

Ерофеев Ермак Владимирович,

студент гр. ИТБ-23-1, Ангарский государственный технический университет,

e-mail: ermak.080@mail.ru

Кулакова Ирина Михайловна

к.т.н., доцент, Ангарский государственный технический университет,

e-mail: iyelkina@mail.ru

ИНСТРУМЕНТ АВТОМАТИЗАЦИИ ОПРОСОВ В КОРПОРАТИВНОЙ ИНФРАСТРУКТУРЕ

Erofeev E.V., Kulakova I.M.

A TOOL FOR AUTOMATING SURVEYS IN THE CORPORATE INFRASTRUCTURE

Аннотация В данной статье представлен проект по разработке бота для создания и управления опросами и голосованиями на платформе Mattermost, реализованный на языке программирования Go. Проект направлен на оптимизацию процессов взаимодействия в организациях. Статья также рассматривает архитектуру хранения данных с использованием Tarantool и PostgreSQL, что гарантирует безопасность и целостность информации. Функциональные возможности бота включают создание опросов, голосование, просмотр результатов и управление опросами, что делает его интуитивно понятным для пользователей. Проект не только улучшает взаимодействие в командах, но и способствует более демократичному и прозрачному процессу принятия решений.

Ключевые слова: Mattermost, опросы, голосование, Go.

Abstract. This article presents a project to develop a bot for creating and managing polls and voting on the Mattermost platform, implemented in the Go programming language. The project is aimed at optimizing the processes of interaction in organizations. The article also examines the architecture of data storage using Tarantool and PostgreSQL, which guarantees the security and integrity of information. The bot's functionality includes creating polls, voting, viewing results, and managing polls, making it intuitive for users. The project not only improves teamwork, but also promotes a more democratic and transparent decision-making process.

Keywords: Mattermost, polls, voting, Go.

Современные организации стремятся к оптимизации процессов взаимодействия и принятия решений. Одним из эффективных инструментов для достижения этой цели являются опросы и голосования, которые позволяют собирать мнения и предпочтения сотрудников. В данной статье представлен проект, разработанный на языке Go [1], который предлагает решение для создания и управления опросами в Mattermost – на платформе для командного общения с открытым исходным кодом [2]. Этот проект не только упрощает процесс создания опросов, но и предоставляет уникальные функции, которые выделяют его среди существующих решений.

Проект вносит значительный вклад в область интерактивных систем опросов и голосований, сочетая современные технологии и подходы к разработке. Использование языка Go обеспечивает высокую производительность и параллелизм, позволяя обрабатывать множество запросов одновременно. В результате

бот становится незаменимым помощником в формировании коллективного мнения, предоставляя пользователям возможность легко создавать опросы, голосовать и просматривать результаты через простые текстовые команды.

Разработка бота на языке Go позволяет достичь высокой производительности и надежности. Бот поддерживает различные команды, что делает его функциональным инструментом для взаимодействия с пользователями. Например, команды @create, @vote, @results, @finished и @delete позволяют пользователям легко управлять опросами. Это обеспечивает интуитивно понятный интерфейс, который не требует от пользователей специальных технических знаний.

Проект интегрирован с Mattermost (рисунок 1), который обеспечивает безопасный и гибкий обмен сообщениями, интеграцию с различными инструментами и возможность кастомизации под нужды бизнеса. Mattermost предоставляет возможность создания каналов, обмена файлами и интеграции ботов. Это позволяет командам организовывать опросы и голосования прямо в контексте их работы, что значительно упрощает процесс взаимодействия.

