателя по его имени;
2. /check/group/group_name – позволяет проверить существование группы по её номеру;
3. /schedule – позволяет получить расписание звонков;
4. /schedule/teacher/teacher_name/week_offset – позволяет получить расписание преподавателя на неделю со смещением на определенное количество недель;
5. /schedule/group/group_name/week_offset – позволяет получить расписание группы по ее номеру на неделю со смещением на определенное количество недель;
6. /schedule/meta – позволяет получить метаданные о расписании.

Спецификация также позволяет указать единую точку входа, откуда будут вызываться все эти методы, например https://campus.syktsu.ru/api. Представим один из методов — получение расписания для преподавателей (рис. 6). Пример представлен на языке разметки YAML.

```
1 '/schedule/teacher/{teacher_name}/{week_offset}':
2   get:
3     responses:
4       '200':
5         content:
6           '*/*':
7             schema:
8               type: array
9               items:
10                $ref: '#/components/schemas/teacher_day'
11      '404':
12        content:
13          '*/*':
14            schema:
15              oneOf:
16                - type: array
17                  items:
18                    type: string
19                - type: boolean
20                  default: false
21      '503':
22        content:
23          '*/*':
24            schema:
25              type: boolean
26                default: false
27    parameters:
28      - name: teacher_name
29        in: path
30        required: true
31        schema:
32          type: string
33      - name: week_offset
34        in: path
35        required: true
36        schema:
37          type: integer
```

Рис. 6. Описание метода для получения расписания преподавателя

Этот метод имеет два обязательных параметра: teacher_name – ФИО преподавателя и week_offset – смещение в неделях, например 0 – текущая неделя, 1 – следующая и т. д. Метод может вернуть один из трех типов ответа, а так как обмен сообщениями осуществляется посредством HTTP-запросов типа GET, можно использовать HTTP-коды для определения типа ответа:

1) 200. Преподаватель был найден, в ответе содержится его расписание на заданную неделю.

2) 404. Преподаватель не был найден, если в базе были найдены преподаватели с похожим ФИО, то возвращает их, иначе – значение false.

3) 503. В данный момент происходит обновление расписания, необходимо сделать повторный запрос позже.

В случае успешного выполнения запроса ответ приходит в виде JSON-объекта, содержащего расписание на все дни недели, начиная с понедельника. Каждый из дней может содержать либо массив занятий, либо false в случае, если в этот день нет занятий. Структура объекта представлена на рис. 7.

```
1  teacher_lesson:
2    type: object
3      properties:
4        lesson_num:
5          type: integer
6        subject_name:
7          type: string
8        group:
9          type: string
10       auditory:
11         type: string
12       time:
13         $ref: '#/components/schemas/lesson_time'
```

Рис. 7. Структура объекта занятия для преподавателя

Модель базы данных, позволяющая реализовать все основные функциональные требования, для описываемой системы представлена на рис. 8.

Organizations содержит список пользователей системы (учебных заведений). Каждое учебное заведение может иметь несколько ботов. Bots хранит информацию о созданных пользователями чат-ботах, в том числе содержит настройки каждого из чат-ботов. Из этой сущности берутся параметры для конфигурации экземпляра чат-бота. Subscribers хранит данные о пользователях чат-бота с привязкой к платформе и

учебному заведению, включая тип пользователя (студент/-преподаватель), группу/имя, подгруппу. Эти данные затем можно использовать при отправке пользователю его расписания.

