

Сбор данных может быть организован в виде таймера или прерывания, особенно если необходимо обеспечить стабильную частоту выборки. Это особенно важно при измерении ЭКГ или при вычислении variability сердечного ритма (HRV), где даже незначительное нарушение интервалов может исказить результат.

Полученные данные можно сразу анализировать на ESP32, либо временно сохранять и отправлять для последующей обработки на внешний сервер или устройство через Wi-Fi, Bluetooth или по UART. Таким образом, сбор данных представляет собой не просто считывание показаний, а организованный поток цифровой информации, с которым можно работать в реальном времени или асинхронно.

Предобработка и анализ кардиосигнала. Предобработка сигнала – ключевой этап в системе анализа кардиоданных. Сырые данные с датчиков (аналоговых, таких как Pulse Sensor и AD8232, или цифровых, например MAX30102) содержат шумы, артефакты движения, дрейф базовой линии и другие искажения. Для очистки применяют цифровую фильтрацию: полосовые фильтры пропускают только нужный диапазон частот (обычно 0,5–4 Гц для ЧСС), фильтры нижних и верхних частот устраняют шумы и дрейф, а метод скользящего среднего дополнительно сглаживает сигнал. При резких выбросах используют пороговую фильтрацию, заменяя некорректные значения предыдущими корректными. Нормализация амплитуды упрощает визуализацию и передачу данных. Для сложных сигналов, таких как ЭКГ, применяются более продвинутые методы: медианные фильтры, фильтры Баттерворта, коррекция базовой линии и контроль равномерности выборки. Эти методы позволяют преобразовать искажённый сигнал в стабильные данные, пригодные для дальнейшего анализа.

Анализ кардиоданных. Анализ начинается с определения частоты сердечных сокращений (ЧСС) путём поиска пиков (например, R-пиков в ЭКГ) и расчёта интервалов между ними ($bpm = 60 / \text{интервал в секундах}$). Для повышения надёжности ЧСС усредняют по нескольким ударам.

Более глубокий уровень анализа – variability сердечного ритма (HRV), отражающая работу вегетативной нервной системы. HRV рассчитывается на основе последовательности RR-интервалов, с использованием показателей SDNN, RMSSD и частотного анализа (LF / HF).

Дополнительно форма ЭКГ-сигнала позволяет выявлять аритмии, экстрасистолы, тахикардии и другие патологии, анализируя морфологию комплексов QRS и интервалы между ними.

Передача и визуализация данных. Обработанные данные можно передавать и визуализировать через Wi-Fi или Bluetooth. ESP32 отправляет данные на серверы по MQTT, HTTP или WebSocket, обеспечивает отображение графиков в реальном времени либо передаёт данные на смартфоны через BLE. При необходимости ESP32 может работать как локальный веб-сервер, отображая пульс и другие показатели без доступа в интернет.

Литература

1 ESP32 – Википедия. – URL: <https://ru.wikipedia.org/wiki/ESP32> (дата обращения: 15.04.2025).

УДК 004.9

Д. С. Чернокал

О СОЗДАНИИ ТЕЛЕГРАМ-БОТА ДЛЯ ИНТЕРНЕТ-МАГАЗИНА

В статье рассматриваются вопросы разработки телеграм-бота для интернет-магазина. Описываются основные функции бота. Приводится схема работы

программы. Отмечаются принципы разработки телеграм-бота, отмечена практическая значимость программного обеспечения и его универсальность. Намечены перспективы развития разработки.

Создание Telegram-ботов в наше время приобретает всё большее значение как инструмент автоматизации бизнес-процессов, особенно в области электронной коммерции. В условиях динамичного рынка интернет-магазины стремятся использовать удобные и эффективные решения для улучшения общения с клиентами. Telegram-боты позволяют не только быстро обрабатывать пользовательские запросы, но и предоставлять актуальную информацию о товарах и заказах, а также выполнять множество дополнительных задач.

В связи с этим актуальной становится задача разработки программного обеспечения, которое реализует полноценный функционал Телеграм-бота, использующего библиотеку Aiogram 3 [1], которая предоставляет широкий спектр возможностей для создания интерактивных и многофункциональных ботов, способных решать разнообразные задачи и адаптироваться под потребности бизнеса. В статье подробно рассмотрены основные этапы разработки Телеграм-бота для управления товарами в интернет-магазине.

Особенности разработки Телеграм-бота. Одной из ключевых функций Телеграм-бота для управления товарами в интернет-магазине является просмотр товаров, что обеспечивает пользователям удобный и интуитивно понятный интерфейс для взаимодействия с ассортиментом магазина. Для реализации данного функционала разработан алгоритм, который можно разделить на несколько этапов.

На первом этапе создается объект бота с использованием уникального токена, предоставленного BotFather – официального сервиса платформы Телеграм для управления ботами. Этот токен служит идентификатором бота и обеспечивает безопасность взаимодействия между ботом и платформой Телеграм.

