

Образовательный Телеграм-бот для обучения по программе математики 11 класса

Информатика
Песчанский Д.А.

10 класс, ГБОУ МО "Одинцовский "Десятый лицей", г. Одинцово
Научный руководитель: Пименова О.Р., учитель информатики, ГБОУ МО
"Одинцовский "Десятый лицей", г. Одинцово

ВВЕДЕНИЕ

Актуальность проекта.

В наше время многие соцсети подвергаются блокировкам или хакерским атакам, и многие пользователи переходят на платформу Телеграмма, что сильно повышает её популярность. Аудитория Telegram ежедневно растёт с геометрической прогрессией, этому способствует удобство мессенджера, наличие каналов, чатов, и конечно возможность создавать ботов. Боты могут использоваться в совершенно разных целях, от автоматизации бизнес-задач до специализированных виртуальных учителей [1].

Проблема.

В сфере образования необходимость проводить онлайн обучение в период пандемии высветила немало проблем и показала необходимость поиска новых эффективных форм взаимодействия обучающихся и педагогов, применения новых медиа информационных технологий. Одной из таких новинок стало использование чат-ботов, как элемента игрового обучения.

Цель проекта.

Создать образовательного Телеграмм-бота для обучения по программе математики 11 класса.

Задачи проекта.

- Изучить материал по созданию Телеграм-ботов.
- Выбрать программное обеспечение для создания Телеграм-бота.
- Продумать логику и интерфейс бота.
- Зарегистрировать бота и установить на удаленный сервер.
- Написать основную логику работы бота в виде программного кода.
- Протестировать бота.

Методы исследования:

- Анализ научной и научной-популярной литературы и систематизация информации;
- Изучение технической документации;
- Создание прототипа.

Новизна проекта:

Мной не было найдено ботов, связанных с темой “Математика 11 класс”. Также был разработан собственный алгоритм тестирования пользователя.

Наиболее конкурентоспособными и теми, кого можно поставить в один ряд, можно назвать чат-боты таких корпораций как Slack, Skype, Viber, Telegram. Все они интегрируются в мессенджеры и имеют открытое API и готовые платформы для разработки, имеют возможность выбора языка программирования для создания чат-бота, что увеличивает аудиторию заинтересованных в разработке пользователей и не требует вложения ресурсов для изучения новой технологии.

Сравнивая платформу Telegram со Slack, Skype, Viber, можно выделить такие отличительные черты в пользу Telegram, как облачное хранение всех данных переписок, которое реализовано только в Telegram, что убережет пользователей от потери важных данных, например, если у пользователя Telegram был бот, через который производилась оплата, данные о транзакциях не будут утеряны при переходе на другое устройство, чего не скажешь о Viber,

Slack, Skype. Так же в Telegram существует двухфакторная аутентификация пользователей, что делает использование этой платформы более защищенным.

У Telegram есть собственное зашифрованное облачное хранилище, распределенное по разным юрисдикциям, и оно защищено гораздо лучше, чем хранилища Google и Apple. Что делает возможным использование ботов в корпоративных целях, гарантируя приватность данных, чем не может похвастаться Slack мессенджер, который используется в основном для корпоративного общения, в котором вся история переписки может быть сохранена и передана третьим лицам, по такому же принципу, с точки зрения приватности, работает Skype. Telegram позволяет пользователям иметь доступ к чатам сразу с нескольких устройств одновременно, благодаря облачной синхронизации. Таким образом, пользователи Mac, ПК, iPad и даже сервера на Linux получают один и тот же опыт взаимодействия с мессенджером, исключениями являются только секретные чаты, для которых не применяется сохранение данных в облачные хранилища.

