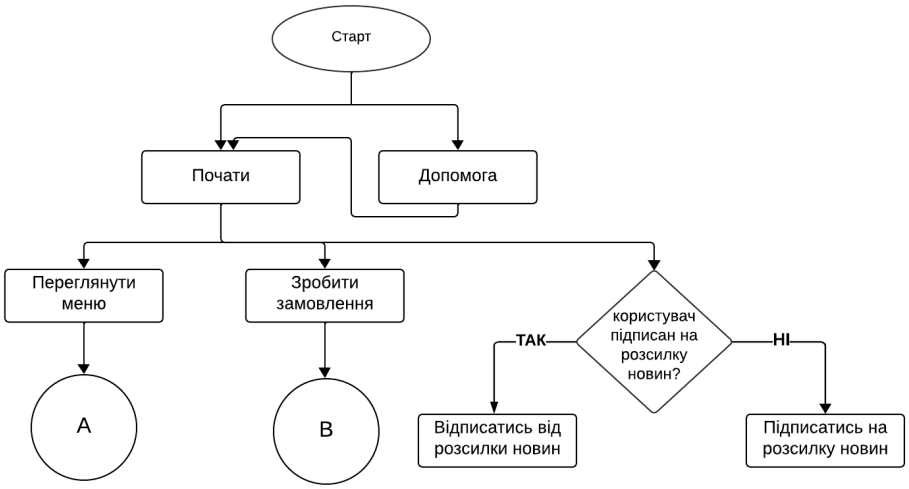
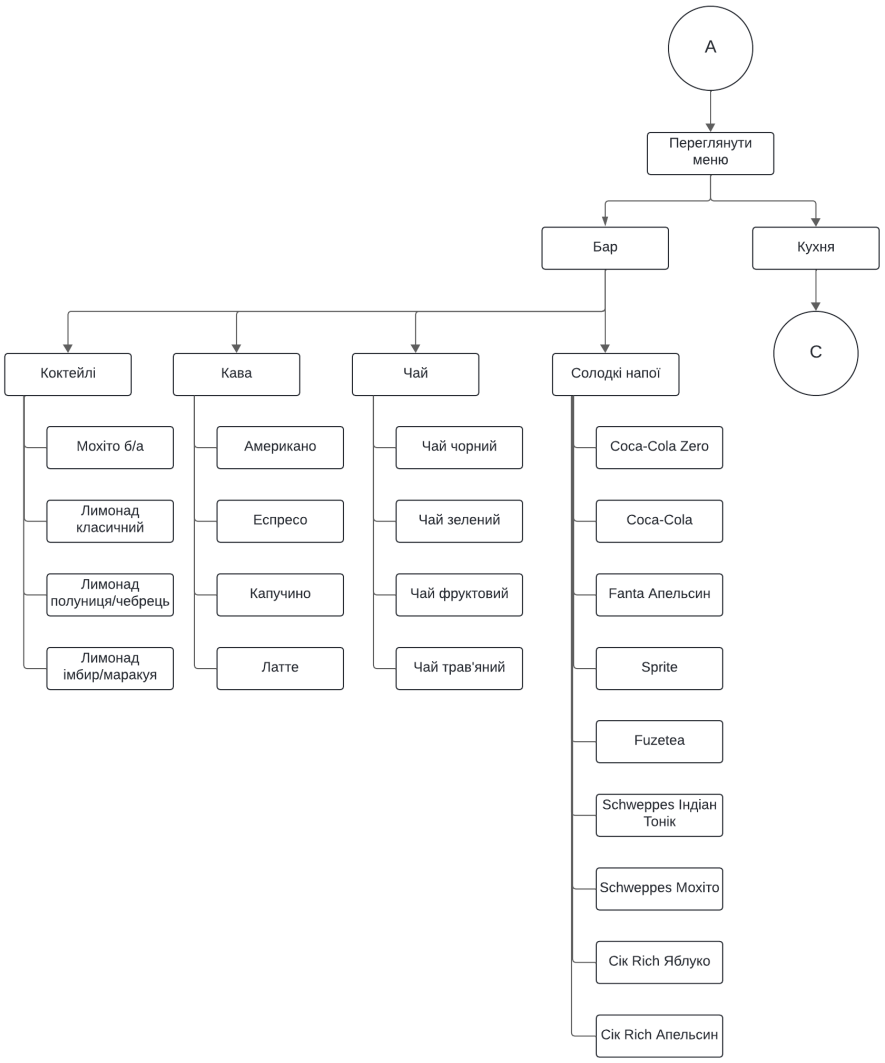
ДОДАТКИ

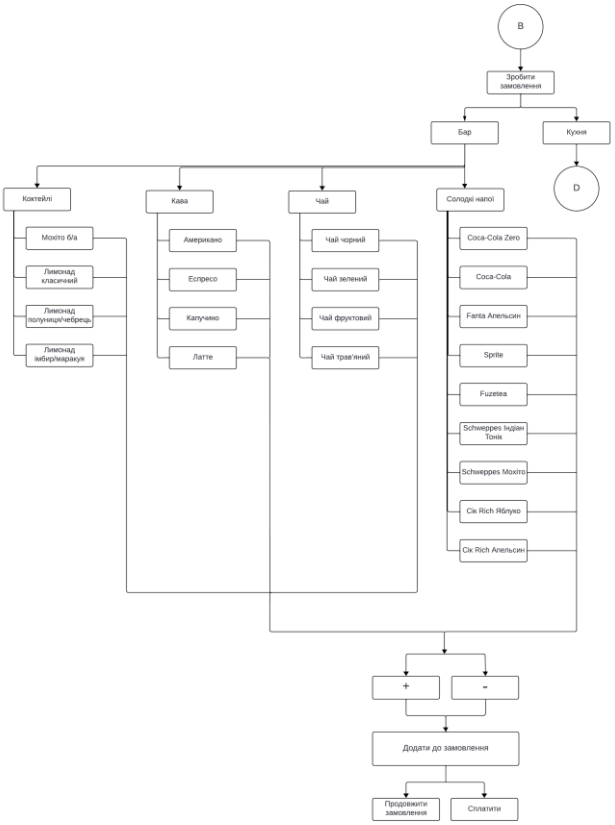
ДОДАТОК А

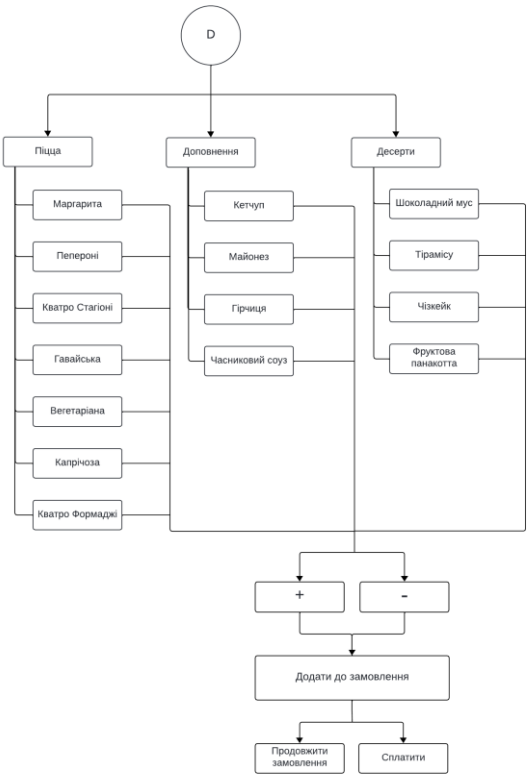
(Довідковий)