После создания объекта бота настраивается диспетчер (Dispatcher), который отвечает за обработку команд и сообщений от пользователей. Диспетчер связывает получаемые данные с соответствующими обработчиками, позволяя боту выполнять различные действия в зависимости от типа входящего сообщения. Таким образом, диспетчер становится центральным элементом, который управляет логикой работы бота и его взаимодействием с пользователями.

Создание базы данных. Для хранения данных о товарах, пользователях и их заказах используется база данных SQLite [2]. Этот выбор обусловлен тем, что SQLite является легкой, встроенной и простой в использовании системой управления базами данных (СУБД), идеально подходящей для небольших и средних проектов. Благодаря своей компактности и отсутствию необходимости в отдельном сервере, SQLite позволяет быстро развернуть базу данных без сложных настроек, что существенно облегчает процесс разработки.

Создание и инициализация базы данных являются важными этапами разработки, так как они позволяют организовать структуру хранения информации. В рамках данного проекта структура базы данных включает таблицы для хранения информации о товарах, пользователях и их заказах. Это позволяет эффективно разделять и структурировать данные, обеспечивая быстрый доступ к нужной информации.

Определение структур данных для товаров и пользователей позволяет формализовать информацию и упростить процесс её обработки. Каждый товар может иметь поля, такие как наименование, описание, цена, наличие на складе и другие характеристики, что даёт возможность пользователям получать полную информацию о каждом товаре.

Схема работы программы. После настройки базы данных и инициализации бота запускается основной цикл работы программы. Этот цикл непрерывно прослушивает входящие сообщения и команды от пользователей, обрабатывая их в реальном времени. Схема работы Телеграм-бота приведена на рисунке 1.

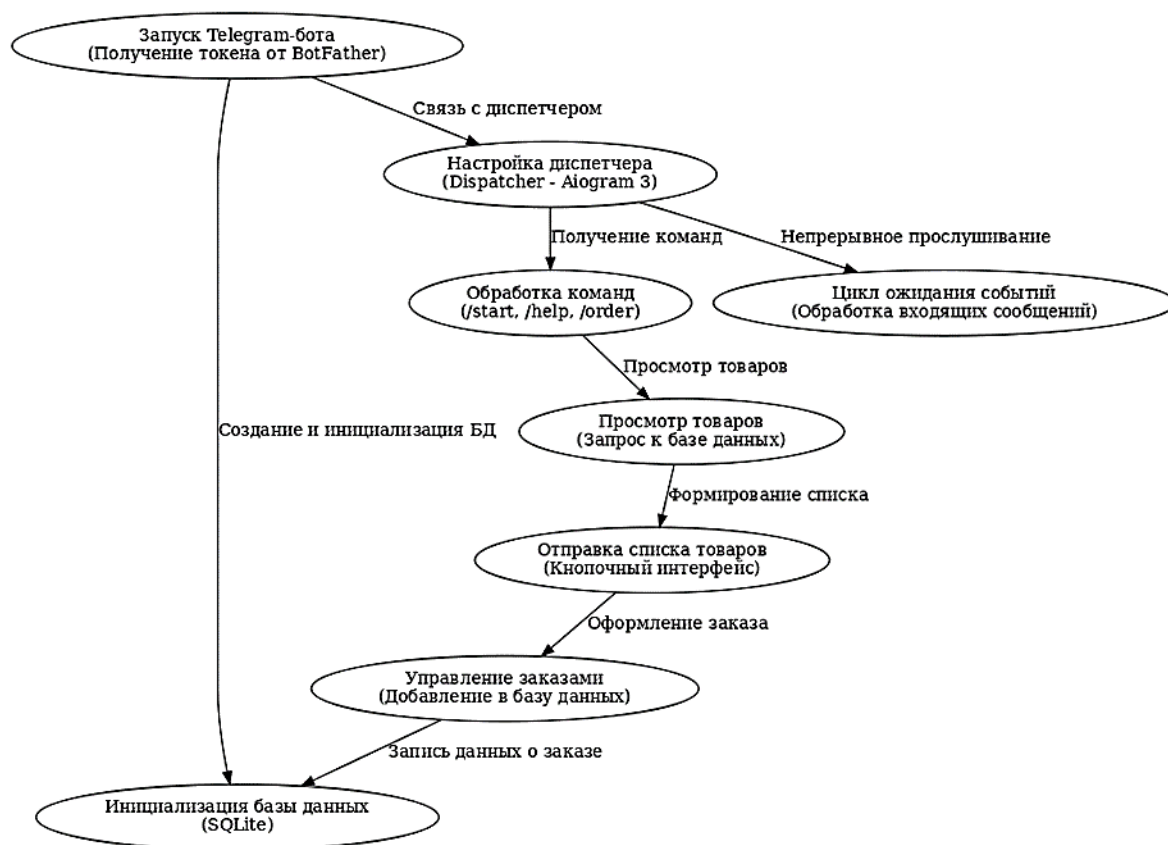


Рисунок 1 – Схема работы Телеграм-бота

Основной цикл обеспечивает постоянную доступность бота, что является важным аспектом для интернет-магазинов, где пользователи могут отправлять запросы в любое время суток. Это позволяет обеспечить высокий уровень обслуживания клиентов и быстро реагировать на их запросы.

