

DOI: 10.12731/2227-930X-2022-12-3-125-137

УДК 004.93

СИСТЕМА ВЕДЕНИЯ ОНЛАЙН СТАТИСТИКИ УЧАСТНИКОВ В МЕССЕНДЖЕРЕ DISCORD ПОСРЕДСТВОМ PYTHON И POSTGRESQL

*Р.Р. Крапивин, Л.М. Ахметов,
М.Р. Хамидуллин, Г.А. Гареева*

Актуальность рассматриваемой темы обусловлена тем, что в одном из сообществ Discord канала была необходимость по сбору активности участников канала за определенное время и продолжительность.

Для решения этой задачи было принято решение разработать базу данных и специального бота, для взаимодействия с этой базой данных и API Discord для выполнения сбора информации.

Цель – разработка системы для анализа, сбора и вывода информации по активности участников, разработка базы данных для хранения всей информации.

Метод или методология проведения работы: в статье рассмотрен проект по сбору, хранению и выводу информации об активности участников.

Результаты: разработан Discord бот на Python, который ведет статистику по активности участников канала, разработана база данных на PostgreSQL, в которой хранятся данные по всем участникам и их активности.

Область применения результатов: функционал бота можно использовать для менеджмента участников канала, выделения самых активных и их поощрения.

Ключевые слова: Discord; Python; PostgreSQL; статистика; API; база данных

ONLINE STATISTICS SYSTEM FOR PARTICIPANTS IN THE DISCORD MESSENGER USING PYTHON AND POSTGRESQL

*R.R. Krapivin, L.M. Akhmetov,
M.R. Khamidullin, G.A. Gareeva*

The relevance of the topic under consideration is since in one of the Discord channel communities there was a need to collect the activity of channel members for a certain time and duration.

To solve this problem, it was decided to develop a database and a special bot to interact with this database and the Discord API to collect information.

The goal: *development of a system for analyzing, collecting, and displaying information on the activity of participants, developing a database to store all information.*

The method or methodology of the work: *the article considers a project for collecting, storing, and displaying information about the activity of participants.*

Results: *a Python discord bot was developed that keeps statistics on the activity of channel members, a PostgreSQL database was developed that stores data on all participants and their activity.*

Scope of the results: *the functionality of the bot can be used to manage channel members, highlight the most active and allocate incentives for greater activity*

Keywords: *Discord; Python; PostgreSQL; Statistics; API; Database*

Введение

В данной статье рассматривается разработка бота для мессенджера Discord, ведущего статистику активности участников и разработка базы данных на PostgreSQL.

Данную разработку целесообразно для менеджмента участников канала для выделения самых активных участников игровых

событий в определенное время для поощрения большей активности на канале.

В течение дня это событие возникает в 7, 16, 21 и 23 часа и продолжается ровно час. В эти часы бот просматривает кто находится в голосовых каналах и запоминает это. После чего бот выводит отчет и графики о прошедшем событии, а именно какое количество минут и в какое время участники присутствовали.

Материалы и методы

Для правильной работы бота необходимо:

- Наличие библиотек;
- Наличие базы данных;
- Присутствие бота на нужном Discord канале сообщества;
- Наличие хостинга, где будет расположен бот и база данных.

При разработке использовались следующие библиотеки Python, представленные на рисунке 1.

```
from asyncio.tasks import sleep
import requests
import discord
import matplotlib.pyplot as plt
from matplotlib import rcParams
from discord.ext import commands, tasks
import datetime
from matplotlib import font_manager as fm, rcParams
import psycpg2
import asyncio
import random
import json
```

Рис. 1. Библиотеки Python

В начале каждого события бот просматривает определенные голосовые каналы Discord сервера на наличие участников [4, 8]. Когда бот замечает присутствие, запоминается уникальный

идентификатор и присваивается минута активности, к примеру участник появился в голосовом канале в 16:14, эти данные запоминаются и записываются в базу данных при каждой проверке наличия участников. Чтобы избежать дубликатов, бот проверяет их наличие в базе данных, и при необходимости обновляет или добавляет информацию [1]. Схема цикла изображена на рисунке 2.