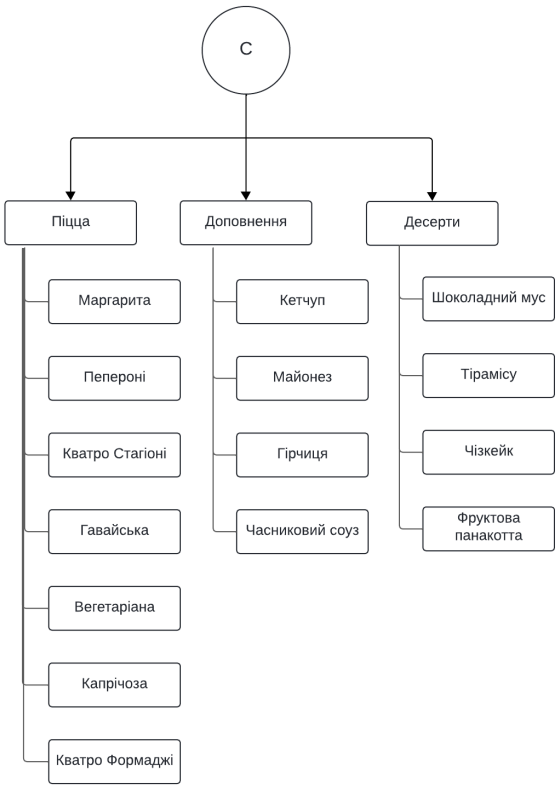
АЛГОРИТМ РОБОТИ ТЕЛЕГРАМ БОТУ


Старт
Почати
Допомога
Переглянути меню
Зробити замовлення
користувач підписан на розсилку новин?
ТАК
НІ
A
B
Відписатись від розсилки новин
Підписатись на розсилку новин


A
Переглянути меню
Бар
Кухня
C
Коктейлі
Кава
Чай
Солодкі напої
Мохіто б/а
Лимонад класичний
Лимонад полуниця/чебрець
Лимонад імбир/маракуя
Американо
Еспресо
Капучино
Латте
Чай чорний
Чай зелений
Чай фруктовий
Чай трав'яний
Coca-Cola Zero
Coca-Cola
Fanta Апельсин
Sprite
Fuzetea
Schweppes Індіан Тонік
Schweppes Мохіто
Сік Rich Яблуко
Сік Rich Апельсин


B
Зробити замовлення
Бар
Кухня
D
Коктейлі
Кава
Чай
Солодкі напої
Мохіто б/а
Лимонад класичний
Лимонад полуниця/чебрець
Лимонад імбир/маракуя
Американо
Еспресо
Калучино
Латте
Чай чорний
Чай зелений
Чай фруктовий
Чай трав'яний
Coca-Cola Zero
Coca-Cola
Fanta Апельсин
Sprite
Fuzetea
Schweppes Індіан Тонік
Schweppes Мохіто
Сік Rich Яблуко
Сік Rich Апельсин
+
-
Додати до замовлення
Продовжити замовлення
Сплатити

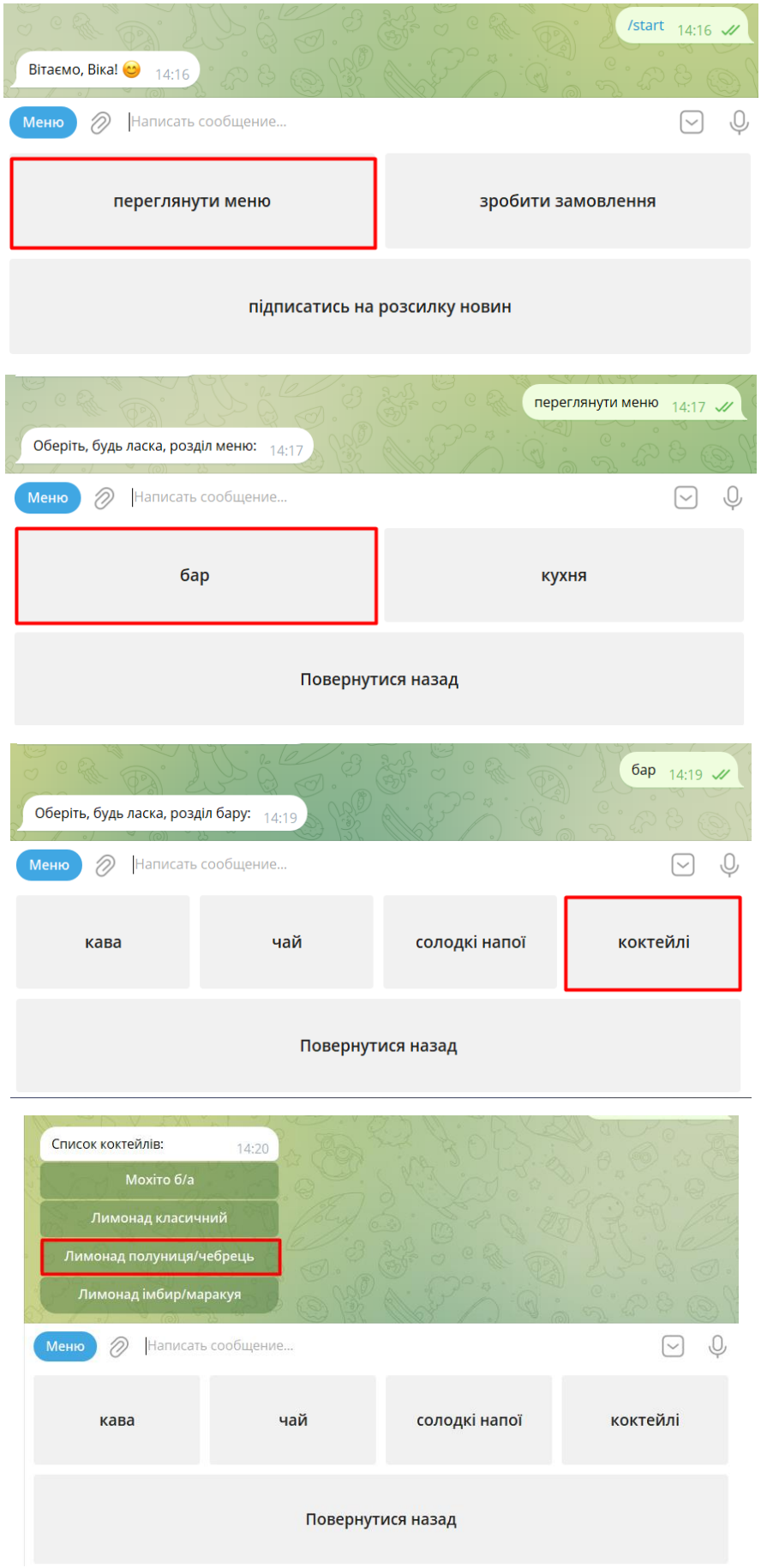

D
Піцца
Доповнення
Десерти
Маргарита
Пепероні
Кватро Стагіоні
Гавайська
Вегетаріана
Капрічоза
Кватро Формаджі
Кетчуп
Майонез
Гірчиця
Часниковий соус
Шоколадний мус
Тірамісу
Чізкейк
Фруктова панакотта
+
-
Додати до замовлення
Продовжити замовлення
Сплатити

