

## Разработка и развертывание Telegram бота для оповещения о новостях ДонНТУ

В.Ю. Олейник<sup>\*1</sup>, А.В. Чернышова<sup>\*2</sup>

<sup>\*1</sup> студент, Донецкий национальный технический университет,  
ultimathul3@ya.ru

<sup>\*2</sup> старший преподаватель, Донецкий национальный технический университет,  
chernyshova.alla@rambler.ru, OrcID: 0000-0003-2546-2167, SPIN-код: 3318-2066

*Олейник В.Ю., Чернышова А.В. Разработка и развертывание Telegram бота для оповещения о новостях ДонНТУ. В статье представлен краткий обзор мессенджера Telegram и его возможностей для разработчиков. Спроектирован, разработан и размещен на сервер Telegram бот для оповещения о новостях ДонНТУ.*

**Ключевые слова:** Telegram, Bot API, язык программирования Go, автоматизированный сбор информации, развертывание.

### Введение

Telegram – это облачное приложение для обмена сообщениями (мессенджер), которое делает акцент на скорость и безопасность. В марте 2022 года Telegram стал самым популярным мессенджером в России. За первые две недели марта его доля в общем объеме трафика в мессенджерах увеличилась с 48% до 63% по сравнению с аналогичным периодом февраля месяца.

Telegram предоставляет два вида API (программных интерфейсов сервиса) для разработчиков:

1. Telegram API – позволяет создавать собственные клиенты Telegram.
2. Bot API – позволяет создавать программы (боты), которые используют сообщения Telegram в качестве интерфейса.

Название “бот” произошло от сокращения слова “робот”. Telegram бот представляет собой специальный аккаунт, который получает команды от пользователя, обрабатывает их и генерирует ответ. Также бот может самостоятельно отправлять сообщения пользователю для каких-либо оповещений. Физически команды обрабатываются на серверах разработчиков ботов обычными программами.

Платформа Telegram содержит более десяти миллионов ботов, которые используются для самых различных целей: аналитика данных, модерация групп и каналов, создание опросов, интеграция с внешними сервисами, размещение HTML5 игр и многое другое.

Боты Telegram используются многими пользователями мессенджера и являются важным его элементом.

### Постановка задачи

Факультеты ДонНТУ имеют собственные каналы для оперативной публикации информации. Целью данной статьи является создание Telegram бота для оповещения о новостях ДонНТУ, которого можно встроить в Telegram каналы факультетов. На момент написания статьи подобных ботов не существует, следовательно, данная задача является актуальной и имеет практическую ценность.

### Анализ требований

Исходя из постановки задачи, были определены следующие требования для Telegram бота:

- Telegram бот должен работать с пользователями, группами и каналами;
- Telegram бот должен иметь команду для подписки на рассылку новостей;
- Telegram бот должен иметь команду для отмены подписки на рассылку новостей;
- Telegram бот должен иметь команду для ручной отправки необходимой новости ДонНТУ по ссылке;
- Telegram бот должен с заданной периодичностью проверять новости с сайта ДонНТУ;
- Telegram бот должен отправлять новости всем подписавшимся на рассылку пользователям, группам и каналам.

## Регистрация Telegram бота

Перед началом разработки бота его нужно зарегистрировать в самом мессенджере Telegram. Для этого необходимо обратиться к боту BotFather и указать название и логин создаваемого бота (см. рис. 1). Там же есть возможность задать описание боту, поменять фото его профиля, создать список доступных команд.