В отличие от Viber и Skype Telegram предлагает своим пользователям продвинутую функциональность, например, групповые чаты до 10000 участников или каналы, которые может читать неограниченное количество пользователей и которые имеют возможность интеграции чат-ботов в них. Подобные технологии не могут быть реализованы с использованием парадигмы «E2E-шифрование плюс бэкапы у третьих лиц» [4]. End-to-end шифрование – это система, в рамках которой, зашифрованная информация передается от устройства к устройству напрямую, без посредников. Правила закрытого ключа не позволяют расшифровать информацию никому, кроме её получателя. Таким образом, зашифровка и расшифровка сообщений происходят без участия сервера. Также, следует отметить, что сегодня Telegram используется гораздо чаще в повседневной жизни, нежели Skype и Slack, поэтому и использование чат-ботов более востребовано. У чат-ботов Telegram очень доступное API, использование которого является бесплатным и создать своего чат-бота может

любой пользователь, что и породило большое комьюнити людей, разрабатывающих под платформу Telegram, что также является неоспоримым плюсом.

Подведя итог, можно сказать, что использование платформы Telegram для создания чат-ботов является наиболее оптимальным, меньше затрат на разработку, больше времени на создание интеллектуального сервиса, анализ данных и обучение. Платформа Telegram является наиболее перспективной для развития и создания чат-ботов [2].

Программное обеспечение для разработки бота

Язык программирования Python.

Python [3] — высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ. Язык является полностью объектно-ориентированным — всё является объектами. Необычной особенностью языка является выделение блоков кода пробельными отступами. Данный язык программирования является интерпретируемым, что негативно сказывается на скорости его работы. Python является мультипарадигмальным языком программирования. Аспектно-ориентированное программирование частично реализовано в этом языке с помощью декораторов.

Сегодня активно развивается версия языка Python 3. Разработка языка ведётся через предложения по расширению языка PEP (англ. Python Enhancement Proposal), в которых описываются нововведения, делаются корректировки согласно обратной связи от сообщества и документируются итоговые решения. Стандартная библиотека включает большой набор полезных переносимых функций, начиная от функционала для работы с текстом и заканчивая средствами для написания сетевых приложений. Дополнительные

возможности, такие как математическое моделирование, работа с оборудованием, написание веб-приложений или разработка игр, могут реализовываться посредством обширного количества сторонних библиотек, а также интеграцией библиотек, написанных на С или С++, при этом и сам интерпретатор Python может интегрироваться в проекты, написанные на этих языках.

Python стал одним из самых популярных языков, он используется в анализе данных, машинном обучении, DevOps и веб-разработке, а также в других сферах, включая разработку игр. За счёт читабельности, простого синтаксиса и отсутствия необходимости в компиляции язык хорошо подходит для обучения программированию, позволяя концентрироваться на изучении алгоритмов, концептов и парадигм. Отладка же и экспериментирование в значительной степени облегчаются тем фактом, что язык является интерпретируемым. Применяется язык многими крупными компаниями, такими как Google. По состоянию на март 2021 года Python занимает третье место в рейтинге TIOBE популярности языков программирования с показателем 10,31 % [4].

База данных PostgreSQL

PostgreSQL [5] — это популярная свободная объектно-реляционная система управления базами данных. PostgreSQL базируется на языке SQL и поддерживает многочисленные возможности.

Преимущества PostgreSQL:

- поддержка БД неограниченного размера;
- мощные и надёжные механизмы транзакций и репликации;
- расширяемая система встроенных языков программирования и поддержка загрузки С-совместимых модулей;
- наследование;

- легкая расширяемость.

Библиотека Aiogram.

Aiogram предоставляет доступ к API (интерфейс прикладного программирования) мессенджера Telegram. Это HTTP-интерфейс, созданный для разработчиков, увлеченных созданием ботов для Telegram. Обработка событий (команд от пользователя) реализована с помощью специальных команд языка Python – декораторов. Каждое сообщение от пользователя возвращается в виде словаря в формате JSON файла, в котором хранится вся информация о сообщении (текст, ID отправителя, наличие вложенных файлов и т. д.). Среди разработчиков aiogram считается лучшей библиотекой для написания ботов на Python. Она асинхронная, использует декораторы и содержит удобные инструменты для разработки [6].

PyCharm — это интегрированная среда разработки для Python, которая имеет полный комплект средств, необходимых для эффективного программирования на Python. PyCharm имеет удобный редактор кода со всеми полезными функциями: подсветкой синтаксиса, автоматическим форматированием, дополнением и отступами. PyCharm позволяет проверять версии интерпретатора языка на совместимость, а также использовать шаблоны кода.