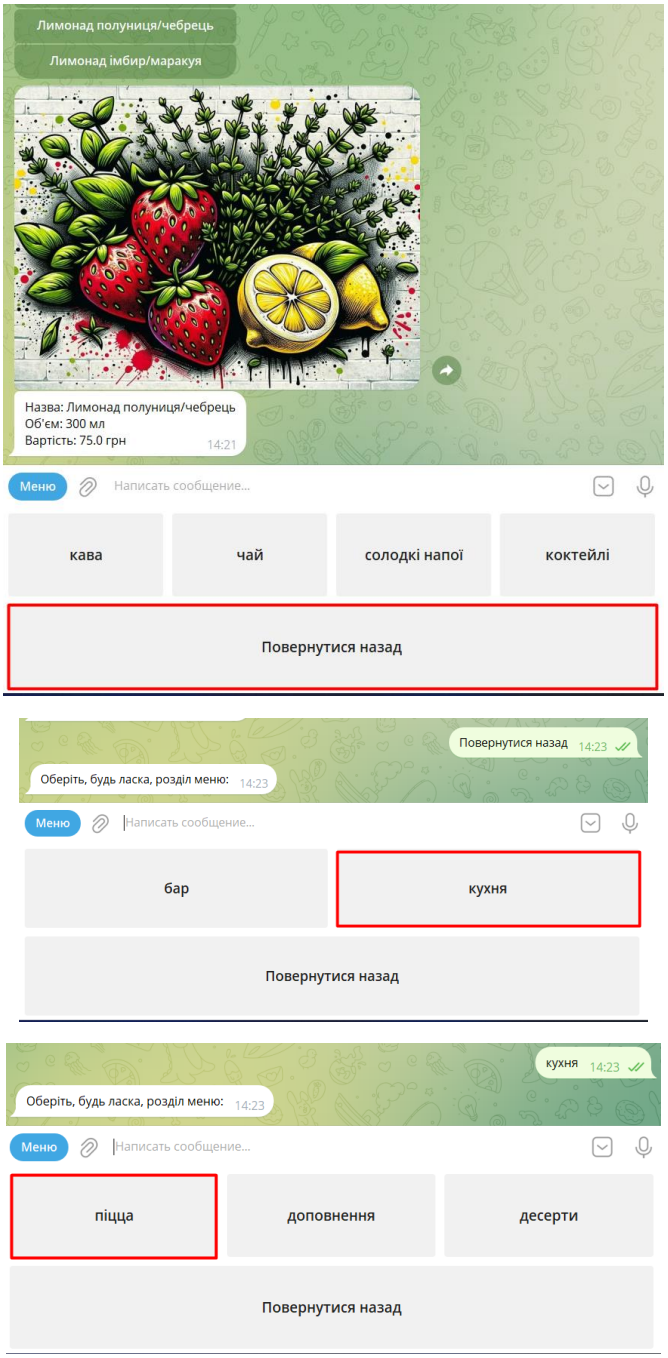

C
Піцца
Доповнення
Десерти
Маргарита
Пепероні
Кватро Стагіоні
Гавайська
Вегетаріана
Капрічоза
Кватро Формаджі
Кетчуп
Майонез
Гірчиця
Часниковий соус
Шоколадний мус
Тірамісу
Чізкейк
Фруктова панакотта

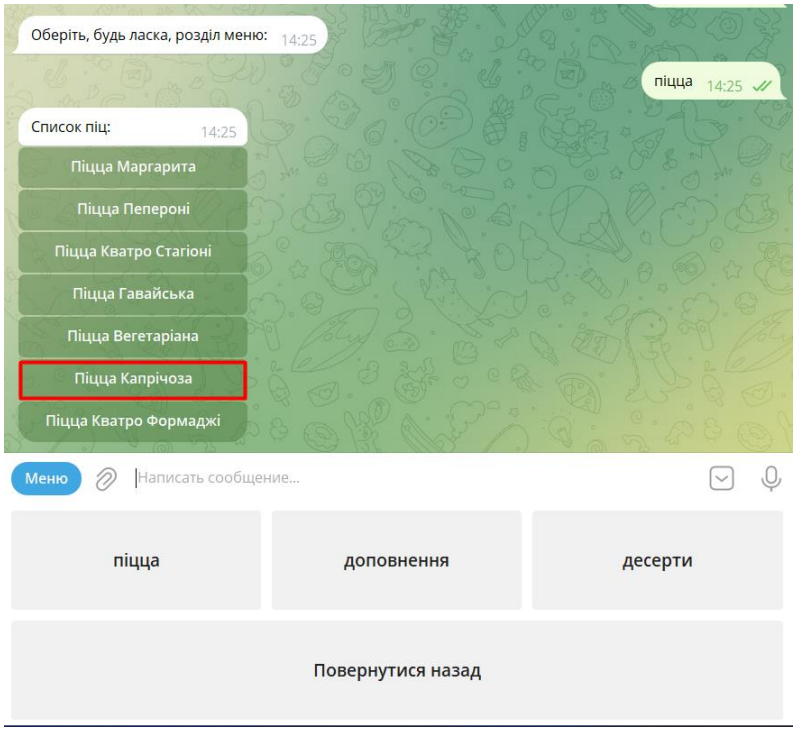
ДОДАТОК Б

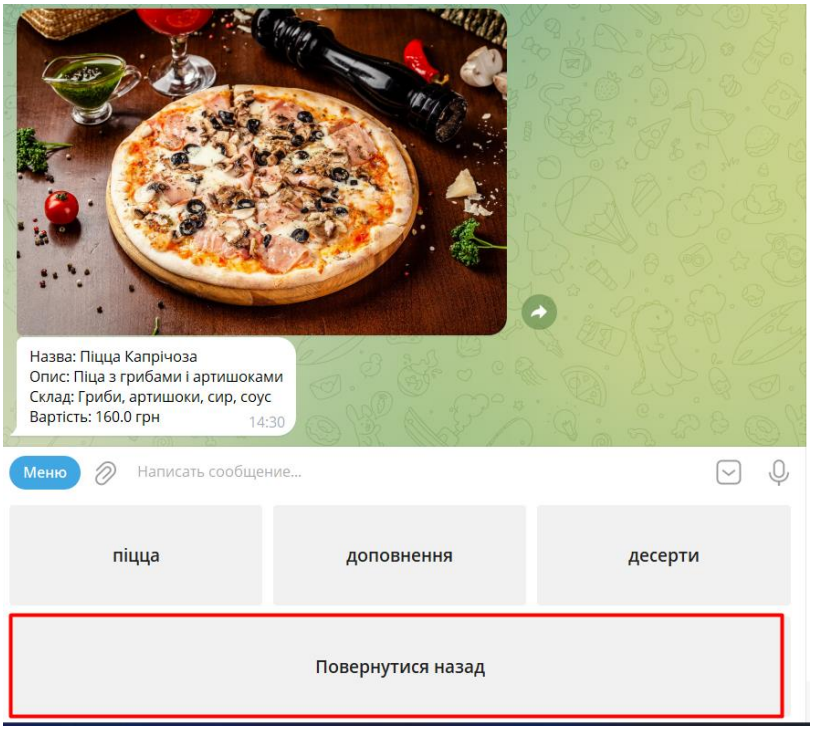
(Довідковий)