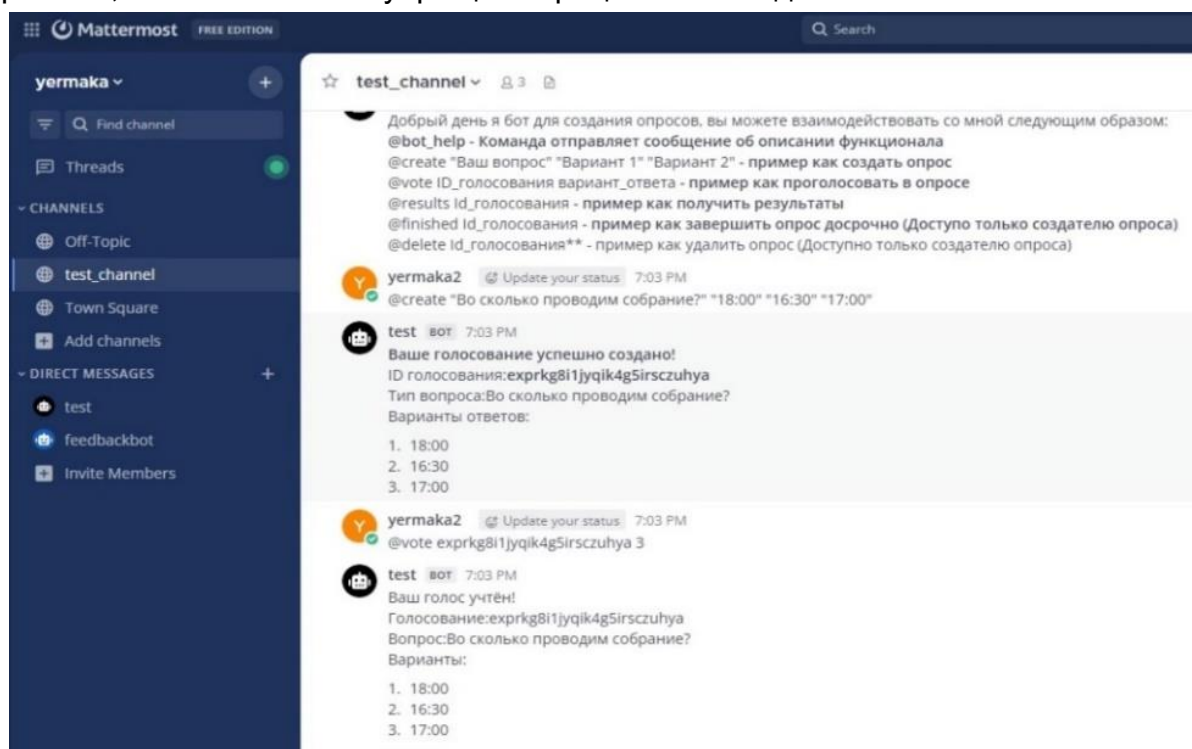
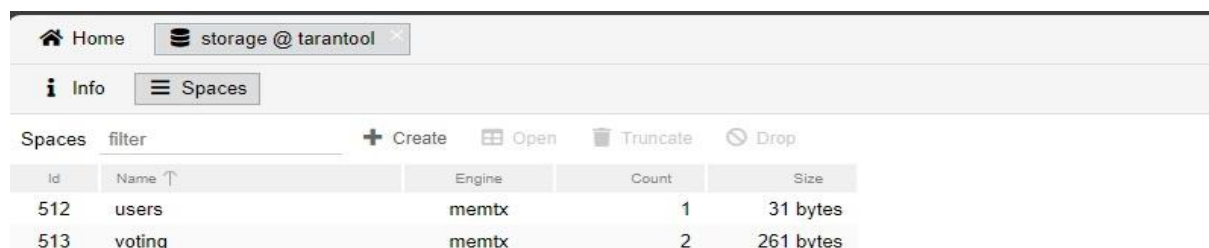


Рисунок 1 – Работа бота в системе Mattermost

Для хранения данных о голосованиях и опросах используется Tarantool – это высокопроизводительная NoSQL база данных и сервер приложений, разработанная для работы с данными в реальном времени [3]. Она сочетает в себе функциональность базы данных с возможностями выполнения бизнес-логики, что позволяет создавать эффективные и масштабируемые приложения,

обеспечивающие высокую производительность и надежность. PostgreSQL поддерживает Mattermost, гарантируя надежное хранение всех данных платформы [4]. Такая архитектура позволяет избежать потерь данных и обеспечивает быстрый доступ к информации.

Tarantool Admin предоставляет удобный веб-интерфейс для просмотра и управления данными, что упрощает администрирование и мониторинг состояния базы данных. Это позволяет администраторам эффективно управлять данными и обеспечивать их целостность (Рисунок 2).



The screenshot shows the Tarantool Admin web interface. At the top, there is a navigation bar with 'Home' and 'storage @ tarantool'. Below it, there are tabs for 'Info' and 'Spaces'. The 'Spaces' tab is active, showing a table with columns: Id, Name, Engine, Count, and Size. The table contains two rows: one for 'users' (Id 512, Count 1, Size 31 bytes) and one for 'voting' (Id 513, Count 2, Size 261 bytes). Above the table, there are buttons for '+ Create', 'Open', 'Truncate', and 'Drop'.