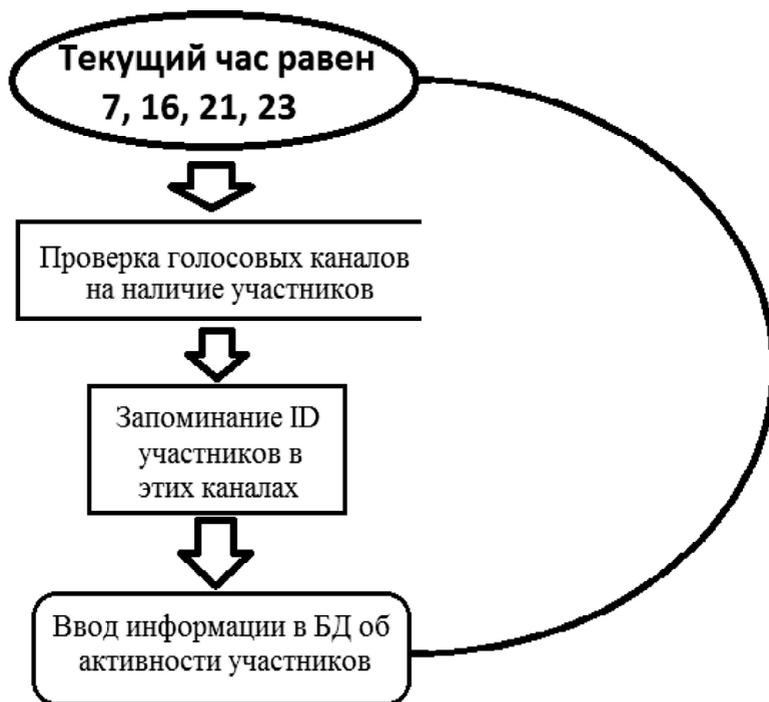


Рис. 2. Схема работы цикла

На рисунке 3 представлен участок кода, который просматривает голосовые каналы и вводит информацию в БД, где цикл повторяется каждые 5 секунд.

После окончания события, бот автоматически формирует график активности по этому событию[2,6]. Для формирования графика используется библиотека `matplotlib`. Для получения данных

по прошедшему событию, используется запрос в БД (рис. 5). По этим данным формируется график активности (рис. 4)

```

for voice_channel in voice_channel_list: # Записывает ID голосовых каналов, в которых будут собираться участники
    if 'БЗС' in str(voice_channel):# При наличии слова "БЗС", этот голосовой канал запоминается для мониторинга
        All_BZS_Channel.append(voice_channel.id)

for Channel_BZS in All_BZS_Channel:# Анализирует каждый голосовой канал
    ID_list = agent.get_channel(Channel_BZS).voice_states.keys()# Получет ID пользователей находящихся в голосовом
    for ID_User in ID_list:
        All_users_id.append(ID_User)# Добавляет Имя участника в общий список All_users
    add_info_in_table(All_users_id, conn, cursor)# Внесение информации об активности участников в БД

```

Рис. 3. Проверка наличия онлайн участников

```

@tasks.loop(seconds = 20)
async def send_itog_everbzs():
    to_time = datetime.datetime.now().strftime("%H:%M")
    if (to_time == '08:01' or to_time == '17:01' or to_time == '22:01' or to_time == '00:01'):
        channel_statistic = agent.get_channel(844548552864563200) # канал, в который будут отправлены графики статистики
        channel_bzs = agent.get_channel(803750654589665300) # второй канал, в который будут отправлены графики статистики
        now = datetime.datetime.now() # Берется текущее время
        to_day = (now - datetime.timedelta(hours=1)).strftime("%Y-%m-%d") # Берется текущая дата, с вычитом часа
        hour = int((now - datetime.timedelta(hours=1)).strftime("%H"))# Берется прошлый час
        title_print = f'БЗС: {to_day} время: {hour} час(ов)' # Формирование заголовка
        emb = discord.Embed( # Формирование embed, это такой тип класса, в котором удобно выводить информацию
            title = title_print,
            colour = discord.Color.blue())
        rows = try_conn(name_table, to_day, hour) # Выполняется запрос в БД, с текущей датой и часом
        if rows != []: # Если в результате запроса нет никаких данных, задача прекращается
            all_img = create_graf_info(rows) # формирование графиков
            await channel_statistic.send(embed = emb) # отправка заголовка
            await channel_statistic.send(files = all_img)# отправка статистики
            all_img = create_graf_info(rows)
            await channel_bzs.send(embed = emb)
            await channel_bzs.send(files = all_img)
        await agent.change_presence(activity=discord.Activity(type=discord.ActivityType.watching, name="АНИМЕ"))
        await asyncio.sleep(61)

