

Утверждаю
Старший разработчик

_____ А. Данилов
(подпись, инициал имени фамилия)

«__» _____ 2023г.

Техническое задание на разработку Telegram бота

Разработал

_____ С. Кривченков
(подпись, инициал имени фамилия)

«__» _____ 2023г.

1. Глоссарий терминов.

Термин	Описание
API	(аббр. от англ. Application Programming Interface) — описание способов взаимодействия одной компьютерной программы с другими.
cURL	cURL — библиотека с открытым исходным кодом, используемая для отправки HTTP-запросов с различных языков программирования в том числе и PHP
HTTPS	(аббр. от англ. HyperText Transfer Protocol Secure) — расширение протокола HTTP для поддержки шифрования в целях повышения безопасности.
HTTP	(англ. HyperText Transfer Protocol — «протокол передачи гипертекста») — протокол прикладного уровня передачи данных, изначально — в виде гипертекстовых документов в формате HTML, в настоящее время используется для передачи произвольных данных.
HTML	(от англ. HyperText Markup Language — «язык гипертекстовой разметки») — стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере. Веб-браузеры получают HTML документ от сервера по протоколам HTTP/HTTPS или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора.

PDO	расширение для PHP, предоставляющее разработчику универсальный интерфейс для доступа к различным базам данных.
Бот	специальный аккаунт в Telegram, созданный для того, чтобы автоматически обрабатывать и отправлять сообщения, комплекс программных средств работающих с таким аккаунтом посредством веб-хука или другими способами. Пользователи могут взаимодействовать с ботами при помощи сообщений, отправляемых через обычные или групповые чаты. Логика бота контролируется при помощи HTTPS запросов к API для ботов.
веб-хук	(англ. webhook) в веб-разработке — метод расширения или изменения поведения веб-страницы или веб-приложения с помощью обратных вызовов. Эти обратные вызовы могут поддерживаться, изменяться и управляться сторонними пользователями и разработчиками, которые не обязательно связаны с исходным веб-сайтом или приложением.
пространство имен	(англ. namespace) — некоторое множество, под которым подразумевается модель, абстрактное хранилище или окружение, созданное для логической группировки уникальных идентификаторов (то есть имен).
токен	программный токен (токен доступа, токен приложения) - строка из набора случайных или заданных по определенному правилу символов, служащих для замены доступа к аккаунту по логину и паролю. Обычно

	выдаются приложению на ограниченный или бессрочный период действия.
--	---

Прочая техническая терминология понимается в соответствии с действующими стандартами и рекомендациями международных органов, ответственных за вопросы стандартизации в сети Интернет.

2. Общие положения.

2.1. Предмет разработки.

Предметом разработки является создание telegram бота (по указанной ссылке представлена [документация Telegram](#)) работающего с заказами сайта интернет магазина.

2.2. Цель разработки telegram бота.

- работа с заказами сайта-магазина (отображение списка, изменение статуса, удаление);
- уведомление пользователя о новых заказах на сайте;
- получение информации о заказе из базы данных;
- удаление заказа из базы данных;
- изменение статуса заказа (новый/выполнен);
- фильтрация полученного списка заказов по заданным параметрам:
 - 1) статус (все заказы, только новые, только выполненные);
 - 2) период (день, неделя, месяц);
 - 3) товар (получение списка заказа содержащего определенный товар).

3. Требования к сайту и программному обеспечению.

3.1. Требования к базе данных:

Дамп тестовой базы данных предоставляется.

3.2. Требования к программному обеспечению.

Набор технологий должен отвечать решению бизнес-задач заказчика и способствовать реализации указанного в данном документе функционала. Набор

технологий утверждается заказчиком на основании предложений исполнителя. Рекомендуемый набор для реализации задачи:

- Веб-сервер Apache (или Nginx)
- База данных MySQL;
- PHP 7+;
- Telegram-клиент (мессенджер).

3.3. Требования к лингвистическому обеспечению.

Проект не является мультиязычным.

3.4. Требования к поддержке и техническому обслуживанию.

Работы по устранению некритичных ошибок осуществляются в рабочем порядке с приоритетом над работами с новым функционалом (если иное не будет оговорено с бизнес-заказчиком). Не критичными считаются ошибки, которые позволяют реализовывать пользователю основные действия на сайте (просмотр основных разделов, покупку).