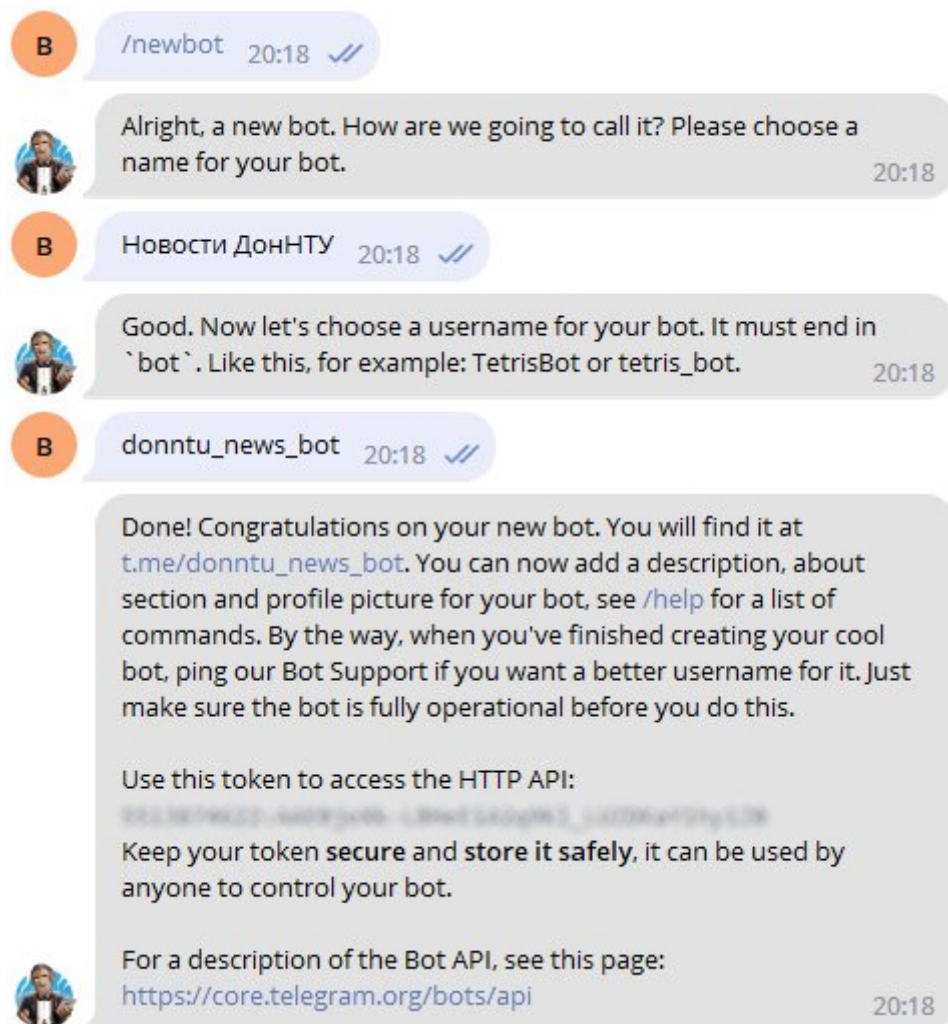


Рисунок 1 – Взаимодействие с ботом BotFather

После регистрации BotFather отправит токен доступа регистрируемого бота.

Токен доступа – это секретный ключ, состоящий из набора цифр, букв латинского алфавита и символов “:”, “-”, “\_”. Токен доступа используется для взаимодействия с Bot API Telegram.

## Взаимодействие с Bot API

Взаимодействие с Bot API происходит с помощью GET или POST HTTP запросов на сервер Telegram. В URL запросов необходимо указать токен доступа бота, имя метода и параметры метода при необходимости. Шаблон и примеры запросов изображены на рисунке 2.

```
https://api.telegram.org/bot<ТОКЕН_ДОСТУПА>/<ИМЯ_МЕТОДА>
```

```
https://api.telegram.org/bot123456:ABC-DEF1234ghIk1-zyx57W2v1u123ew11/getMe
```

```
https://api.telegram.org/bot123456:ABC-DEF1234ghIk1-zyx57W2v1u123ew11/sendMessage?chat_id=12345678&text=Привет
```

Рисунок 2 – Шаблон и примеры запросов

В Bot API имеются методы для отправки сообщений (фото, видео, аудио, документы, геолокация), редактирования информации чатов (редактирование фото, названия, описания), взаимодействия с пользователями чатов (блокировка, разблокировка пользователей, назначение привилегий) и многие другие [1].

Ответ на запрос приходит в формате JSON, который имеет обязательное логическое поле “ok” и необязательное поле “description” с описанием результата. Также в случае неудачного запроса возвращается поле “error\_code” с кодом ошибки.

### **Получение обновлений для Telegram бота**

Для того чтобы бот знал о произошедших событиях (пользователь отправил сообщение, пригласил в чат и др.), он должен получить обновления от сервера Telegram. Существует 2 способа получить обновления для бота:

- метод Bot API “getUpdates”;
- вебхуки (webhooks).

Схематично различия между этими двумя способами изображены на рисунке 3.

