|  |
| --- |
| **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Уральский государственный экономический университет»**  **(УрГЭУ)** |

**КУРСОВАЯ РАБОТА**

**Дисциплина:** Проектирование информационных систем

**Тема: Разработка python бота для Telegram**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Институт информационных технологий управления и информационной безопасности | Студент | | Пиджаков | |
| Дмитрий Сергеевич | | | |
| (ФИО) | | | |
| Направление подготовки  Проектирование информационных систем | Группа | ОЗИВТ-23-2-у | | |
| Руководитель | | | Панов |
| Михаил Александрович, старший преподаватель | | | |
| (ФИО, должность, звание) | | | |
|  |  | | | |
|  |  | | | |

Екатеринбург

2023 г.

СОДЕРЖАНИЕ

[**ВВЕДЕНИЕ** 4](#_Toc153208523)

[**1 ОСНОВНАЯ (ТЕХНОЛОГИЧЕСКАЯ) ЧАСТЬ** 6](#_Toc153208524)

[**2 ОПИСАНИЕ РАЗРАБОТАННОГО ПРИЛОЖЕНИЯ** 7](#_Toc153208525)

[**ЗАКЛЮЧЕНИЕ** 11](#_Toc153208526)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 12](#_Toc153208527)

# **ВВЕДЕНИЕ**

В настоящее время мессенджеры стали неотъемлемой частью нашей повседневной жизни. Один из самых популярных и функциональных мессенджеров - Telegram. Благодаря своим возможностям и гибкости, Telegram стал платформой выбора для разработавшихся и развивающихся ботов.

Боты - это автоматизированные программы, которые могут выполнять различные задачи, отвечать на вопросы пользователей, предоставлять информацию и даже участвовать в играх. Они стали неотъемлемой составляющей мессенджеров, обеспечивая дополнительную функциональность и интерактивность для пользователей. Целью данной курсовой работы является изучение и разработка бота для Telegram с использованием языка программирования Python и его библиотеки python-telegram-bot. В работе будет рассмотрено создание бота с основными функциональными возможностями, такими как автоматические ответы на сообщения, отправка информационных сообщений, выполнение команд и другие действия. Основной упор будет сделан на изучение процесса разработки ботов для Telegram, включая работу с API Telegram, организацию архитектуры бота и передачу данных с использованием обратных вызовов (callbacks).

Также будет уделено внимание вопросам безопасности и масштабируемости бота. В ходе работы будут рассмотрены различные аспекты разработки бота, такие как обработка пользовательских команд, использование базы данных для хранения информации, работа с медиафайлами и другие полезные функции. Также будет представлено руководство пользователя и описание процесса развертывания бота на сервере. Результаты данной курсовой работы могут быть полезными для студентов и разработчиков, заинтересованных в создании своего собственного бота для Telegram. Работа также поможет расширить понимание работы с API Telegram, а также применение практических навыков программирования на платформе Python.

На данный момент на платформе Telegram существует большое количество ботов с различными функциональными возможностями. Они могут быть использованы для различных целей, включая автоматическую рассылку новостей, игры, создание опросов, интеграцию с внешними сервисами и многое другое.

Одним из популярных инструментов для разработки ботов для Telegram является python-telegram-bot. Это библиотека на языке программирования Python, которая облегчает взаимодействие с API Telegram и предоставляет удобные средства для разработки функциональных ботов.

Python-telegram-bot позволяет просто и эффективно создавать ботов для Telegram. Он предоставляет обширный набор функций, включая обработку входящих сообщений, отправку текстовых и медиа-сообщений, работу с клавиатурами и многое другое. Библиотека также имеет подробную документацию и активное сообщество разработчиков, что облегчает процесс разработки и решение возникших проблем.

Python-telegram-bot также обеспечивает безопасность бота, предоставляя возможность конфиденциального хранения токенов доступа API и обработки различных типов входящих сообщений. Это позволяет разработчикам создавать безопасные и надежные боты для пользователя.

