

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра информатики и программирования

**РАЗРАБОТКА ЧАТ-БОТА ДЛЯ ИЗУЧЕНИЯ АНГЛИЙСКОГО ЯЗЫКА  
АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ**

студента 4 курса 441 группы

направления 02.03.03 Математическое обеспечение и администрирование  
информационных систем

факультета компьютерных наук и информационных технологий

Беднова Андрея Викторовича

Научный руководитель:

Старший преподаватель

кафедры ИиП

\_\_\_\_\_

Портенко М.С.

подпись, дата

Зав. кафедрой ИиП,

к.ф.-м.н., доцент

\_\_\_\_\_

Огнева М.В.

подпись, дата

Саратов 2022

## ВВЕДЕНИЕ

Чат-бот – это программа, которая способна вести переписку с пользователями в чате, имитируя при этом поведение человека. Она может работать на любой известной платформе, например, Messenger, Telegram, Viber и т.д.

На сегодняшний день чат-боты широко используются в разных сферах. Если несколько лет назад они могли казаться игрушкой или экзотикой, то сейчас пользователи к ним привыкли и начали активно использовать.

Очевидно, что те функции, которые взяли на себя чат-боты, могут быть реализованы (и успешно реализуются) в более привычной форме – через веб-интерфейс сайта или предустановленные приложения. Но в данных формах взаимодействия с пользователем стали заметны следующие недостатки:

- люди устали от классического веб-интерфейса;
- пользователи мобильных устройств неохотно устанавливают что-то новое, экономя ресурсы своих телефонов и планшетов;
- пользователи не хотят тратить время на изучение продукта, долго кликать и разбираться в меню и интерфейсе.

Чат-боты успешно справляются с данными проблемами и активно используются людьми в повседневной жизни. Например, согласно документу, опубликованному командой Telegram, посредством их мессенджера 52 млн человек регулярно используют почти 800 000 активных ботов.

Основным преимуществом данного вида общения с пользователем является упрощенная коммуникация, человеку необязательно посещать сайт, чтобы воспользоваться нужным ему сервисом, достаточно просто открыть чат в мессенджере. Так, например, в России за август 2021 года мессенджером Telegram воспользовалось 37 миллионов человек, а за февраль 2022 года – 54 миллиона. Данное преимущество делает чат-ботов очень удобным инструментом для самообразования, например, для изучения английского языка.

**Актуальность данной работы** подтверждается тем, что большая часть существующих Телеграм-ботов для изучения английского языка, например, @LennyEnglishBot или @AlphaEnglishBot, в данный момент просто не работают, а те, которые работают, имеют очень ограниченный функционал. К примеру, бот @mk\_englishBot умеет спрашивать переводы фраз и присылать их произношение в виде звукового файла, но не умеет запоминать ответы пользователя и, как следствие, не имеет возможности отображать прогресс, также отсутствует возможность изучения грамматики. Стоит отметить, что существуют чат-боты для изучения английского языка, которые являются частью огромных платформ, например, Duolingo, но большая часть контента в них является платной.

**Цель бакалаврской работы** – разработка чат бота на платформе ASP.NET для мессенджера Телеграм для изучения английского языка.

Поставленная цель определила **следующие задачи**:

1. Изучить основы разработки веб приложений на платформе ASP.NET Core;
2. Изучить основы взаимодействия веб приложений посредством API;
3. Изучить шаблон проектирования веб-приложений MVC;
4. Изучить технологии хранения информации на сервере;
5. Изучить библиотеку Telegram.Bot для языка C#;
6. Изучить методики обучения английскому языку;
7. Разработать наиболее подходящую архитектуру для реализации чат бота;
8. Реализовать и протестировать чат бота для мессенджера Телеграм.

**Методологические основы** разработки веб-приложений при помощи фреймворка ASP.NET и их взаимодействия посредством API представлены в работах Э. Лок, Арно Лоре, Патанкар Митхун, А. Фримен.

**Практическая значимость бакалаврской работы** состоит в том, что разработанный чат-бот может использоваться для изучения лексики и грамматики английского языка, а построенная архитектура может быть

использована для быстрой реализации совершенно других расширяемых Телеграм-ботов.

**Структура и объём работы.** Бакалаврская работа состоит из введения, пяти разделов, заключения, списка использованных источников и одиннадцати приложений. Общий объём работы – 85 страниц, из них 60 страниц – основное содержание, включая 19 рисунков, цифровой носитель в качестве приложения, список использованных источников информации – 23 наименования.

## КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

**Первый раздел «Разработка веб-приложений»** посвящен описанию основ разработки веб-приложений при помощи платформы .NET. Рассматривается фреймворк ASP.NET, шаблон проектирования веб-приложений MVC и трехслойная архитектура.