Регистрация и установка на сервер

Для начала работы мне было необходимо зарегистрировать своего бота и получить уникальный токен. Для этого надо написать специальному чат-боту **@BotFather**. Это бот позволил мне настроить моего бота: создать имя, установить аватарку бота, настроить систему команд и в конечном итоге зарегистрировать бота. Затем требовалось подобрать платформу по предоставлению серверов. Мой выбор пал на сервис под названием **Heroku**. Он довольно быстрый, у них удобный интерфейс для настройки сервера, у него большое кол-во хороших отзывов, а самое главное он бесплатный, что позволяет уменьшить затраты на выполнение проекта. Воспользовавшись инструкцией на

сайте платформы, я установил бота, а также базу данных на удаленный сервер: это дало возможность работать моему боту на постоянной основе.

Программная реализация работы

Инициализация бота

Для начала я импортировал (подключил) библиотеки `asynpcrg`(для работы с БД) и `aiogram`(для написания бота) с помощью оператора `import`. Далее инициализировал бота с заранее созданным токеном в чат-боте `@BotFather`. Для хранения токена я использовал переменные окружения, они помогают надежно хранить данные, которые являются ключами и паролями для доступа к работе бота. А затем подключил их. В главном файле инициализировал бота с помощью токена, который я подключил.

Написание кода для взаимодействия с базой данных

Для данного проекта нужна база данных, в которой будут храниться теоретический материал, упражнения, тесты, а главное сведения о пользователях. В первую очередь я инициализировал подключение к базе данных. Это делается следующими строками. Затем я написал код для получения информации о пользователях из БД. Далее мне было необходимо реализовать код для получения упражнений, тестов и материалов теории.

Обработка сообщений

Следующим шагом мы объявляем декоратор, для сообщений, содержащих команду `/start`. Данная функция будет отправлять кнопки с главным меню. Главное меню состоит из 4 пунктов:

- Прогресс по курсу (показывает в процентах какую часть курсу уже прошел человек)
- Продолжить обучение (отправляет материал теории, на которой пользователь остановился в предыдущий раз)

- Сброс курса
(сбрасывает курс, пользователь начинает с самого начала)
- Интересные факты из мира Математики
(отправляет интересные истории и факты из мира математики, но пока находится в доработке)

Следующим шагом я объявил декоратор, для сообщений, содержащих команду /theory. Бот будет выполнять функцию и отправлять предупреждение о неопределённой команде и попросит отправить боту команду /start. Затем я объявил декоратор, для сообщений, содержащих команду /theory. Данная функция будет отправлять кнопки с разделами теории, которые пользователь уже прошел. Это позволит ему повторить пройденный материал, если он захочет.

Функционал для обработки тестов

После каждого пройденного блока курса пользователь должен будет проходить тесты, чтобы проверить на сколько хорошо он усвоил материал.

Алгоритм работы тестирования:

1. Вывод инструкции о прохождении теста. (начало тестирования)
2. Команда /start_test начинает тестирование.
3. Затем пользователю отправляются задания и начинается отсчет времени, которое было дано на выполнение теста.
4. Если пользователь успел сделать задания до конца времени, то ему отсылаются сообщения с помощью, которых он вводит свои ответы на задания, иначе форма для заполнения ответов отправится после истечения времени.
5. Затем бот анализирует полученные ответы и на основе этого отправляет фрагменты теории, где приведен конкретный пример, чтобы пользователь разобрался с примером, в котором допустил ошибку. Если пользователь сделал все правильно, то бот отправляет его в следующий раздел теории.

6. Конец тестирования

Запуск бота. В заключении мы запускаем функцией `executor.start_polling`. При запуске всем администратором, которые следят за работой бота (на данный момент это только я) отсылается уведомление о запуске бота с помощью функции `on_startup`. Также устанавливаются команды по умолчанию для взаимодействия с ботом.