Когда пользователь отправляет команду /start, диспетчер вызывает соответствующий обработчик. Этот обработчик отвечает за отправку приветственного сообщения и инструкций по использованию бота, что помогает пользователям быстро ориентироваться в функционале.

Кроме того, могут быть предусмотрены дополнительные команды, такие как /help, /order и другие, которые помогают пользователям находить нужную информацию и использовать возможности бота на полную мощность.

Одной из функций бота является просмотр товаров. Когда пользователь выбирает соответствующую опцию, бот запрашивает информацию из базы данных о доступных товарах.

Затем бот формирует список товаров, который отправляется пользователю. Список может содержать название товара, его цену, краткое описание и другие параметры, такие как наличие на складе и особенности доставки. Часто используется кнопочный интерфейс, что позволяет пользователю выбрать определённый товар для получения дополнительной информации или оформления заказа. Это значительно улучшает пользовательский опыт и делает процесс покупки более удобным.

Для повышения скорости и удобства навигации по ассортименту бота был встроен механизм поиска товаров. Пользователь может ввести ключевое слово или фразу, после чего бот выполнит поиск по базе данных и вернёт список товаров, соответствующих запросу. Поиск осуществляется по наименованию и описанию товаров с использованием SQL-запроса с оператором LIKE. Это позволяет находить нужные позиции даже при неполном совпадении слов, что существенно улучшает пользовательский опыт.

Для управления ассортиментом без необходимости ручного доступа к базе данных была реализована административная панель внутри самого бота. Авторизованные администраторы могут добавлять новые товары, редактировать информацию о существующих позициях, а также удалять устаревшие или недоступные товары. В целях безопасности данная функция доступна только пользователям с определённым `user_id`, внесённым в список администраторов. Ввод данных при добавлении и редактировании реализован пошагово с использованием FSM, что позволяет избежать ошибок при вводе и делает процесс максимально удобным.

Также была реализована система категорий, которая структурирует ассортимент магазина и облегчает пользователям поиск нужных товаров. При взаимодействии с ботом пользователь сначала выбирает интересующую категорию, после чего получает список товаров, относящихся к данной группе. Категории отображаются в виде кнопок, что делает интерфейс более визуально удобным и современным. Внутри базы данных для каждого товара предусмотрено поле `category`, что позволяет гибко управлять структурой ассортимента и легко добавлять новые группы товаров.

Созданный многофункциональный Телеграм-бот имеет интуитивно понятный пользовательский интерфейс. Бот включает в себя структуру хранения данных, что значительно упрощает доступ к информации о товарах и позволяет быстро обновлять ассортимент. Кроме того, бот обеспечивает удобство управления заказами и может служить основой для более сложных проектов.

Одним из возможных направлений развития такого бота является интеграция с платежными системами, что позволяет пользователям оформлять и оплачивать заказы прямо в мессенджере. Это открывает новые горизонты для автоматизации процессов интернет-магазина и улучшения пользовательского опыта, позволяя делать покупки более удобными и быстрыми.

Практическая значимость разработки заключается в том, что бот предоставляет удобный инструмент для управления покупками, а также демонстрирует основы работы с библиотекой `Aiogram`. Это делает его полезным не только для конечных пользователей, которые могут легко получать информацию о товарах и оформлять заказы, но и для разработчиков, которые изучают создание Телеграм-ботов.

Полученные знания и опыт могут быть использованы для разработки более сложных систем автоматизации в различных областях, включая, но не ограничиваясь, сферой услуг, образования и развлечений.

Литература

1 Aiogram 3.19.0 documentation: [сайт]. – aiogram Team, 2025. – URL: <https://docs.aiogram.dev/en/v3.19.0/#> (дата обращения: 24.03.2025).

2 SQLite documentation: [сайт]. – 2025. – URL: <https://www.sqlite.org/docs.html> (дата обращения: 24.03.2025).

УДК 004.522:004.62

В. А. Шкарубо

ИНТЕГРАЦИЯ РАСПОЗНАВАНИЯ РЕЧИ И ОБЛАЧНЫХ СЕРВИСОВ В НАСТОЛЬНЫХ ПРИЛОЖЕНИЯХ ДЛЯ ОБРАБОТКИ МЕДИЦИНСКИХ ДАННЫХ

В статье рассматривается опыт разработки настольного приложения для работы с медицинскими данными, в котором реализованы технологии распознавания речи и облачного хранения. Приводится архитектура программного решения, описываются