В пункте 1.1 приводятся понятия, связанные с разработкой приложений при помощи ASP.NET Core, а также основные преимущества данного фреймворка.

**Microsoft .NET** – это платформа для создания, развертывания и запуска Web-сервисов и приложений, которая состоит из трех основных частей – общезыковой среды выполнения (common language runtime), иерархического множества унифицированных библиотек классов и компонентной версии ASP, называемую ASP.NET.

**ASP.NET** – это часть технологии .NET, используемая для написания мощных клиент-серверных интернет-приложений.

**ASP.NET Core** – это существенная переработка ASP.NET, это новая кроссплатформенная платформа с открытым исходным кодом для создания современных интернет-приложений.

Используя ASP.NET Core, можно получить следующие фундаментальные преимущества:

- Единообразное проектирования веб-интерфейса пользователя и Web-API
- Готовая к работе в облаке система конфигурации
- Встроенная инъекция зависимостей
- Новый легковесный и модульный конвейер HTTP-запросов
- Возможность размещение в IIS или в собственном процессе
- Построен на .NET Core, который поддерживает параллельное управление версиями приложений.
- Поставляются полностью в виде NuGet пакетов
- Новый и современный инструментарий веб-разработки

- Приложения построенные на ASP.NET Core являются кроссплатформенными и могут запускаться в Windows, Mac и Linux
- Открытый исходный код и ориентированность на сообщество

В пункте 1.2 рассматривается шаблон проектирования веб-приложений MVC и его основные компоненты.

**MVC** – это паттерн проектирования веб-приложений, при котором приложение делится на три отдельных компонента: модель данных приложения, пользовательский интерфейс и логика взаимодействия пользователя с системой, благодаря чему модификация одного из этих компонентов оказывает минимальное воздействие на остальные или не оказывает его вовсе.

Компоненты MVC:

- Модель/Model – представляет собой объектную модель некой предметной области, включает в себя данные и методы работы с этими данными, реагирует на запросы из контроллера, возвращая данные и/или изменяя своё состояние.
- Представление/View – отвечает за отображение информации (визуализацию). Одни и те же данные могут представляться различными способами и в различных форматах.
- Контроллер/Controller – обеспечивает связь между пользователем и системой, использует модель и представление для реализации необходимой реакции на действия пользователя.

В пункте 1.3 приводится описание трехслойной архитектуры приложения и рассматриваются слои.

**Трехслойная архитектура приложений** – это модульная клиент-серверная архитектура, которая состоит из слоя представления, слоя логики и слоя доступа к данным:

- Слой представления – это тот слой, с которым непосредственно взаимодействует пользователь.

- Слой логики – реализует всю необходимую логику приложения, все вычисления.
- Слой доступа к данным – связан с хранением и извлечением данных приложения.

**Второй раздел «Способы передачи данных между веб-сервисами»** посвящен описанию механизмов передачи данных между сервисами таких как интерфейс прикладного программирования (API) и webhook, а также рассматривается платформа ASP.NET Web API.

В пункте 2.1 рассматриваются основы понятия API и его основные достоинства.

Интерфейс прикладного программирования (Application Programming Interface или API) представляет собой совокупность различных инструментов; функций, реализованных в виде интерфейса для создания новых приложений, благодаря которому одна программа может взаимодействовать с другой.

В пункте 2.2 рассматриваются дополнительные средства, предоставляемые модификацией фреймворка ASP.NET под названием ASP.NET Web API.

В пункте 2.3 описывается механизм webhook, а также рассматриваются проблемы, которые он позволяет решить.

**Третий раздел «Технологии хранения информации»** посвящен описанию использованных в работе технологий хранения информации и их основных преимуществ.

В пункте 3.1 приводится описание системы управления базами данных SQL Server, а также ее ключевые особенности.

В пункте 3.2 рассматривается технология работы с данными ADO.NET и описываются основные сущности, которые она предоставляет.

В пункте 3.3 говорится о фреймворке Entity framework core и его основных концепциях.

#### **Четвертый раздел «Используемые технологии и инструментарий»**

посвящен описанию требований к функциональности чат-бота и выбору подходящего инструментария.

Для разработки чат-бота использовался следующий инструментарий:

- Среда разработки: Microsoft Visual Studio 2022;
- Платформа: .NET 5;
- Язык: C# 9.0;
- Веб-фреймворк: ASP.NET Core MVC (шаблон проекта: Web-API);
- Взаимодействие с Telegram Bot API: TelegramBots;
- Доступ к данным: EF Core;
- Контейнер зависимостей: Autofac;
- Сопоставление объектов: AutoMapper;
- СУБД: SQL Server 2019;
- Администрирование БД: SQL Server Management Studio 18;
- Веб-хостинг: Somee.com.