Экономическая и экологическая составляющие (таб.1)

Таблица 1. Расчёт затрат, возникших вследствие работы

	<i>Цена за месяц</i>	<i>Количество месяцев</i>	<i>Сумма</i>
Интернет	50 р.	5	250 р.
Электричество	65 р.	5	325р.
Бумажные носители	26 р.	5	130 р.
Канцелярия	30 р.	5	150 р.
Подписка на пользование средой разработки	400 р.	3	1200 р.
ИТОГО:		2 055 р.	

Экологичность

В результате выполнения проекта, создается программный продукт, который не оказывает на окружающую среду никакого вредного воздействия. Использование ЭВМ не производит загрязнения биосферы, так как отсутствуют выбросы газов, паров, дыма и аэрозолей в атмосферу, загрязнение сточных вод побочными продуктами промышленного и сельскохозяйственного

производства, загрязнение почвы твёрдыми и жидкими отходами, загрязнение окружающей среды болезнетворными микроорганизмами, а электромагнитные излучения от ЖК мониторов отсутствуют.

Утилизация технологических приборов и материалов так же не представляет собой сложности и не наносит вред окружающей среде. Устаревшие модели ЭВМ заменяются на более новые, а старые разбираются, отсортировываются по наличию драгоценных металлов и перерабатываются.

Выводы

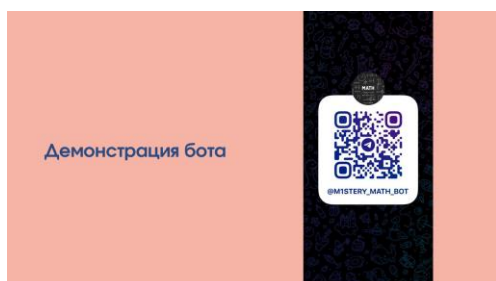
Запустив приведенный выше код и проанализировав работу представленного бота, я сделал вывод о том, что программная реализация исполнена корректно. Однако в процессе тестирования было выявлено, что необходимо доработать некоторые моменты, а именно:

- 1) Сделать красивый и форматированный вывод теоретического материала;
- 2) Написать алгоритм для отслеживания активности использования бота;
- 3) Сделать обработчик ошибок, чтобы в случае, когда пользователь делает что-то не так, он получал подсказку от бота.
- 4) Создать функционал, который будет высылать интересные факты из мира математики в перерывах между занятиями.

Моя разработка — это бот, который поможет изучить материал старших классов по математике. Основная идея сводится к тому, что обучение с помощью бота очень удобное и занимает немного времени, то есть человек в свободное время может зайти в мессенджер, ответить людям, которые ему написали, а также провести время с пользой, получая знания благодаря моему боту. (рис.1) Принцип работы моего бота очень прост:

- Новому пользователю сначала предлагается пройти «проверку знаний», которая поможет подобрать уровень теории и упражнений.

- С помощью кнопок меню и команд бота можно перемещаться по его функционалу.
- После пройденного материала теории пользователь должен выполнить упражнения, после которых он или перейдет к следующей теме, или ему будет отправлен материал конкретной темы, в упражнениях по которой он допустил ошибки.
- Также в перерывах между занятиями пользователю будут высылаются интересные факты и новости из мира математики. **Рисунок 1. Демонстрация Телеграм-бота.**



Список литературы

1. Смыслова, Л. В. Чат-бот как современное средство интернет-коммуникаций / Л. В. Смыслова. — Текст : непосредственный // Молодой ученый. — 2018. — № 9 (195). — С. 36-39.
2. Документация для работы с Telegram-ботами – <https://core.telegram.org/bots/api>
3. Yogesh Rana. Python: Simple though an Important Programming language // International Research Journal of Engineering and Technology (IRJET). — 2019. — 2 February (vol. 06, iss. 2). — P. 1856—1858.
4. Документация языка программирования Python – <https://www.python.org/doc/>
5. Документация для работы с базой данных PosgreSQL <https://www.postgrespro.ru/docs/postgresql>

6. Документация для работы с Python библиотекой aiogram

<https://github.com/aiogram/aiogram>