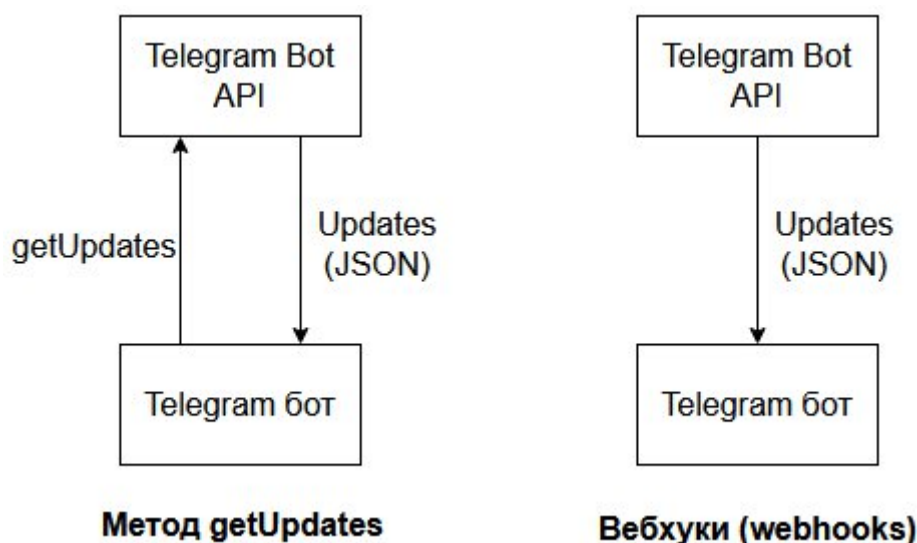


Рисунок 3 – Различия между способами получения обновлений

Способ с методом “getUpdates” предполагает, что бот с некоторой периодичностью обращается к серверу Telegram для получения обновлений. Данный способ имеет несколько недостатков:

- пользователь может ощущать значительную задержку получения ответа от бота (в зависимости от периода обращения бота к серверу);
- процессорное время сервера бота тратится без пользы в случае длительного отсутствия обновлений.

Способ с вебхуками лишен недостатков способа с методом “getUpdates”. В этом случае сервер Telegram самостоятельно посылает обновления на сервер бота. Однако для этого сервер бота должен соответствовать следующим требованиям:

- белый IP-адрес сервера (все устройства в сети Интернет могут напрямую обращаться к устройству с белым IP-адресом);
- сервер должен работать по протоколу HTTPS.

Для разрабатываемого бота был выбран способ получения обновлений с помощью вебхуков. Активация данного способа была произведена с помощью метода setWebhook со следующими параметрами:

- secret\_token (секретный токен);
- url (унифицированный адрес ресурса).

В параметре “secret\_token” был указан произвольный ключ, который Telegram будет отправлять боту в каждом запросе с обновлениями в заголовке “X-Telegram-Bot-API-Secret-Token”. Это необходимо для того, чтобы убедиться, что входящий запрос был отправлен именно сервером Telegram.

В параметре “url” был указан домен donntu-news.ga, полученный через сервис поставщика бесплатных доменов Freenom. Настройки домена изображены на рисунке 4.

## Modify Records

Name	Type	TTL	Target	
	A	3600	194.87.69.188	Delete
WWW	A	3600	194.87.69.188	Delete

Рисунок 4 – Настройки домена donntu-news.ga

В настройках должны быть как минимум две доменные записи, содержащие IP-адреса серверов, на которые будет ссылаться домен. Второй IP-адрес может быть отличен от первого, тогда, если сервер на первом IP-адресе недоступен, запрос будет отправлен на сервер второго IP-адреса. Колонка TTL (time to live, время жизни) указывает на максимальное количество маршрутизаторов, через которые может пройти IP-пакет до своей цели, перед тем как будет уничтожен. Благодаря этому IP-пакет не будет бесконечно блуждать по сети Интернет.

Физически бот будет расположен на виртуальном сервере (VPS) российского хостинга JustHost. IP-адрес этого сервера и был указан в настройках домена. Характеристики выбранного VPS следующие:

- ОС Ubuntu Server 20.04 64bit;
- ЦП Intel Xeon E5-2697 (доступно 1 ядро);
- 512 Мб ОЗУ;
- 10 Гб ПЗУ.

Данных характеристик достаточно для работы разрабатываемого бота. Подключение к VPS и работа с ним происходит по протоколу SSH.