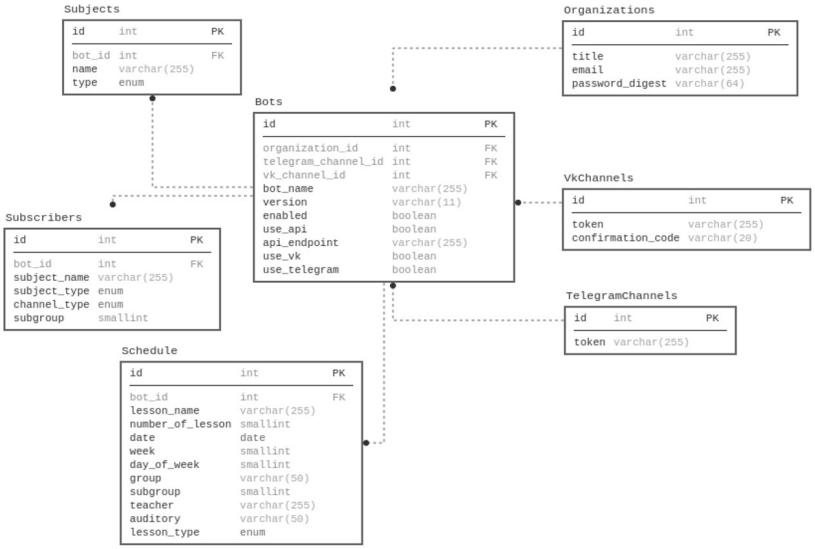

Рис. 8. Модель базы данных

Дальнейший процесс проектирования и разработки диалогов между ботом и пользователем строился с учетом принципов, приведенных в статьях [5; 6]. В качестве демонстрации того, как выглядят диалоги, на рис. 9 приведены скриншоты.

Заключение

В учебном заведении одним из важнейших процессов является составление и доставка участникам образовательного процесса расписания. В рамках данной работы описаны анализ и проектирование информационной системы, позволяющий любому учебному заведению создать собственного чат-бота для доставки расписания на различных платформах, основные взгляды на программную архитектуру системы в целом, модель реляционной базы данных и некоторые аспекты

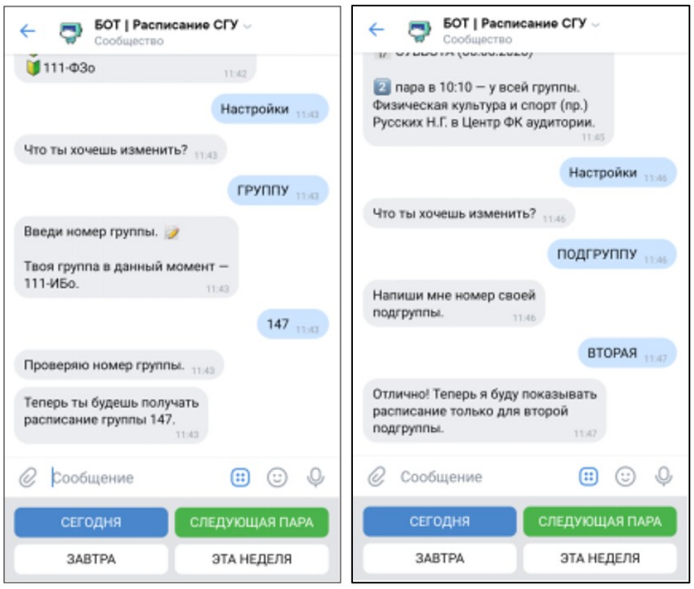

Рис. 9. Настройка академической группы и подгруппы

реализации межсистемного взаимодействия. Подобные технологии могут быть эффективными как самостоятельно, так и при интеграции в другие информационные системы учебного заведения, например, описанные в работах [7; 8].

Стоит отметить, что эксплуатация чат-бота показала его актуальность и в целом востребованность студентами подобного рода сервисов. В период испытаний (с конца 2019 по середину 2020 года) количество запросов колебалось приблизительно от 180 до 600 в период активной учебы и резко упало до нескольких десятков в мае, что связано с переходом к дистанционному обучению и завершением учебного семестра при примерно 1 тысяче зарегистрированных пользователей.

На сегодняшний день в группе бота «ВКонтакте» зарегистрировано более 1,7 тысяч пользователей.

Список литературы

1. Рынок чат-ботов в цифрах и фактах. Инфографика // *Журнал ПЛАС* [Электронный ресурс] URL: https://plusworld.ru/daily/tehnologii/403076-2/ (дата обращения: 26.05.2021).

2. Что такое чат-бот? Oracle Россия и СНГ [Электронный ресурс]. URL: https://www.oracle.com/ru/chatbots/what-is-a-chatbot/ (дата обращения: 26.05.2021).