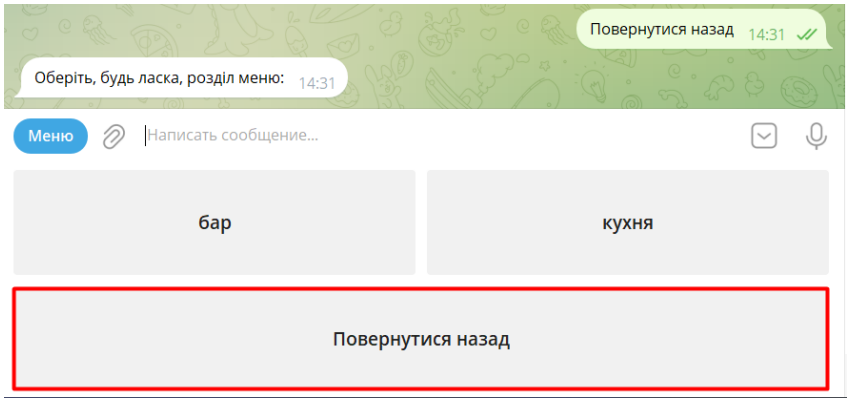
ПРИКЛАД РОБОТИ ТЕЛЕГРАМ БОТУ

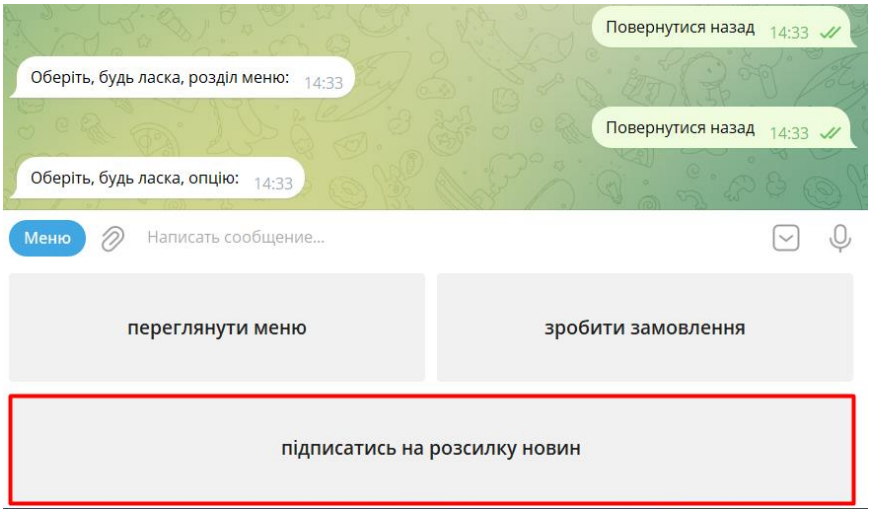

/start 14:16
Вітаємо, Віка! 14:16
Меню
Написать сообщение...
переглянути меню
зробити замовлення
підписатись на розсилку новин
переглянути меню 14:17
Оберіть, будь ласка, розділ меню: 14:17
Меню
Написать сообщение...
бар
кухня
Повернутися назад
бар 14:19
Оберіть, будь ласка, розділ бару: 14:19
Меню
Написать сообщение...
кава
чай
солодкі напої
коктейлі
Повернутися назад
Список коктейлів: 14:20
Мохіто б/а
Лимонад класичний
Лимонад полуниця/чебрець
Лимонад імбир/маракуя
Меню
Написать сообщение...
кава
чай
солодкі напої
коктейлі
Повернутися назад

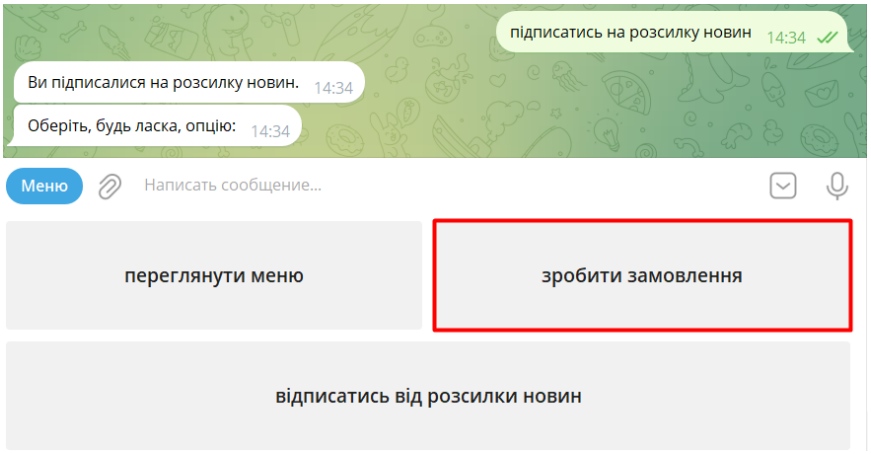

Лимонад полуниця/чебрець
Лимонад імбир/маракуя
Назва: Лимонад полуниця/чебрець
Об'єм: 300 мл
Вартість: 75.0 грн
14:21
Меню
Написать сообщение...
кава
чай
солодкі напої
коктейлі
Повернутися назад
Повернутися назад 14:23
Оберіть, будь ласка, розділ меню: 14:23
Меню
Написать сообщение...
бар
кухня
Повернутися назад
кухня 14:23
Оберіть, будь ласка, розділ меню: 14:23
Меню
Написать сообщение...
піцца
доповнення
десерти
Повернутися назад

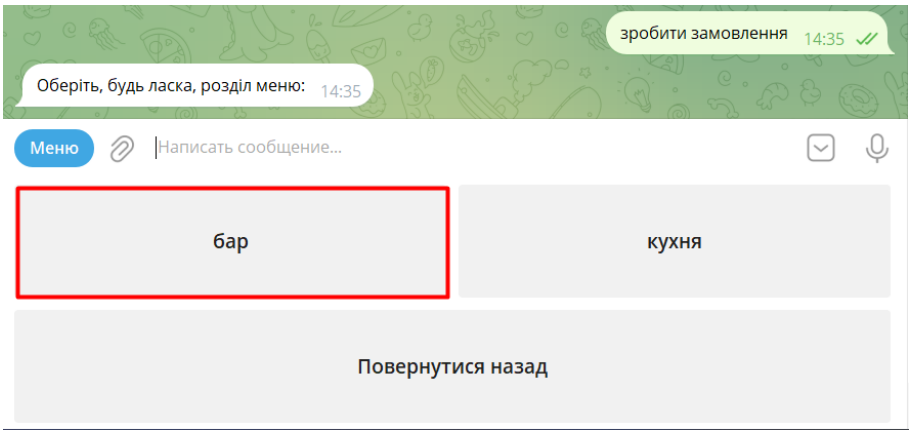

Оберіть, будь ласка, розділ меню: 14:25
піцца 14:25
Список піц: 14:25
Піцца Маргарита
Піцца Пепероні
Піцца Кватро Стагіоні
Піцца Гавайська
Піцца Вегетаріана
Піцца Капрічоза
Піцца Кватро Формаджі
Меню
Написать сообщение...
піцца
доповнення
десерти
Повернутися назад

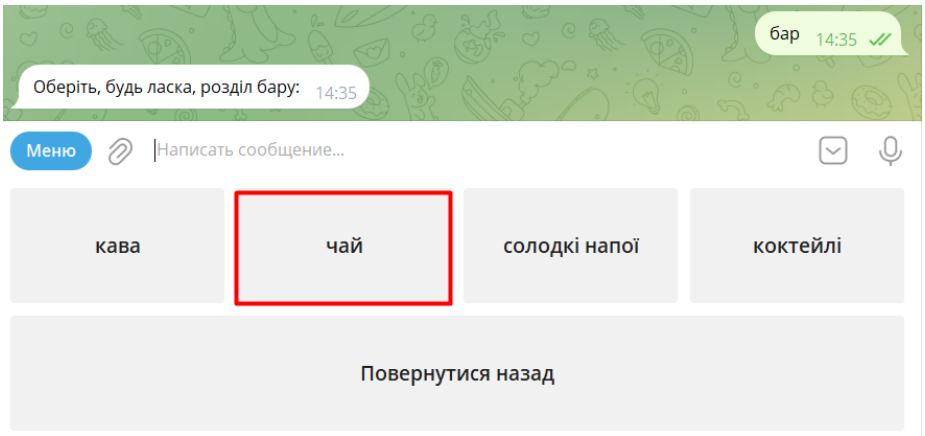

Назва: Піцца Капрічоза
Опис: Піца з грибами і артишоками
Склад: Гриби, артишоки, сир, соус
Вартість: 160.0 грн
14:30
Меню
Написать сообщение...
піцца
доповнення
десерти
Повернутися назад