Лучшие практики разработки ботов для Telegram включают следующие аспекты:

1. Структурирование и организация кода: Организуйте код вашего бота в модульную архитектуру с использованием классов и функций. Разделение функций и задач бота на отдельные модули облегчит поддержку и масштабирование кода.
2. Отлавливание ошибок: Убедитесь, что ваш бот обрабатывает возможные ошибки и исключения, чтобы предотвратить аварийное завершение или неправильное поведение.
3. Хранение данных: Разработайте систему хранения данных, чтобы сохранять информацию о пользователе, истории диалогов и других данных, необходимых вашему боту. Используйте базу данных или другие облачные сервисы для сохранения данных.
4. Реагирование на сообщения: Бот должен быть способен обрабатывать различные типы сообщений от пользователя, включая текстовые сообщения, команды, медиафайлы и местоположение. Учтите все возможности API Telegram для обработки различных типов сообщений.
5. Обратные вызовы (callbacks): Используйте обратные вызовы для осуществления взаимодействия с пользователем и предоставления дополнительных деталей или функций. Обратные вызовы позволяют создавать интерактивные элементы, такие как кнопки и меню.
6. Пользовательский интерфейс: Создайте дружественный и интуитивно понятный пользовательский интерфейс с помощью встроенных клавиатур или меню. Это поможет пользователям взаимодействовать с ботом и использовать его функции.

# **1 ОСНОВНАЯ (ТЕХНОЛОГИЧЕСКАЯ) ЧАСТЬ**

Таким образом, разработка бота для Telegram является актуальной и перспективной темой для курсовой работы, которая позволяет изучить основы разработки ботов, получить опыт работы с API мессенджера Telegram и применить полученные знания в контексте бизнеса или создания инновационных решений.

Так был выбран язык программирования «Python», который используется для создания сайтов, игр, программного обеспечения, машинного обучения.  
Чтобы реализовать эту идею нужно было выбрать Фреймворк или готовый движок.   
Для создания телеграмм бота разработчиками были разработаны API под названием Telebot. Ниже представлена установка его в проект.



Рисунок 1 — установка Telebot.

Так же для того что бы наш Python файл определил бота и подключился нужно в специальном боте от телеграмм получить его API. Получается API через бота под названием BotFather.

Вводи в него следующие команды:

1. /start – запускаем бота в телеграмме
2. /newbot – создаём нового бота
3. Bot name – пишем имя бота

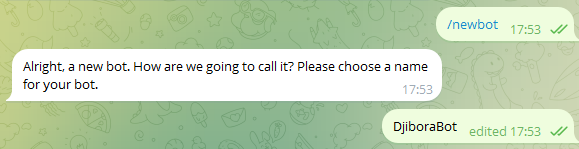


Рисунок 2 — Создание бота

# **2 ОПИСАНИЕ РАЗРАБОТАННОГО ПРИЛОЖЕНИЯ**

Сам бот состоит из набора декораторов и команд в telebot. Создаём начало бота

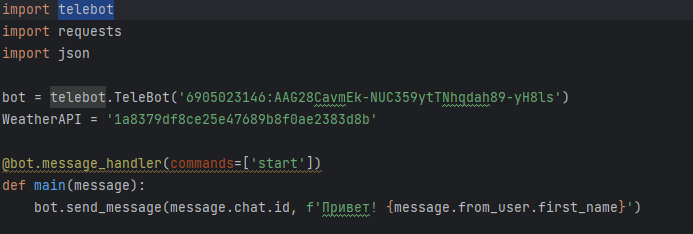


Рисунок 3 – Начало бота

На рисунке 3 видно импортирование telebot в проект далее подключение его API для связи и декоратор @bot.message\_handler далее внутри декоратора прописана команда start. Теперь при сохранении заходим в бота и пишем /start

Он выдаст нам привет и имя пользователя. По сути мы создали команду для бота с названием start на которую он выдаёт ответ.

Далее в боте были созданы следующие функции:

1. status;
2. start;
3. help;
4. weather.

Status отвечает за вывод информации о боте и его работоспособности. Работает он или нет. Рисунок представлен ниже.

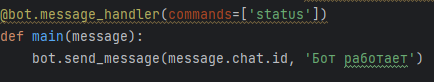


Рисунок 4 – Команда status

Команда help выводи какой набор команд доступен для бота и что можно в него ввести

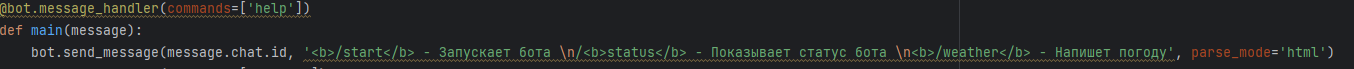


Рисунок 4 – Команда help

Weather команда которая отвечает за вывод погоды в выбранном городе как на русском так и на английском языке.