```

Рис. 4. Функция по выводу статистики после события

Запрос на получение активности всех участников на указанную дату представлен на рисунке 4.

```

def try_conn(name_table, to_day, hour):
    conn = init_bd()
    cursor = conn.cursor()
    cursor.execute(f"SELECT * FROM {name_table} WHERE DATA_BZS = '{to_day}' and HOUR = '{hour}';")
    rows = cursor.fetchall()
    conn.close()
    return rows

```

Рис. 5. Запрос, который заполняется в Python и отправляется на выполнение в PostgreSQL

Для получения информации по всей активности за последнюю неделю, используется запрос в базу данных показанный на рисунке 6.

```

def get_rows_week():
    conn = init_bd()
    cursor = conn.cursor()
    cursor.execute(f""" SELECT player_id, SUM(ARRAY_LENGTH(score,1)) AS bzs_sum FROM (name_table) WHERE
    data_bzs BETWEEN (NOW() - '7 days'::INTERVAL)::DATE AND NOW()::DATE GROUP BY player_id""")
    rows = cursor.fetchall()
    conn.close()
    return rows

```

Рис. 6. Запрос на получение активности всех участников за неделю

Для хранения данных была выбрана база данных PostgreSQL. PostgreSQL является реляционной базой данных, она подходит для реализации функционала хранения информации по активности. База данных состоит из 1 главной таблицы, показанной на рисунке 7.

id	player_id	score	data_bzs	hour	
[PK] integer	bigint	integer[]	date	integer	
1	2445	369427180834521100	{0,1,2,3,4,5,6,7}	2021-12-04	16
2	2444	552061405910532116	{15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33}	2021-11-30	16
3	2443	201783737631899648	{24,25,26,27,28,29,30,31}	2021-11-28	16
4	2442	518825460986019855	{33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51}	2021-11-28	16
5	2441	369427180834521100	{41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59}	2021-11-28	16
6	2440	246659285885386753	{44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59}	2021-11-27	16
7	2439	518825460986019855	{0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22}	2021-11-27	16
8	2438	518825460986019855	{29}	2021-11-26	23
9	2437	425312253026500628	{32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50}	2021-11-26	16
10	2436	369427180834521100	{32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50}	2021-11-26	16
11	2435	552061405910532116	{0,1,2,3,4,5,6}	2021-11-25	21
12	2434	518064875411341353	{41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59}	2021-11-25	16
13	2433	425312253026500628	{41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59}	2021-11-25	16
14	2432	201783737631899648	{0,1,2,3,4,5,6}	2021-11-25	16
15	2431	518825460986019855	{14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32}	2021-11-25	16

Рис. 7. База данных PostgreSQL

Таблица состоит из следующих столбцов, где:

- player_id – это идентификационный номер пользователя в Discord
- score – массив, в котором хранятся минуты онлайн
- data_bzs – это день когда был пользователь был онлайн
- hour – час это в какой именно час были в онлайн

Результаты

В результате сбора информации во время события, бот формирует график активности всех участников, которые на нем присутствовали, используя запрос в базе данных. Пример графика показан на рисунке 8, где ось X – это минутная полоса, ось Y – это полоса активности, над графиком указано имя участника канала, который присутствовал на событии [7].