Id	Name	Engine	Count	Size
512	users	memtx	1	31 bytes
513	voting	memtx	2	261 bytes

Рисунок 2 – Веб-интерфейс для просмотра и управления данными Tarantool

Проект можно быстро развернуть в любом окружении с помощью Docker и Docker Compose. Для этого необходимо настроить файл .env с необходимыми параметрами, такими как токены доступа и пароли. Установка Docker и Docker Compose является первым шагом, который необходимо выполнить перед запуском проекта. Запуск проекта осуществляется с помощью команды docker-compose up --build, которая создает и запускает все необходимые компоненты в виде контейнеров. Это позволяет значительно упростить процесс развертывания и минимизировать время на настройку.

Взаимодействие с ботом происходит по протоколу WebSocket, что обеспечивает быстрый обмен сообщениями между пользователями и сервером. Бот поддерживает несколько ключевых операций (Рисунок 3):

- Создание опросов: Команда @create "вопрос" "вариант 1" "вариант 2" позволяет пользователям создавать новые опросы.
- Голосование: Команда @vote ID_голосования вариант_ответа позволяет пользователям голосовать в созданных опросах.
- Просмотр результатов: Команда @results Id_голосования выводит результаты текущего голосования.
- Завершение опроса: Команда @finished Id_голосования завершает опрос досрочно и доступна только создателю.
- Удаление опроса: Команда @delete Id_голосования позволяет удалить опрос и доступна только создателю.

- Справка о боте: Команда @bot_help предоставляет информацию о функционале бота.

При первом открытии личных сообщений с ботом он отправляет вступительное сообщение с описанием своего функционала каждому новому пользователю, кроме создателя бота. Это позволяет быстро ознакомиться с возможностями бота и начать взаимодействие.



Рисунок 3 – Ключевые операции бота

Файл .env содержит конфиденциальные данные, такие как пароли и токены, используемые в проекте. Настройка этого файла является важным шагом, который позволяет обеспечить безопасность и корректную работу бота. На рисунке 4 представлен пример содержимого файла .env.

```
PASS=12345 # пример пароля для postgres
BOT_TOKEN=z8yxbmc5tbbbt9k4j4j3jghxw # пример токена доступа для бота в mattermost
MATTERMOST_URL=http://app:8065
MATTERMOST_WS=ws://app:8065
TARANTOOL_ADDR=tarantool:3301
TARANTOOL_PASS=password
TARANTOOL_USER=storage
```

Рисунок 4 – Наполнение env файла

В качестве основных преимуществ использования данного решения отметим следующие:

- Высокая производительность: язык Go обеспечивает быструю обработку запросов и минимальное время отклика, что особенно важно для команд с большим количеством пользователей.
- Безопасность данных: интеграция с Mattermost и использование надежных баз данных, таких как PostgreSQL и Tarantool, гарантируют защиту данных и их целостность.
- Гибкость и настраиваемость: открытая архитектура Mattermost позволяет легко интегрировать дополнительные функции и адаптировать бот под специфические нужды команды.

- Удобство использования: интуитивно понятный интерфейс и команды делают процесс создания и участия в опросах простым и доступным для всех пользователей, независимо от их технического уровня.

Проект по созданию опросов и голосований в Mattermost, реализованный на языке Go, представляет собой мощный инструмент для улучшения взаимодействия в командах и организации процесса принятия решений. Его высокая производительность, безопасность и интуитивно понятный интерфейс делают его отличным выбором для организаций, стремящихся оптимизировать свои внутренние процессы. С помощью этого решения команды могут быстро и эффективно собирать мнения, что способствует более демократичному и прозрачному принятию решений.

ЛИТЕРАТУРА

1. Go documentation - [Электронный ресурс]. - URL: <https://go.dev/doc/> (дата обращения: 29.04.2025).
2. Mattermost Documentation. - [Электронный ресурс]. - URL: <https://github.com/mattermost/docs> (дата обращения: 30.04.2025)
3. Tarantool – documentation - [Электронный ресурс]. - URL: <https://www.tarantool.io/en/doc/latest/> (дата обращения 30.04.2025)
4. PostgreSQL documentation – [Электронный ресурс] – URL: <https://www.postgresql.org/docs/> (дата обращения 30.04.2025)