3. **Maniou T.A., Veglis A.** Employing a Chatbot for News Dissemination during Crisis: Design, Implementation and Evaluation // *Future Internet.* 2020. 12, No. 7. 109 p. DOI: https://doi.org/10.3390/fi12070109.

4. **Carisi M., Albarelli A., Luccio F.L.** Design and implementation of an airport chatbot // *Proceedings of the 5th EAI International Conference on Smart Objects and Technologies for Social Good (GoodTechs '19). Association for Computing Machinery*, New York. Pp. 49–54. DOI: https://doi.org/10.1145/3342428.3342664.

5. **Zarouali B., Evert Van den Broeck, Walrave M., Poels K.** Predicting Consumer Responses to a Chatbot on Facebook // *Cyberpsychology, Behavior, and Social Networking*, 2018. Vol. 21. No. 8. Pp. 491–497. DOI: http://doi.org/10.1089/cyber.2017.0518.

6. **Aarthi Ganitha N., Vaishnavee V., Oviya K., Jayaseelan J. Salem.** Implementation of Chatbot in Trading Application Using SQL and Python // *Bioscience Biotechnology Research Communications,* 2020. Vol. 13. No. 2. Pp. 111–115.

7. **Tsai M-H., Chan H-Y., Liu L-Y.** Conversation-Based School Building Inspection Support System // *Applied Sciences,* 2020. Vol. 10. No. 11. 3739. DOI: https://doi.org/10.3390/app10113739.

8. **Ho C. Chun, Lee H. L., Lo W. K., Lui K. F. A.** Developing a Chatbot for College Student Programme Advisement // International Symposium on Educational Technology (ISET). 2018. Pp. 52-56. DOI: https://doi.org/10.1109/ISET.2018.00021.

9. **Lee L-K., Fung Y-C., Pun Y-W., Wong K-K., Yu M. T-Y., Wu N-I.** Using a Multiplatform Chatbot as an Online Tutor in a University Course // *International Symposium on Educational Technology (ISET).* 2020. Pp. 53–56. DOI: https://doi.org/10.1109/ISET49818.2020.00021.

10. **Wang J., Hwang G-H., Chang C-Y.** Directions of the 100 most cited chatbot-related human behavior research: A review of academic publications // *Computers and Education: Artificial Intelligence.* 2021, 2, 100023.

11. **Гольчевский Ю. В., Виноградов И. М.** Опыт разработки интернет-сервиса расписания учебных занятий // *Информатизация образования и науки.* 2016. № 1. С. 16-25.

12. **Красильников Р. Б., Гольчевский Ю. В.** Непериодический подход к организации и представлению электронного расписания // Двадцать шестая годичная сессия Ученого совета СГУ им. Питирима Сорокина (Февральские чтения) : сборник материалов [Электронный ресурс]: Сыктывкар: Изд-во СГУ им. Питирима Сорокина, 2019. С. 47–76. 1 опт. компакт-диск (CD-ROM).

13. **Skjuve M., Folstad A., Fostervold K. I., Brandtzaeg P. B.** My Chatbot Companion - a Study of Human-Chatbot Relationships // *International Journal of Human-Computer Studies*. 2021. Vol. 149. May, 102601.

Summary

Golchevskiy Yu. V., Nepein A. V. Design and development of a chatbot for presenting a schedule in a social network

The paper presents a study of the problem of delivering schedules to the educational process participants based on the design and development of a chatbot for a social network. The business process of schedule creating was modeled, the platforms for implementing dialog interfaces (chatbots) were analyzed, the software architecture and database were designed and developed, as well as some aspects of implementing the software interaction interface.

Keywords: chatbot, schedule, educational institution, software architecture, social network.

References

1. The chatbot market in numbers and facts. Infographics. *Zhurnal PLAS* [Journal PLAS]. Available at: https://plusworld.ru/daily/tehnologii/403076-2/ (access date: 26.05.2021).

2. *Chto takoe chat-bot? Oracle Rossiia i SNG* [What is a chatbot? Oracle Russia and CIS]. Available at: https://www.oracle.com/ru/chatbots/what-is-a-chatbot/ (access date: 26.05.2021).