Рис. 8. График с данными о событии

Бот формирует еженедельный отчет, который отправляется в понедельник после 00:00 по МСК. В нем отображены имена участников Discord канала, имена аккаунтов и их суммарное присутствие на событиях за всю неделю, так же бот показывает всех участников, которые ни разу не присутствовали во время события. Пример отчета показан на рисунке 9.

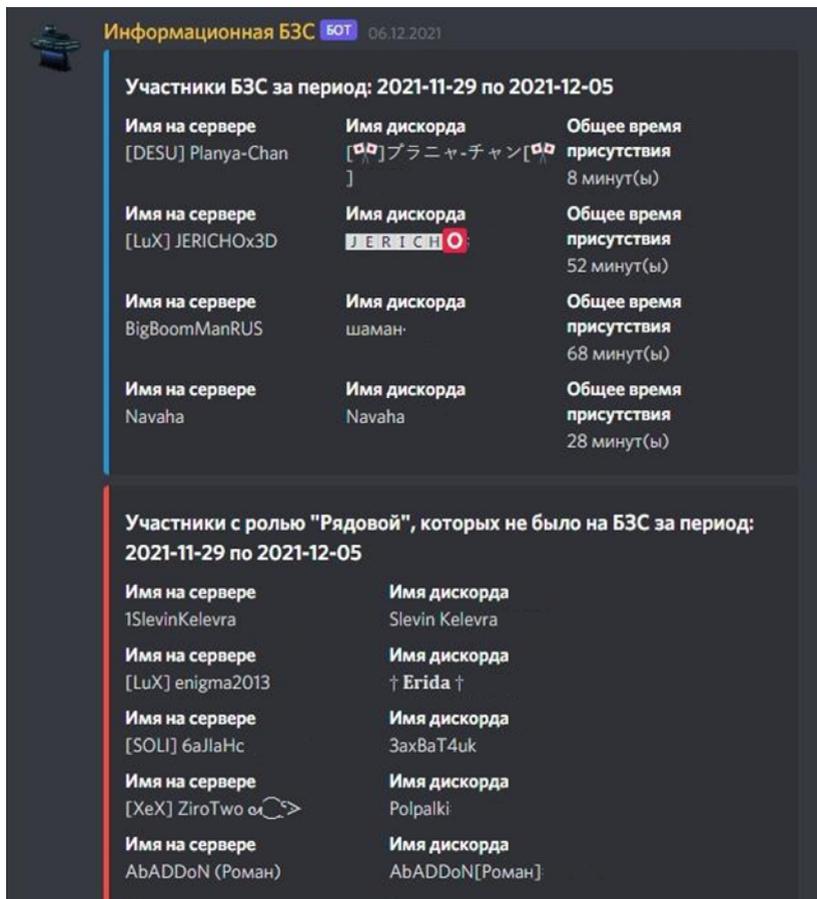


Рис. 9. Отчет по участникам за неделю

Выводы

Был разработан Discord бот на Python, который ведет статистику по активности участников канала, разработана база данных на PostgreSQL, в которой хранятся данные по всем участникам и их активности, для взаимодействия с базой данных, используется библиотека psycorg2, разработан статистический вывод данных в виде отчетов и графиков, использующий

графические инструменты из библиотеки matplotlib. Функционал бота можно использовать для поощрения самых активных участников событий или для отслеживания неактивных, повышая менеджмент активности Discord сообщества. Бот размещен на хостинге Heroku там же и расположена база данных. В будущем, функционал бота можно расширить, добавив возможность вывода активности отдельно взятых участников, указывая интересующую дату или период [10].