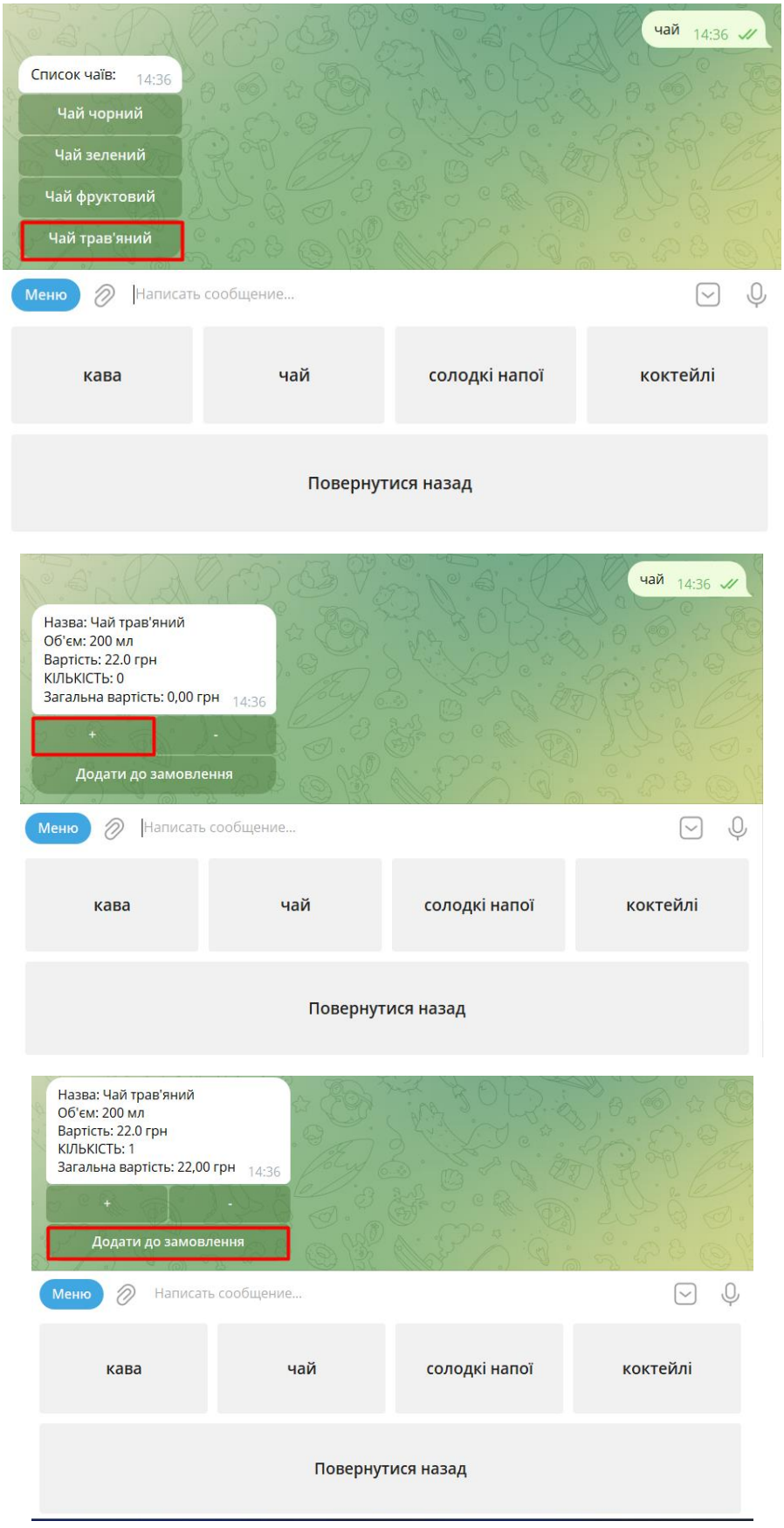

Повернутися назад 14:31
Оберіть, будь ласка, розділ меню: 14:31
Меню
Написать сообщение...
бар
кухня
Повернутися назад

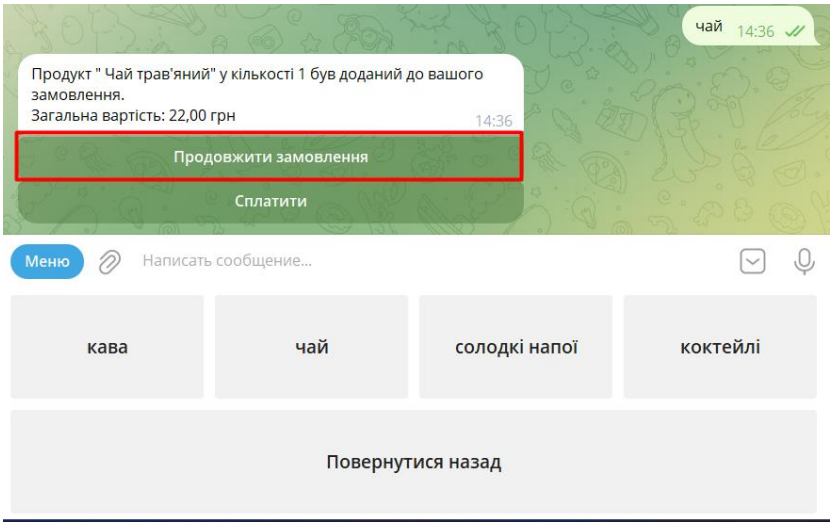

Повернутися назад 14:33
Оберіть, будь ласка, розділ меню: 14:33
Повернутися назад 14:33
Оберіть, будь ласка, опцію: 14:33
Меню
Написать сообщение...
переглянути меню
зробити замовлення
підписатись на розсилку новин

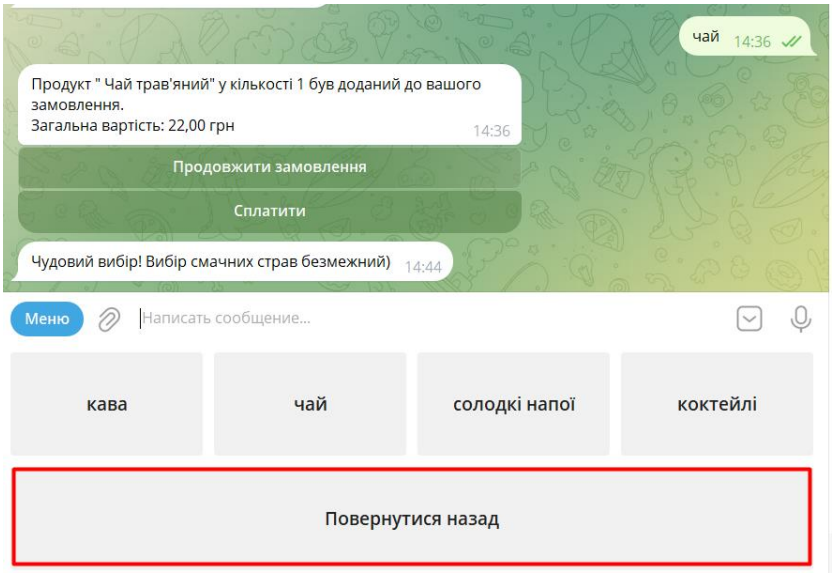

підписатись на розсилку новин 14:34
Ви підписалися на розсилку новин. 14:34
Оберіть, будь ласка, опцію: 14:34
Меню
Написать сообщение...
переглянути меню
зробити замовлення
відписатись від розсилки новин

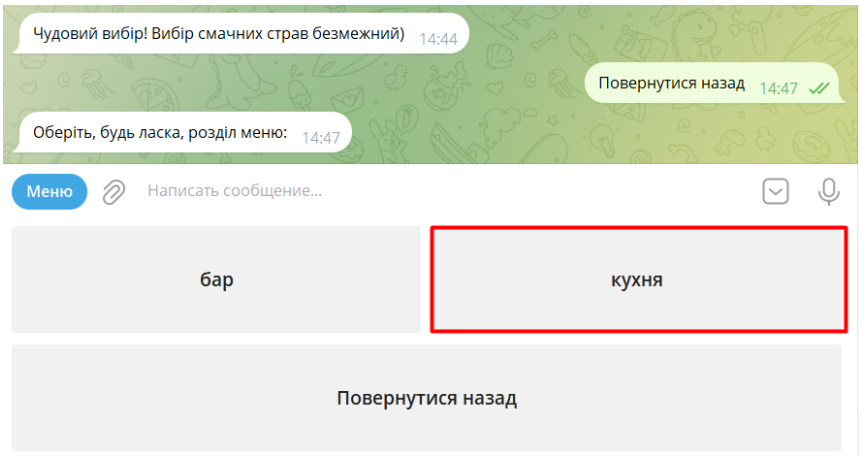

зробити замовлення 14:35
Оберіть, будь ласка, розділ меню: 14:35
Меню
Написать сообщение...
бар
кухня
Повернутися назад