### Разработка Telegram бота

Для разработки Telegram бота выбран язык программирования Go. Язык был официально представлен компанией Google в 2009 году. Он имеет лаконичный синтаксис, является компилируемым и в своём составе содержит множество библиотек для разработки серверных приложений, следовательно, хорошо подходит для решения поставленной задачи.

Проект приложения бота состоит из следующих компонентов:

- пакет “api”;
- пакет “db”;
- пакет “handler”;
- пакет “logger”;
- пакет “observer”;
- пакет “parser”;
- пакет “types”;
- файл “.env”;
- файл “main.go”.

В файле “main.go” содержится точка входа в программу, где запускается сервер. Перед этим инициализируются модуль логирования и база данных, считывается “.env” файл, в котором находятся настройки бота: имя домена, токен доступа, секретный токен и период проверки новостей сайта ДонНТУ в минутах. Для работы сервера по протоколу HTTPS необходим SSL сертификат, который позволяет использовать безопасное зашифрованное соединение. Стандартный пакет “autocert” языка Go позволяет получить SSL сертификат от центра сертификации Let’s Encrypt (см. рис. 5).

```
// autocert provides automatic access to certificates from Let's Encrypt
certManager := autocert.Manager{
    Prompt:      autocert.AcceptTOS,
    HostPolicy:  autocert.HostWhitelist(os.Getenv("DOMAIN")),
    Cache:       autocert.DirCache("certs"),
}
```

Рисунок 5 – Менеджер пакета “autocert” для получения SSL сертификата от Let’s Encrypt

В менеджере пакета “autocert” передается имя домена (считанное из “.env” файла) и директория, куда будет сохранен SSL сертификат.

Получение сертификата происходит по протоколу ACME (Automated Certificate Management Environment, протокол автоматического управления сертификатами), разработанный Исследовательской группой по интернет-безопасности для их сервиса Let’s Encrypt [2].

Telegram позволяет использовать и самоподписанные SSL сертификаты, т.е. сертификаты, подписанные самим сервером. В этом случае в методе “setWebhook” необходимо передать параметр “certificate” для загрузки публичного ключа сертификата. Т.к. для самоподписанного сертификата не обязательно иметь домен, вместо параметра “url” можно передать параметр “ip\_address” (IP-адрес сервера).

Пакет “types” содержит необходимые типы данных (структуры) для работы бота:

- Update (обновление от Telegram);
- Response (ответ от Telegram);
- InputMedia (для отправки медиа файлов).

Пакет “api” содержит необходимые методы для взаимодействия с Bot API:

- отправка сообщения;
- отправка группы фотографий.

Пакет “db” содержит необходимые методы для взаимодействия с базой данных. В качестве системы управления базами данных была выбрана SQLite3, её особенностью является то, что вся база данных состоит всего из одного файла. Это позволяет с легкостью переносить базу данных с одного сервера на другой. Для выполняемой задачи не требуется высокая производительность от СУБД, следовательно, выбор SQLite3 является оптимальным. База данных состоит из двух таблиц:

- subscribers (пользователи, каналы или группы, подписанные на рассылку новостей);
- last\_news (таблица, которая хранит дату последней новости на сайте ДонНТУ).

Пакет “logger” содержит модуль логирования, с помощью которого во время работы бота в файл записываются следующие события с временной меткой: запуск бота, получение обновлений от Telegram, отправка сообщений пользователям, проверка новостей, возникновение исключительных состояний. Использование данного модуля поможет быстро установить и исправить ошибки в случае их возникновения.

Пакет “handler” содержит функции для обработки запросов:

- функция отправки новости вручную (ссылка на новость задаётся пользователем);
- функция подписки на рассылку;
- функция отмены подписки на рассылку.

Перед тем, как обработать запрос, проверяется заголовок “X-Telegram-Bot-Api-Secret-Token” на соответствие секретному токenu, считанному из “.env” файла.

Каждый запрос обрабатывается в отдельной горутине (goroutine, go-подпрограмма) [3]. По принципу работы горутин схожи с потоками в операционных системах, однако имеют значительные отличия по реализации:

- горутинami управляет встроенный в язык Go планировщик, а потоками – планировщик ОС;
- горутина занимает гораздо меньше оперативной памяти (от 2 КБ, потоки – от 1-2 МБ);
- у горутин время запуска быстрее, чем у потоков.

