

## **ИНТЕРНЕТ-СЕРВИС ДЛЯ КОНСТРУИРОВАНИЯ И АДМИНИСТРИРОВАНИЯ TELEGRAM-БОТОВ**

Под ботом понимается программа, которая автоматически выполняет последовательность заранее predetermined действий. Чат-боты – наиболее распространенный подвид ботов, задачей которых является автоматизация разговора с пользователем. Приложение Telegram, в свою очередь, является самым популярным мессенджером в Беларуси [1] и одним из самых популярных мессенджеров в мире с ежемесячной аудиторией более полумиллиарда человек. С марта 2015 года мессенджер предоставляет API для разработки чат-ботов [2]. Возможность автоматизации взаимодействия с клиентами, которую предоставляют чат-боты, привлекает бизнес, однако запуск собственного бота сопряжен с рядом трудностей, которые нацелен решить разработанный мной интернет-сервис для конструирования и администрирования Telegram-ботов.

Основными особенностями, отличающей работу с веб-сервисом от самостоятельного запуска бота, являются:

1. Удобное создание ботов: блочный конструктор диалогов предоставляет визуальный интерфейс, который позволяет пользователям с небольшим опытом программирования или без него создавать и настраивать своих собственных ботов.

2. Быстрая разработка: конструктор на основе блоков ускоряет процесс разработки бота, поскольку пользователи могут просто перетаскивать готовые блоки для создания желаемых функций без написания кода с нуля.

3. Централизованный хостинг: размещение ботов на одной платформе упрощает развертывание и управление, гарантируя, что пользователям не придется беспокоиться о поиске отдельных решений для хостинга своих ботов.

Архитектурно приложение состоит из клиентской и серверной частей, базы данных и демона Docker. Так как блочный конструктор диалогов представляет собой нетривиальный высокоинтерактивный пользовательский интерфейс, для клиентской части была выбрана библиотека React. Для реализации серверной части была выбрана платформа ASP.NET Core. Выбор пал на нее по причине кроссплатформенности и адаптивной архитектуры, которая позволяет разработчику свободно расширять и модифицировать ее функциональность под свои нужды. Коммуникация между клиентом и сервером осу-

ществляется не только с помощью привычного REST, но и по Websockets [3], дабы предоставить пользователям возможность узнавать об изменении состояния бота в реальном времени. В качестве базы данных используется MongoDB. Сделано это главным образом из-за ее гибкости: база данных используется в основном для хранения конфигурации ботов, состоящих из блоков различного типа, обладающих различным набором свойств, для чего прекрасно подходит база, не имеющая жесткой схемы данных. Кроме этого, MongoDB известна своей масштабируемостью. Docker-демон используется для управления экземплярами ботов: каждый экземпляр создается в отдельном контейнере. Данное решение обусловлено поддержкой сервисом пользовательских модулей, которые представляют собой .NET DLL библиотеки. Контейнеры обеспечивают необходимый уровень изоляции, позволяющий выполнять пользовательский код, не опасаясь за работоспособность сервиса.

Разработанный сервис [4], хоть и имеет аналоги, такие как TARS [5] и Botsify [6], выгодно отличается от них за счет адаптивного и субъективно более приятного интерфейса, поддержки пользовательских модулей и наличия русскоязычной локализации, однако сервису недостает интеграции с аналитическими платформами, которая есть у конкурентов. Решение этой проблемы входит в планы дальнейшего развития сервиса, также планируется постепенно внедрять поддержку других платформ (мессенджеров и соцсетей), что позволит пользователям сервиса использовать своих существующих ботов на других платформах, не тратя время на разработку бота под каждую платформу в отдельности.

## ЛИТЕРАТУРА

1. Messaging Apps Popularity [Электронный ресурс] – Режим доступа: <https://eagernomad.com/most-popular-messaging-apps-by-country>. Дата доступа: 26.03.2023.
2. Telegram Bot API [Электронный ресурс] – Режим доступа: <https://core.telegram.org/bots/api>. Дата доступа: 23.03.2023.
3. Websockets API [Электронный ресурс] – Режим доступа: [https://developer.mozilla.org/en-US/docs/Web/API/WebSockets\\_API](https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API). Дата доступа: 29.03.2023.
4. Duckgram [Электронный ресурс] – Режим доступа: <https://github.com/ferdinandutkin/duckgram>. Дата доступа: 23.03.2023.
5. TARS [Электронный ресурс] – Режим доступа: <https://hellotars.com>. Дата доступа: 20.03.2023.
6. Botsify [Электронный ресурс] – Режим доступа: <http://botsify.com>. Дата доступа: 28.03.2023.