3.5. Дополнительные требования заказчика.

- при формировании пользователем заказа, реализовать моментальное оповещение последнего, посредством telegram бот вне зависимости от количества пользователей совершающих однотипное действие;
- предусмотреть и заложить возможность развития дополнительных возможностей telegram_бот таких как: редактирование состава заказов, отображение информации о покупателях, отображение статистики и аналитики по продажам, а также реализация функционала уведомлений покупателей и общения с ними непосредственно через чат бота.
- реализация исполнителем технического задания в обязательном порядке включает удобочитаемость кода, сопутствующую документацию и возможность его поддержки разработчиками;
- при разработке использовать базу данных заказчика (пример дампа базы данных предоставляется, описание таблиц и полей указано в Приложении 1).

4. Страница заказа интернет магазина.

Реализовать функцию или метод для создания нового заказа, который можно будет вызывать из любого места кода. Создать тестовую страницу с формой заказа для демонстрации работы этого метода.

5. Разработка функционала telegram бота.

5.1. Уведомление о новом заказе.

Реализовать возможность пользователю получать уведомления от telegram бота о совершении нового заказа в интернет магазине. Получаемое пользователем уведомление в обязательном порядке должно содержать следующую информацию и кнопки клавиатуры:

- “Новый заказ № N” - где N отображение уникального номера (идентификатора ID) заказа;
- “Товар” - отображение названия товара;
- “Количество” - отображение количества товара содержащегося в заказе;
- “Цена” - отображение стоимости единицы товара на момент оформления заказа;
- “Сумма” - отображение общей стоимости указанного в заказе товара (рассчитывается по формуле: $C * K = S$, где C - это цена единицы товара, K - количество одноименного товара указанного в заказе, S - сумма т.е. полученный результат).
- кнопка “Новый” - предназначена для отображения значения статуса отображаемого заказа, а также для его изменения (реализация описана в разделе 5.5);
- кнопка “Удалить” - предназначена для предоставления пользователю возможности удаления заказа (реализация описана в разделе 5.6).

5.2. Вывод информации о заказе.

Реализовать возможность для пользователя получить информацию о заказе. При передаче команды обязательным параметром указать идентификационный номер заказа.

Предусмотреть ошибочные действия пользователя:

- Пользователь не указал номер заказа - вывести сообщение об ошибке: “Необходимо указать ID заказа”;
- Пользователь указал несуществующий номер заказа - вывести сообщение об ошибке: “Заказ с ID не существует”.

Получаемое пользователем уведомление в обязательном порядке должно содержать следующую информацию и кнопки:

- “Заказ № N” - где N уникальный номер (идентификатор ID) заказа;
- “Товар” - отображение названия товара;
- “Количество” - отображение количества товара содержащегося в заказе;
- “Цена” - отображение стоимости единицы товара на момент оформления заказ;
- “Сумма” - отображение общей стоимости указанного в заказе товара (рассчитывается по формуле: $C \cdot K = S$, где C - это цена единицы товара, K - количество одноименного товара указанного в заказе, S - сумма т.е. полученный результат);
- “Создан” - отображение даты и времени создания заказа;
- “Изменен” - отображение даты и времени внесения последнего изменений в заказ.
- кнопка “Выполнен” - предназначена для отображения значения статуса отображаемого заказа, а также для его изменения (реализация описана в разделе 5.5);
- кнопка “Удалить” - предназначена для предоставления пользователю возможности удаления заказа (реализация описана в разделе 5.6).

Получение данных осуществить путем передачи пользователем текстовой команды. Рекомендуемый вид команды запроса данных: “/order ID”. Где ID - это числовой идентификатор товара.

5.3. Вывод списков заказов

Реализовать возможность для пользователя получить список заказов. Пользователю отображается 10 последних заказов, отсортированных в обратном порядке по дате создания (первыми отображаются более поздние заказы).