Список литературы

1. Khamidullin M.R., Mardanshin R.G., Prozorov A.V., Karimov R.I. The Introduction of QR-Codes in Production Processes // Journal of Environmental Treatment Techniques. 2019, Special Issue on Environment, Management and Economy. P. 1097-1100.
2. Akhmetov L.M., Bikov D.I., Khamidullin M.R., Gareeva G.A., Gabdullina G.K. Development of a system for analyzing and unloading road traffic using artificial intelligence // Journal of Physics: Conference Series, 2021, vol. 2094. <http://dx.doi.org/10.1088/1742-6596/2094/3/032036>
3. Ахметов Л.М. Разработка системы анализа влажности и температуры в помещении на базе микроконтроллера // Инновационные технологии, экономика и менеджмент в промышленности: сборник научных статей IV международной научной конференции. 22-23 апреля 2021 г. Часть 1.: материалы конференции. Волгоград: ООО «Конверт», 2021. 248 с.
4. Ахметов Л.М., Биков Д.И., Хамидуллин М.Р. Разработка системы для анализа и разгрузки дорожного трафика с применением искусственного интеллекта // International Journal of Advanced Studies, 2021, vol. 11(1), pp. 87-98.
5. Биков Д.И., Насибулин Р.О., Гареева Г.А. Потенциал и перспективы использования технологии интернет вещей // Приоритетные направления инновационной деятельности в промышленности: сборник научных статей международной научной конференции. 30-31 января 2021 г. Казань: ООО «Конверт», 2021. С.188-189.

6. Бич, Мартин. Микроконтроллеры семейства XC166. Вводный курс разработчика / Мартин Бич, Дэвид Гринхилл. М.: ДМК Пресс, Додэка XXI, 2016. 200 с.
7. Кечиев Л.Н. IBIS-модели и их применение в задачах ЭМС. М.: Грифон, 2016. 638 с.
8. Крапивин Р.Р., Гареева Г.А. Получение доступа к данным путем авторизации в аккаунт с помощью библиотеки Requests в языке Python // Инновационные технологии, экономика и менеджмент в промышленности: сборник научных статей IV международной научной конференции. 22-23 апреля 2021 г. Часть 1. Волгоград: ООО «Конверт», 2021. С. 206-208.
9. Правоткин И.А. Настройка и запуск программ на Python на удалённом хостинге // Приоритетные направления инновационной деятельности в промышленности сборник научных статей по итогам двенадцатой международной научной конференции. Казань: ООО «Конверт», 2020. С. 78-80.
10. Тугов В.В. Проектирование автоматизированных систем управления : учебное пособие / В.В. Тугов, А.И. Сергеев, Н.С. Шаров. Санкт-Петербург: Лань, 2019. 172 с.

References

1. Khamidullin M.R., Mardanshin R.G., Prozorov A.V., Karimov R.I. The Introduction of QR-Codes in Production Processes. *Journal of Environmental Treatment Techniques*. 2019, *Special Issue on Environment, Management and Economy*, pp. 1097-1100.
2. Akhmetov L.M., Bikov D.I., Khamidullin M.R., Gareeva G.A., Gabdullina G.K. Development of a system for analyzing and unloading road traffic using artificial intelligence. *Journal of Physics: Conference Series*, 2021, vol. 2094. <http://dx.doi.org/10.1088/1742-6596/2094/3/032036>
3. Akhmetov L.M. *Innovatsionnye tekhnologii, ekonomika i menedzhment v promyshlennosti: sbornik nauchnykh statey IV mezhdunarodnoy nauchnoy konferentsii. 22-23 aprelya 2021 g. Chast' 1.: materialy konferentsii* [Innovative technologies, economics and management in