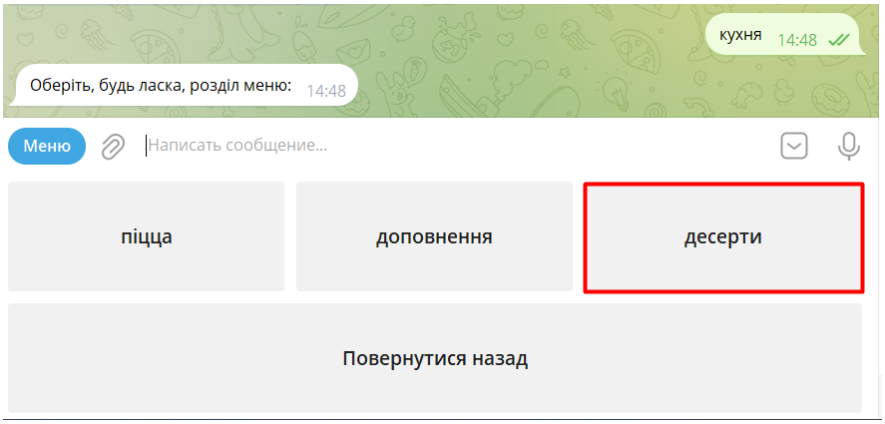

бар 14:35
Оберіть, будь ласка, розділ бару: 14:35
Меню
Написать сообщение...
кава
чай
солодкі напої
коктейлі
Повернутися назад

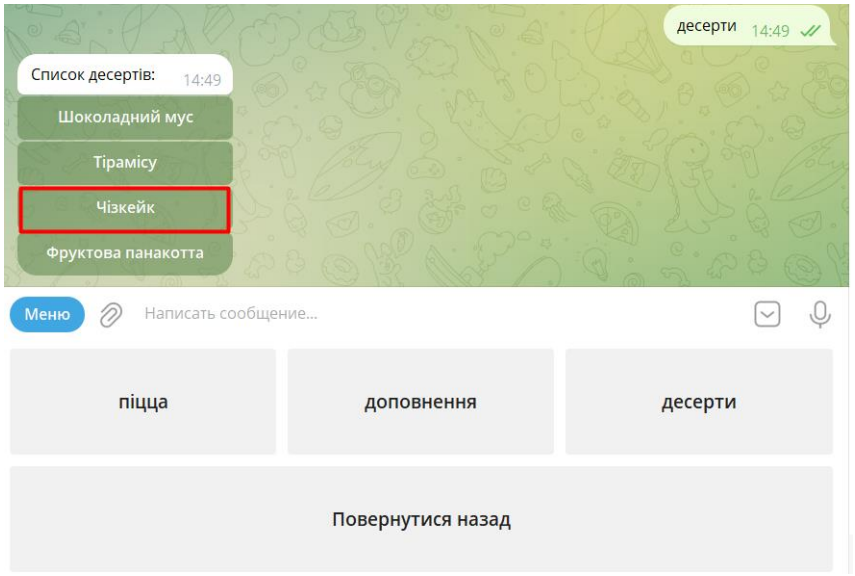

чай 14:36
Список чаїв: 14:36
Чай чорний
Чай зелений
Чай фруктовий
Чай трав'яний
Меню
Написать сообщение...
кава
чай
солодкі напої
коктейлі
Повернутися назад
чай 14:36
Назва: Чай трав'яний
Об'єм: 200 мл
Вартість: 22.0 грн
КІЛЬКІСТЬ: 0
Загальна вартість: 0,00 грн 14:36
+
-
Додати до замовлення
Меню
Написать сообщение...
кава
чай
солодкі напої
коктейлі
Повернутися назад
Назва: Чай трав'яний
Об'єм: 200 мл
Вартість: 22.0 грн
КІЛЬКІСТЬ: 1
Загальна вартість: 22,00 грн 14:36
+
-
Додати до замовлення
Меню
Написать сообщение...
кава
чай
солодкі напої
коктейлі
Повернутися назад

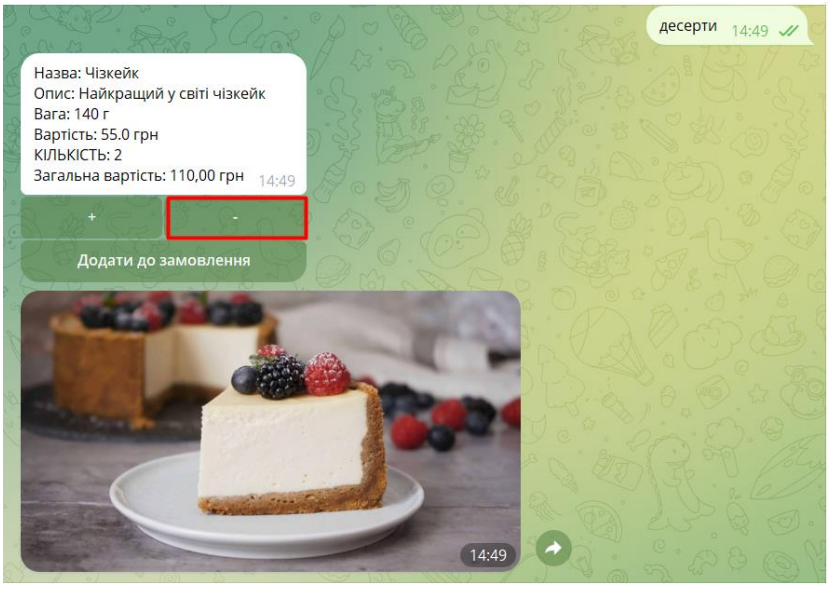

чай 14:36
Продукт " Чай трав'яний" у кількості 1 був доданий до вашого замовлення.
Загальна вартість: 22,00 грн
14:36
Продовжити замовлення
Сплатити
Меню
Написать сообщение...
кава
чай
солодкі напої
коктейлі
Повернутися назад


чай 14:36
Продукт " Чай трав'яний" у кількості 1 був доданий до вашого замовлення.
Загальна вартість: 22,00 грн
14:36
Продовжити замовлення
Сплатити
Чудовий вибір! Вибір смачних страв безмежний) 14:44
Меню
Написать сообщение...
кава
чай
солодкі напої
коктейлі
Повернутися назад

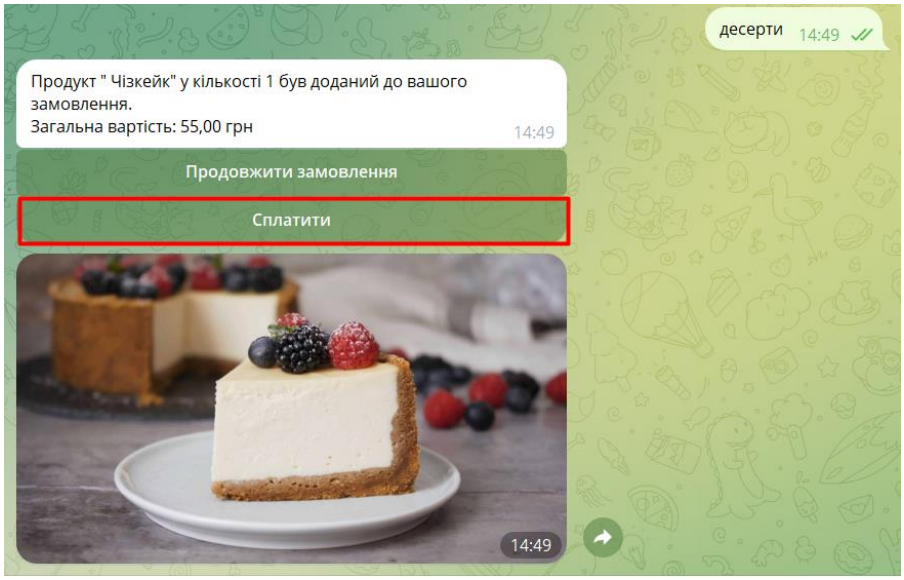

Чудовий вибір! Вибір смачних страв безмежний) 14:44
Повернутися назад 14:47
Оберіть, будь ласка, розділ меню: 14:47
Меню
Написать сообщение...
бар
кухня
Повернутися назад