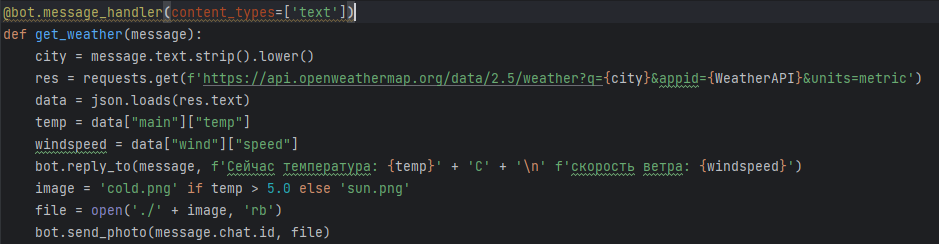




Рисунок 5 – Команда weather

Так же в боте реализована выполнение команд в не заранее заготовленных команд.

Пользователь может написать например слово «помощь» в любом регистре и бот выдаст «/help».

На основе полученных данных с сайта телеграмм для его запуска требуется следящая конфигурация :

1. Операционная система: Windows ХР/Виста/7/8/8.1/10
2. Память (ОЗУ): Требуется 512 МБ оперативной памяти
3. Место на жестком диске: Требуется 70 МБ свободного места
4. Процессор: Двухъядерный процессор Intel или более поздняя версия

# **ЗАКЛЮЧЕНИЕ**

В заключение, разработка бота для Telegram является актуальной и перспективной темой, которая предоставляет множество возможностей и преимуществ. Telegram является популярным мессенджером с большой аудиторией пользователей, что обеспечивает широкие возможности для взаимодействия с ними. Боты в Telegram позволяют автоматизировать задачи, предоставлять информацию, взаимодействовать с пользователями и предоставлять различные сервисы.

Разработка бота для Telegram с использованием python-telegram-bot является удобным и эффективным подходом. Модуль python-telegram-bot предоставляет множество функций для работы с ботами, таких как обработка команд, сообщений, клавиатур и взаимодействие с Telegram API. Это обеспечивает разработчикам удобную среду для создания и настройки ботов с минимальными усилиями.

Разработка бота для Telegram может представлять интерес для студентов, желающих расширить свои навыки программирования и изучить работу с API мессенджера Telegram. Python-telegram-bot является удобным инструментом для разработки ботов на языке Python, что позволяет студентам на практике применить полученные знания программирования и создать полноценное решение для Telegram.

Таким образом, разработка бота для Telegram является актуальной и перспективной темой, которая предоставляет множество возможностей для создания инновационных решений, улучшения пользовательского опыта и расширения навыков программирования.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Введение в визуальные новеллы, часть первая [Электронный ресурс] /Animemanga — Япония, аниме, манга. — Режим доступа: <http://animemaga.ru/news/vvedenie-vizualnie-novelli-chast-pervaya-8349> ;
2. Визуальный роман [Электронный ресурс] / Википедия — свободная энциклопедия. — Режим доступа: <https://ru.wikipedia.org/wiki/Визуальный_роман>;
3. Галенкин С. Маркетинг игр [Текст] / С. Галенкин. — СанктПетербург: Питер, 2014. — 78 с;
4. Как создать визуальную новеллу самостоятельно: советы и рекомендации [Электронный ресурс] / FB.ru. — Режим доступа: <http://fb.ru/article/385903/kak-sozdat-vizualnuyu-novellu-samostoyatelno-sovetyi-i-rekomendatsii> ;
5. Adobe Photoshop [Электронный ресурс]. — Режим доступа:https://www.adobe.com/ru/products/photoshop.html ;
6. Microsoft Word [Электронный ресурс]. — Режим доступа: <https://products.office.com/ru-ru/word> ;
7. The Ren'Py Visual Novel Engine [Электронный ресурс]. — Режим доступа: <https://www.renpy.org> ;
8. The Visual Novel Database [Электронный ресурс]. — Режим доступа: <https://vndb.org/v4> ;
9. Типы визуальных новелл. – URL: <http://bit.ly/2lHu7Ov>;