Получаемое пользователем уведомление в обязательном порядке должно содержать следующую информацию:

- “Заказ № N” - где N уникальный номер (идентификатор ID) заказа;
- “Сумма” - отображение общей стоимости указанного в заказе количества товара (рассчитывается по формуле: $C * K = S$, где C - это цена единицы товара, K - количество одноименного товара указанного в заказе, S - сумма т.е. полученный результат);
- “Создан” - отображение даты и времени создания заказа;

Под каждым заказом разместить ссылку-команду на получение подробной информации о заказе содержащую номер заказа. При клике пользователем по указанной ссылке отображается подробная информация о заказе описанная в разделе 5.2.

Получение данных осуществить путем передачи пользователем текстовой команды. Рекомендуемый вид команды запроса данных: “/orders”

5.4. Фильтр списка заказов.

Реализовать возможность для пользователя осуществить фильтрацию существующих в базе данных заказов по заранее заданным параметрам (новый, выполнен, день, неделя, месяц, ID товара) с последующим отображением результата в формате вывода списка заказов описанного ранее в п. 5.3.

значение параметров:

- новый - заказы со статусом новый;
- выполнен - заказы со статусом выполнен;
- день - все заказы за текущий день (текущие сутки начиная с 00 часов 00 минут);
- неделя - все заказы со дня предшествующего текущему на семь дней;

- месяц - все заказы со дня предшествующего текущему на 30-31 день в зависимости от календарного значения;
- ID - все заказы с указанным идентификатором товара.

Получение данных осуществить путем передачи пользователем текстовой команды с соответствующими по необходимому запросу параметрами.

Рекомендуемый вид команды запроса данных путем фильтрации: `"/orders [new|done] [today|week|month] [ID]"`. Где new - статус заказа "Новый", done - статус заказа "Выполнен", today - период "день", week - период "неделя", month - период "месяц", ID - идентификатор товара. Квадратные скобки показывают необязательность параметра. Вертикальная черта показывает альтернативность использования параметров.

5.5. Смена статуса заказа

Реализовать для пользователя возможность смены статуса заказа. Кнопка для смены статуса отображается под сообщением с подробной информацией о заказе (описана в разделах 5.1 и 5.2).

При нажатии пользователем на эту кнопку, осуществляется смена указанного статуса на противоположный (выполнен/новый) с соответствующими изменениями статуса заказа в базе данных, кроме того при нажатии пользователем по кнопке изменяется как название кнопки клавиатуры так и в строке "Изменен" отображаются дата и время внесения изменений.

5.6. Удаление заказа.

Реализовать для пользователя возможность удаления заказа используя telegram бот. Удаление заказа производить при нажатии пользователем кнопки "Удалить" (описана в разделах 5.1 и 5.2) в два этапа: первый - предложить пользователю подтвердить свое действие на удаление записи о заказе из базы данных, второй - непосредственно удаление заказа.

При клике пользователем по кнопке удалить - отображается сообщение содержащее:

- текст “Удалить заказ № N” - где N - номер заказа выбранного пользователем для удаления;
- кнопка “Да” - при клике пользователем по кнопке осуществляется удаление из базы данных записи о заказе с последующим отображением сообщения содержащего текст “Заказ № N удален” вместо сообщения-вопроса о подтверждении, где N - номер заказа выбранного пользователем для удаления;
- кнопка “Отмена” - при клике пользователем по кнопке, сообщение о подтверждении - “Удалить заказ № N” исчезает.

5.7. Список команд.

Используемые текстовые команды на получение требуемых результатов в обязательном порядке должны быть оформлены и прописаны в сопроводительной документации для пользователей и разработчиков.

Приложение 1. Описание таблиц и полей базы данных

Тестовая база данных предоставляется в формате дампа sql. База содержит таблицы products и orders.

Таблица products предназначена для хранения товаров. Поля таблицы:

- id - уникальный идентификатор заказа
- name - наименование товара
- price - цена товара
- image - ссылка на файл с изображением товара

Таблица orders предназначена для хранения заказов:

- id - уникальный идентификатор заказа
- product_id - идентификатор заказываемого товара
- product_name - наименование товара такое каким оно было на момент заказа
- product_count - количество товара, заказанное покупателем
- product_price - цена товара на момент заказа
- created_at - дата и время создания заказа
- modified_at - дата и время изменения заказа
- status - статус заказа или его состояние - новый/выполнен, принимает одно из значений - 0 новый, 1 выполнен