#### **Пятый раздел «Разработка чат-бота для мессенджера Telegram»**

посвящен описанию всех этапов разработки чат-бота для мессенджера Telegram на платформе ASP.NET Core с использованием выбранных технологий, также приводится обзор разработанной архитектуры и рассматривается реализованная функциональность чат-бота.

В пункте 5.1 приводятся обоснования для выбора контейнера зависимостей, а также процесс его подключения к проекту.

В пункте 5.2 объясняется концепция консольного клиента, рассматриваются его преимущества и недостатки, а также описывается процесс его добавления в проект.

В пункте 5.3 объясняется концепция серверного клиента, рассматриваются его преимущества и недостатки, а также описывается процесс его добавления в проект.



В пункте 5.4 описывается разработка классов-моделей и генерация из них базы данных. Также в данном пункте описано создание классов доступа к данным в БД.

В пункте 5.5 говорится о механизме идентификации пользователей бота, а также описывается его реализация в приложении.

В пункте 5.6 описывается разработка класса, который позволяет выполнять отправку, редактирование и удаление сообщений и заключается в себе всю логику по взаимодействию бота с чатом.

В пункте 5.7 описывается разработанный механизм приемником, а также приводится иерархия добавленных классов приемников.

В пункте 5.8 рассказывается о предоставляемой Telegram Bot API возможности добавления кнопок к сообщениям, а также о разработанном на основе этой возможности механизме простого добавления кнопок к сообщениям.

В пункте 5.9 говорится о разработанном обработчике обновлений, который является реализацией концепции «машина состояний», а также описывается принцип работы этого обработчика и приводится граф возможных состояний бота.

В пункте 5.10 описывается разработанная бизнес-логика бота, а также описывается структура классов каждого модуля логики.

В пункте 5.11 приводится обзор разработанной архитектуры, объясняется назначение каждого модуля и приводится описание процесса расширения бота новой функциональностью.

В пункте 5.12 приводится обзор разработанной функциональности чат-бота и демонстрируются примеры использования.

## ЗАКЛЮЧЕНИЕ

В ходе выполнения данной работы были изучены основные принципы разработки веб-приложений на платформе ASP.NET Core, архитектурный паттерн model-view-controller, трехслойная архитектура приложений, обмен данными между приложениями через Web-API и Webhooks, хранение информации при помощи MS SQL Server, а также доступ к этой информации при помощи ADO.NET и EF Core.

Таким образом, в результате данной работы с использованием изученных технологий была разработана архитектура для построения расширяемых чат-ботов со сложной бизнес-логикой и реализован чат-бот для изучения английского языка для мессенджера Telegram.

Чат-боты, несомненно, полезный и нужный инструмент для использования в современных реалиях, когда человеку требуется быстрое получение нужной ему информации без лишних взаимодействий с сайтом или сторонним приложением; гораздо быстрее и удобнее написать команду боту в привычном мессенджере и сразу получить ответ, нежели самостоятельно искать информацию. И, как было показано в данной работе, область применения чат-ботов не ограничивается привычным поиском информации или развлекательной деятельностью, это и удобный, эффективный помощник для изучения нового материала, в данном случае, изучения английского языка.

## Основные источники информации

1. Общие сведения об ASP.NET Core [Электронный ресурс]. –URL: <https://docs.microsoft.com/ru-ru/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-6.0> (Дата обращения 22.05.2022)
2. Лок Э. ASP.Net Core в действии / пер. с англ. Д. А. Беликова. – М.: ДМК Пресс, 2021. – 906 с.: ил.
3. Adam Freeman. Pro ASP.NET Core 3: Develop Cloud-Ready Web Applications Using MVC, Blazor, and Razor Pages. New York: Apress Media, 2020
4. Фримен Адам. ASP.NET Core MVC 2 с примерами на С# для профессионалов, 7-е изд. : Пер. с англ. — СПб. : ООО “Диалектика”, 2019. — 1008 с. : ил. — Парал. тит. Англ
5. Арно Лоре. Проектирование веб-API / Пер. с англ. Д. А. Беликова.– М.: ДМК Пресс, 2020.– 440 с.
6. Pattankar Mithun, Hurbuns Malendra. Mastering ASP.NET Web API. Packt Publishing, 2017.
7. Фримен Адам. Entity Framework Core 2 для ASP.NET Core MVC для профессионалов. Пер. с англ. и ред. Ю.Н. Артеменко — М.: Диалектика, 2019. — 624 с.: ил.