3. **Maniou T.A., Veglis A.** Employing a Chatbot for News Dissemination during Crisis: Design, Implementation and

Evaluation. *Future Internet*, 2020, 12, No. 7, 109 p. DOI: https://doi.org/10.3390/fi12070109.

4. **Carisi M., Albarelli A., Luccio F. L.** Design and implementation of an airport chatbot. *Proceedings of the 5th EAI International Conference on Smart Objects and Technologies for Social Good (GoodTechs '19). Association for Computing Machinery*, New York, Pp. 49–54. DOI: https://doi.org/10.1145/3342428.3342664.

5. **Zarouali B., Evert Van den Broeck, Walrave M., Poels K.** Predicting Consumer Responses to a Chatbot on Facebook. *Cyberpsychology, Behavior, and Social Networking,* 2018, Vol. 21, No. 8, Pp. 491–497. DOI: http://doi.org/10.1089/cyber.2017.0518.

6. **Aarthi Ganitha N., Vaishnavee V., Oviya K., Jayaseelan J. Salem.** Implementation of Chatbot in Trading Application Using SQL and Python. *Bioscience Biotechnology Research Communications*, 2020, Vol. 13, No. 2, Pp. 111–115.

7. **Tsai M-H., Chan H-Y., Liu L-Y.** Conversation-Based School Building Inspection Support System. *Applied Sciences*, 2020, Vol. 10, No. 11. 3739. DOI: https://doi.org/10.3390/app10113739.

8. **Ho C. Chun, Lee H. L., Lo W. K., Lui K. F. A.** Developing a Chatbot for College Student Programme Advisement. *International Symposium on Educational Technology (ISET)*, 2018, Pp. 52–56. DOI: https://doi.org/10.1109/ISET.2018.00021.

9. **Lee L-K., Fung Y-C., Pun Y-W., Wong K-K., Yu M. T-Y., Wu N-I.** Using a Multiplatform

Chatbot as an Online Tutor in a University Course. *International Symposium on Educational Technology (ISET)*, 2020, Pp. 53–56. DOI: https://doi.org/10.1109/ISET49818.2020.00021.

10. **Wang J., Hwang G-H., Chang C-Y.** Directions of the 100 most cited chatbot related human behavior research: A review of academic publications. *Computers and Education: Artificial Intelligence*, 2021, 2, 100023.

11. **Golchevskiy Yu. V., Vinogradov I. M.** Experience in developing an online class schedule service. *Informatizatsiia obrazovaniia i nauki* [Informatization of education and science]. 2016. No. 1, Pp. 16–25.

12. **Krasilnikov R. B., Golchevskiy Yu. V.** Non-periodic approach to organizing and presenting electronic timetable. *Dvadtsat shestaia godichnaia sessiia Uchenogo soveta SGU im. Pitirima Sorokina* (Fevralskie chteniia) [Twenty-sixth Annual Session of the Academic Council of SyktSU (February Readings)]: sbornik materialov [collection of materials]: tekstovoe nauchnoe elektronnoe izdanie na kompakt-diske. Syktyvkar: Izd-vo SGU im. Pitirima Sorokina, 2019. Pp. 471–476.

13. **Skjuve M., Folstad A., Fostervold K. I., Brandtzaeg P. B.** My Chatbot Companion – a Study of Human-Chatbot Relationships. *International Journal of Human-Computer Studies*, 2021, Vol. 149, May, 102601.

Для цитирования: Гольчевский Ю. В., Непеин А. В. Проектирование и разработка чат-бота для представления расписания в социальной сети // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика.* 2021. Вып. 3 (40). С. 41–61. DOI: 10.34130/1992-2752_2021_3_41

For citation: Golchevskiy Yu. V., Nepein A. V. Design and development of a chatbot for presenting a schedule in a social network. *Bulletin of Syktyvkar University, Series 1: Mathematics. Mechanics. Informatics*, 2021, No. 3 (40), pp. 41–61. DOI: 10.34130/1992-2752_2021_3_41

СГУ им. Питирима Сорокина *Поступила 11.08.2021*