- industry: collection of scientific articles of the IV international scientific conference. April 22-23, 2021 Part 1: conference proceedings]. Volgograd: Envelope LLC, 2021, 248 p.
4. Akhmetov L.M., Bikov D.I., Khamidullin M.R. *International Journal of Advanced Studies*, 2021, vol. 11(1), pp. 87-98.
 5. Bikov D.I., Nasibulin R.O., Gareeva G.A. *Prioritetnye napravleniya innovatsionnoy deyatel'nosti v promyshlennosti: sbornik nauchnykh statey mezhdunarodnoy nauchnoy konferentsii. 30-31 yanvarya 2021 g* [Priority areas of innovative activity in industry: collection of scientific articles of the international scientific conference. January 30-31, 2021]. Kazan: Envelope LLC, 2021, pp.188-189.
 6. Beach, Martin. *Mikrokontrollery semeystva XC166. Vodnyy kurs razrabotchika* [Microcontrollers of the XC166 family. Introductory Developer Course] / Martin Beach, David Greenhill. M.: DMK Press, Dodeka XXI, 2016, 200 p.
 7. Kechiev L.N. *IBIS-modeli i ikh primeneniye v zadachakh EMS* [IBIS-models and their application in EMC problems]. M.: Grifon, 2016, 638 p.
 8. Krapivin R.R., Gareeva G.A. *Innovatsionnye tekhnologii, ekonomika i menedzhment v promyshlennosti: sbornik nauchnykh statey IV mezhdunarodnoy nauchnoy konferentsii. 22-23 aprelya 2021 g. Chast' I* [Innovative technologies, economics and management in industry: collection of scientific articles of the IV international scientific conference. April 22-23, 2021 Part 1]. Volgograd: Envelope LLC, 2021, pp. 206-208.
 9. Pravotkin I.A. *Prioritetnye napravleniya innovatsionnoy deyatel'nosti v promyshlennosti sbornik nauchnykh statey po itogam dvenadtsatoy mezhdunarodnoy nauchnoy konferentsii* [Priority areas of innovative activity in the industry. Collection of scientific articles following the results of the twelfth international scientific conference]. Kazan: Envelope LLC, 2020, pp. 78-80.
 10. Tugov V.V. *Proektirovaniye avtomatizirovannykh sistem upravleniya : uchebnoye posobie* [Design of automated control systems: textbook] / V.V. Tugov, A.I. Sergeev, N.S. Sharov. St. Petersburg: Lan, 2019, 172 p.

ДАННЫЕ ОБ АВТОРАХ**Крапивин Роман Русланович**, студент

*Набережночелнинский филиал Казанского национального
исследовательского технического университета им. А.Н.
Туполева
Академика Королева, 1, г. Набережные Челны, 423814, Рос-
сийская Федерация
Jerichotyrant1@yandex.ru*

Ахметов Линар Марселевич, студент

*Набережночелнинский филиал Казанского национального
исследовательского технического университета им. А.Н.
Туполева
Академика Королева, 1, г. Набережные Челны, 423814, Рос-
сийская Федерация
diobrandmayer@gmail.com*

Хамидуллин Марат Раисович, доцент, кандидат экономических наук

*Набережночелнинский филиал Казанского национального
исследовательского технического университета им. А.Н.
Туполева
Академика Королева, 1, г. Набережные Челны, 423814, Рос-
сийская Федерация
наука_prom@mail.ru*

Гареева Гульнара Альбертовна, доцент, кандидат педагогиче-ских наук

*Набережночелнинский филиал Казанского национального
исследовательского технического университета им. А.Н.
Туполева
Академика Королева, 1, г. Набережные Челны, 423814, Рос-
сийская Федерация
gagareeva1977@mail.ru*

DATA ABOUT THE AUTHORS

Roman R. Krapivin, student

*Kazan National Research Technical University named after A. N. Tupolev – KAI, Branch in Naberezhnye Chelny
1, Akademika Koroleva Str., Naberezhnye Chelny, 423814, Russian Federation
Jerichotyrant1@yandex.ru*

Linar M. Akhmetov, student

*Kazan National Research Technical University named after A. N. Tupolev – KAI, Branch in Naberezhnye Chelny
1, Akademika Koroleva Str., Naberezhnye Chelny, 423814, Russian Federation
diobrandmayer@gmail.com*

Marat R. Khamidullin, PhD

*Kazan National Research Technical University named after A. N. Tupolev – KAI, Branch in Naberezhnye Chelny
1, Akademika Koroleva Str., Naberezhnye Chelny, 423814, Russian Federation
nayka_prom@mail.ru
ORCID: <https://orcid.org/0000-0002-3326-0955>*

Gulnara A. Gareeva, PhD

*Kazan National Research Technical University named after A. N. Tupolev – KAI, Branch in Naberezhnye Chelny
1, Akademika Koroleva Str., Naberezhnye Chelny, 423814, Russian Federation
gagareeva1977@mail.ru*

Поступила 10.05.2022

После рецензирования 25.05.2022

Принята 30.05.2022

Received 10.05.2022

Revised 25.05.2022

Accepted 30.05.2022