Таким образом, обработка запросов на многоядерных серверах будет осуществляться параллельно и эффективно благодаря горутинам.

Пакет “parser” содержит функции для парсинга веб-страницы новостей сайта ДонНТУ. Парсинг (parse, разбор) HTML – это обработка HTML документа страницы и извлечение из неё необходимой информации. Для этих целей была использована библиотека “goquery”, которая преобразовывает HTML документ в узлы (nodes), с которыми можно взаимодействовать средствами языка. Поиск нужных узлов осуществляется с помощью CSS-селекторов. CSS-селектор – это шаблон, соответствующий элементам на веб-странице. На рисунке 6 изображен фрагмент кода поиска всех изображений новости (тег “img”), которые расположены в элементе с классом “l-content” (в данном случае div блок, контент самой новости на странице).

```
doc.Find(".l-content").Find("img").Each(func(i int, s *goquery.Selection) {
    src, _ := s.Attr("src")

    if i == 0 {
        titleImg = fmt.Sprintf("<a href=\"%s\">Фото</a>", src)
    } else {
        otherImgs = append(otherImgs, src)
    }
})
```

Рисунок 6 – Фрагмент кода поиска изображений новости

У каждого тега “img” извлекается атрибут “src” – ссылка на изображение. Главное изображение новости в виде тега “a” со ссылкой отправляется вместе с содержанием новости. Остальные изображения ввиду особенностей Bot API добавляются в список и отправляются отдельным сообщением. Чтобы Telegram правильно интерпретировал сообщения с разметкой HTML необходимо в метод отправки сообщения (“sendMessage”) передать параметр “parse\_mode” (режим разбора) со значением “HTML”. Также, необходимо учитывать, что Telegram поддерживает не все теги HTML, а значит, требуется после парсинга содержимого новости удалить лишние теги, либо заменить их нужными символами для приближения к оригинальной разметке.

Пакет “observer” представляет собой функцию “наблюдатель”, которая запускается в отдельной горутине. Данная функция с заданной в “.env” файле периодичностью парсит дату и время последней опубликованной новости на сайте и сравнивает с датой и временем последней рассылаемой новости (хранится в БД). В случае если первая дата позже второй, новость парсится полностью и отправляется всем пользователям, каналам и группам, подписавшимся на рассылку. Дата и время этой новости записываются в БД.

## **Выводы**

Разработанный в соответствии с требованиями Telegram бот автоматизирует процесс рассылки новостей ДонНТУ. Бот был размещён на виртуальный сервер и протестирован. Исходный код доступен в github репозитории “donntu-news-tg-bot” [4].

К недостатком данной реализации можно отнести то, что в случае изменения разметки страницы новостей на сайте ДонНТУ может наблюдаться некорректная работа бота. Возможные улучшения реализации бота могут быть в пакете “parser” и направлены на увеличение гибкости парсинга новостей для уменьшения вероятности возникновения нештатных ситуаций в будущем.

## **Литература**

1. Telegram Bot API Reference [Электронный ресурс]. – Режим доступа: <https://core.telegram.org/bots/api>
2. Automatic Certificate Management Environment (ACME) RFC [Электронный ресурс]. – Режим доступа: <https://datatracker.ietf.org/doc/rfc8555/>
3. Go на практике / Батчер М., Фарина М. – М.: ДМК Пресс, 2017. – 374 с.
4. GitHub репозиторий Telegram бота для оповещения о новостях ДонНТУ [Электронный ресурс]. – Режим доступа: <https://github.com/ultimathul3/donntu-news-tg-bot>

**Олейник В.Ю., Чернышова А.В. Разработка и развертывание Telegram бота для оповещения о новостях ДонНТУ.** В статье представлен краткий обзор мессенджера Telegram и его возможностей для разработчиков. Спроектирован, разработан и размещен на сервер Telegram бот для оповещения о новостях ДонНТУ.

**Ключевые слова:** Telegram, Bot API, язык программирования Go, автоматизированный сбор информации, развертывание.

**Oleinik Vadim, Chernyshova Alla. Development and deployment of Telegram bot for notification of DonNTU news.** The article provides a brief overview of the Telegram messenger and its capabilities for developers. Designed, developed and deployed on the server Telegram bot for notification of DonNTU news.

**Key words:** Telegram, Bot API, Go programming language, automated information collection, deployment.