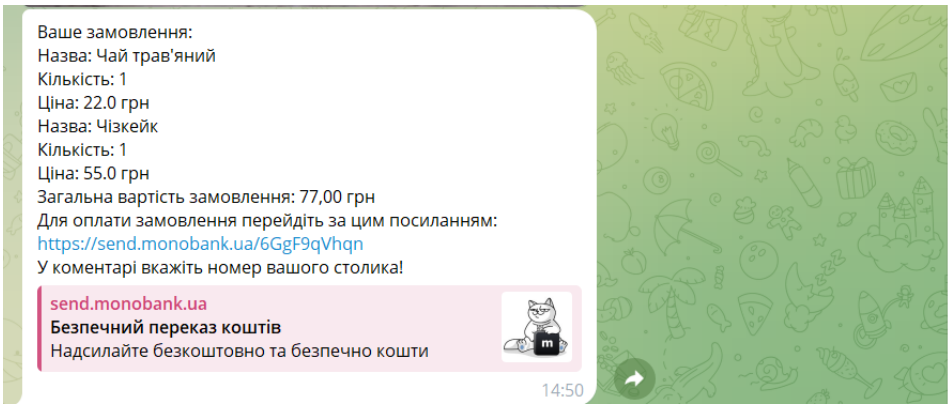

кухня 14:48
Оберіть, будь ласка, розділ меню: 14:48
Меню
Написать сообщение...
піцца
доповнення
десерти
Повернутися назад

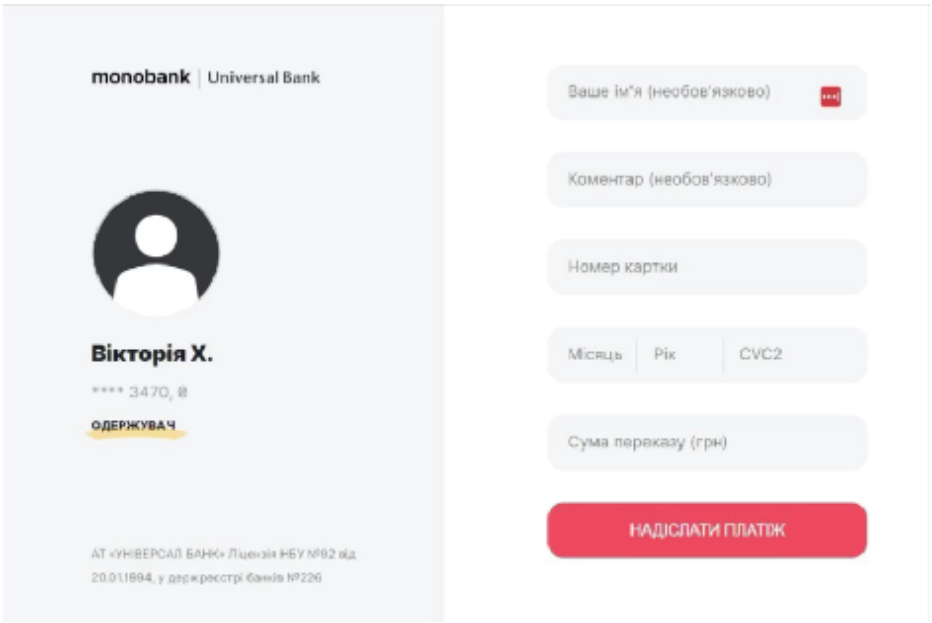

десерти 14:49
Список десертів: 14:49
Шоколадний мус
Тірамісу
Чізкейк
Фруктова панакотта
Меню
Написать сообщение...
піцца
доповнення
десерти
Повернутися назад


десерти 14:49
Назва: Чізкейк
Опис: Найкращий у світі чізкейк
Вага: 140 г
Вартість: 55.0 грн
КІЛЬКІСТЬ: 2
Загальна вартість: 110,00 грн 14:49
+
-
Додати до замовлення
14:49


Назва: Чізкейк
Опис: Найкращий у світі чізкейк
Вага: 140 г
Вартість: 55.0 грн
КІЛЬКІСТЬ: 1
Загальна вартість: 55,00 грн 14:49
+
-
Додати до замовлення


десерти 14:49
Продукт " Чізкейк" у кількості 1 був доданий до вашого замовлення.
Загальна вартість: 55,00 грн
14:49
Продовжити замовлення
Сплатити
14:49


Ваше замовлення:
Назва: Чай трав'яний
Кількість: 1
Ціна: 22.0 грн
Назва: Чізкейк
Кількість: 1
Ціна: 55.0 грн
Загальна вартість замовлення: 77,00 грн
Для оплати замовлення перейдіть за цим посиланням:
https://send.monobank.ua/6GgF9qVhqn
У коментарі вкажіть номер вашого столика!
send.monobank.ua
Безпечний переказ коштів
Надсилайте безкоштовно та безпечно кошти
14:50


monobank | Universal Bank
Вікторія Х.
**** 3470, ₴
ОДЕРЖУВАЧ
АТ «УНІВЕРСАЛ БАНК» Ліцензія НБУ №92 від 20.01.1994, у держреєстрі банків №226
Ваше ім'я (необов'язково)
Коментар (необов'язково)
Номер картки
Місяць
Рік
CVC2
Сума переказу (грн)
НАДІСЛАТИ ПЛАТІЖ

ДОДАТОК В

(Довідковий)

ПУБЛІКАЦІЯ ЗА ТЕМОЮ РОБОТИ


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ
Матеріали XXVIII Міжнародного молодіжного форуму
«Радіоелектроніка та молодь у XXI столітті»
ТОМ 3
«Інформаційні радіотехнології та технічний захист інформації»
Харків 2024

ДОДАТОК Г

(Обов'язковий)

СЛАЙДИ ПРЕЗЕНТАЦІЇ

Харківський національний університет радіоелектроніки
Кафедра Радіотехнологій інформаційно-комунікаційних систем

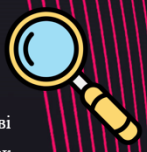
Кваліфікаційна робота бакалавра на тему:

СИСТЕМА ОБРОБКИ ЗАМОВЛЕНЬ В ЗАКЛАДАХ ХАРЧУВАННЯ

Студентка: Хаустова Вікторія Сергіївна
Група: ІТІР-20-1
Керівник : Ганшин Дмитро Геннадійович

Харків 2024

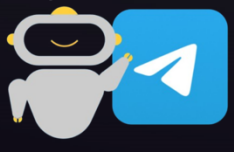
ВСТУП



У сучасному світі роль технологій в галузі харчування постійно зростає, надаючи закладам нові можливості для покращення обслуговування та залучення клієнтів. Швидкий технологічний розвиток призвів до того, що клієнти стають все вимогливішими до якості послуг і зручності обслуговування. Відповідно, закладам харчування потрібно адаптуватися до цих змін, пропонуючи інноваційні рішення, що задовольняють потреби сучасного споживача.

Однією з актуальних тенденцій є використання месенджерів для взаємодії з клієнтами. Телеграм, як один з найпопулярніших месенджерів, надає унікальні можливості для створення автоматизованих рішень, таких як Телеграм боти. Ці боти можуть оптимізувати процеси обслуговування, забезпечуючи зручну та ефективну взаємодію з клієнтами, що може підвищити конкурентоспроможність закладу на ринку.

Метою даної дипломної роботи була розробка система на основі Телеграм боту для автоматизації обслуговування клієнтів закладу харчування.



ОГЛЯД ІСНУЮЧИХ РІШЕНЬ АВТОМАТИЗАЦІЙ В СФЕРІ ХАРЧУВАННЯ

Автоматизація в галузі харчування відіграє ключову роль у покращенні ефективності, збільшенні продуктивності та підвищенні якості обслуговування. Ось деякі аспекти ролі автоматизації в галузі харчування:

- покращення ефективності процесів;
- мінімізація помилок;
- підвищення задоволення клієнтів;
- адаптація до технологічних тенденцій.

Недоліки:

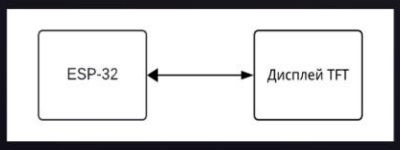
- витрати на придбання та налаштування;
- неможливість зробити замовлення заздалегідь.

УЗАГАЛЬНЕНА СТРУКТУРНА СХЕМА СИСТЕМИ ОБРОБКИ ЗАМОВЛЕНЬ

Для ефективної взаємодії між користувачами Telegram боту і співробітниками кухні використовується три основні компоненти: телефон з Telegram ботом, сервер для обробки замовлень і сенсорний дисплей. Користувачі надсилають замовлення через Telegram бот, сервер обмінюється інформацією з базою даних для зберігання та отримання даних про замовлення і надсилає їх на дисплей, розташований на кухні, де співробітники можуть оновлювати статус замовлень.

ВЗАЄМОДІЯ ЗІ СПІВРОБІТНИКАМИ КУХНІ

Підключення ESP-32 до дисплею



Код для ESP-32, який буде взаємодіяти з сервером і відображати дані на дисплеї з використанням HTTP запиту

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <ArduinoJson.h>
#include <Adafruit_GFX.h>
#include <Adafruit_ILI9341.h>

const char* ssid = "my_SSID"; // SSID Wi-Fi мережі
const char* password = "my_PASSWORD"; // Пароль до Wi-Fi мережі

#define TFT_CS    15
#define TFT_RST   4
#define TFT_DC    2

Adafruit_ILI9341 tft = Adafruit_ILI9341(TFT_CS, TFT_DC, TFT_RST);

void setup() {
  Serial.begin(115200);

  // Підключення до WiFi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");

  // Ініціалізація дисплея
  tft.begin();
  tft.setRotation(1); // Орієнтація дисплея
  tft.fillScreen(ILI9341_BLACK); // Заливка екрану чорним кольором
  tft.setTextColor(ILI9341_WHITE); // Колір тексту білий
  tft.setTextSize(2); // Розмір тексту
}
```

```
void loop() {
  if (WiFi.status() == WL_CONNECTED) {
    HTTPClient http;
    http.begin("http://server.com/api/orders"); //адреса сервера
    int httpResponseCode = http.GET();

    if (httpResponseCode > 0) {
      String payload = http.getString();
      Serial.println(payload); // Відображення отриманих даних у серійному моніторі

      // Парсинг JSON
      DynamicJsonDocument doc(2048); // Збільшено розмір для більшого JSON
      deserializeJson(doc, payload);
      JsonArray orders = doc["orders"];

      // Відображення даних на дисплеї
      tft.fillScreen(ILI9341_BLACK); // Очищення екрану
      tft.setCursor(0, 0); // Встановлення курсору у верхній лівий кут
      for (JsonObject order : orders) {
        tft.print("Order ID: ");
        tft.println(order["id"].as<String>());
        tft.print("Order: ");
        tft.println(order["order"].as<String>());
        tft.print("Status: ");
        tft.println(order["status"].as<String>());
        tft.println(); // Додати порожній рядок між замовленнями
      }
    } else {
      Serial.print("Error on HTTP request: ");
      Serial.println(httpResponseCode);
    }
    http.end();
  }
  delay(10000); // Оновлення за секунд перед наступним запитом
}
```

Приклад отриманих від сервера даних у форматі JSON

```
{
  "orders": [
    {"id": 1, "status": "Готується", "order": "Піцца Маргарита і чорний чай"},
    {"id": 2, "status": "Готово", "order": "Чізкейк і капучино"},
    {"id": 3, "status": "Готово", "order": "Лимонад полуниця/чізкейк і піцца Капрічоза"}
  ]
}
```


ІНТЕРФЕЙС ДЛЯ ВИВОДУ НА ДИСПЛЕЙ

Код файлу script.js

```
document.addEventListener('DOMContentLoaded', function() {
    fetchOrders();
    setInterval(fetchOrders, 5000);  // Оновлення кожні 5 секунд
});

function fetchOrders() {
    fetch('/api/orders')  // Запит до вашого API
        .then(response => response.json())
        .then(data => {
            const ordersContainer = document.getElementById('orders-container');
            ordersContainer.innerHTML = '';
            data.orders.forEach(order => {
                const orderElement = document.createElement('div');
                orderElement.className = 'order';
                orderElement.innerHTML = `
                    <p>Order ID: ${order.id}</p>
                    <p>Product: ${order.product}</p>
                    <p class="order-status">Status: ${order.status}</p>
                    <button onclick="updateOrderStatus(${order.id}, 'Готово')">Mark as Ready</button>
                `;
                ordersContainer.appendChild(orderElement);
            });
        });
}

function updateOrderStatus(orderId, status) {
    fetch(`/api/orders/${orderId}`, {
        method: 'PUT',
        headers: {
            'Content-Type': 'application/json',
        },
        body: JSON.stringify({ status }),
    })
    .then(response => response.json())
    .then(() => {
        fetchOrders();  // Оновити список замовлень після зміни статусу
    });
}
```

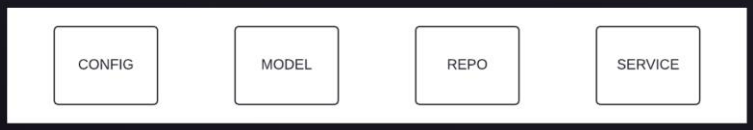
Код файлу index.html

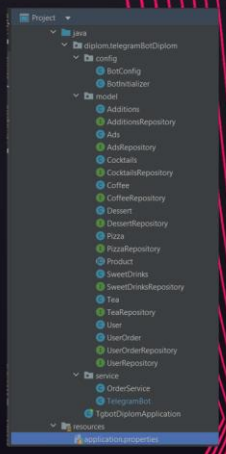
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Order Management</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <div id="orders-container">
        <!-- Інформація про замовлення буде динамічно додаватися сюди -->
    </div>
    <script src="script.js"></script>
</body>
</html>
```

Код файлу styles.css

```
body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
    background-color: #f4f4f4;
}
#orders-container {
    width: 80%;
    margin: 20px auto;
    padding: 10px;
    background-color: #fff;
    border: 1px solid #ccc;
    box-shadow: 2px 2px 12px rgba(0, 0, 0, 0.1);
}
.order {
    padding: 10px;
    border-bottom: 1px solid #ccc;
}
.order:last-child {
    border-bottom: none;
}
.order-status {
    font-weight: bold;
}
```

ОСНОВНІ КОМПОНЕНТИ ПРОЄКТУ:



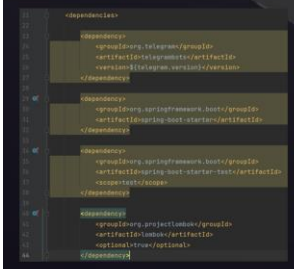


Пакет diplom.telegramBotDiplom — це основний пакет проєкту, який містить кілька підпакетів для різних аспектів реалізації:

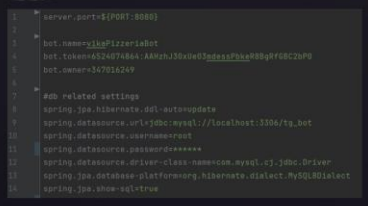
- пакет config — цей пакет містить конфігураційні класи для налаштування Телеграм бота, а саме: BotConfig (клас для налаштування конфігурації бота, наприклад, токену доступу та інших параметрів) і BotInitializer (клас для ініціалізації та запуску бота);
- пакет model містить класи, що представляють модель даних, і відповідні репозиторії для доступу до бази даних
- пакет service містить сервіси, що забезпечують основну бізнес-логіку додатка. А саме це: OrderService (сервіс для обробки замовлень) і TelegramBot (клас, що реалізує основну логіку взаємодії з Телеграм ботом).
- кореневий клас TgbotDiplomApplication — це головний клас, що запускає додаток, використовуючи Spring Boot.

7

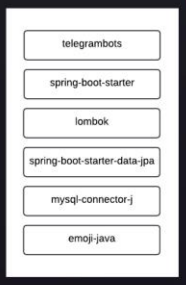
ОБОВ'ЯЗКОВІ ФАЙЛИ



Файл pom.xml, який використовується в системі керування збірками Apache Maven, забезпечує організацію залежностей і керування збіркою проєкту. Він визначає всі необхідні залежності, плагіни та властивості, автоматизуючи процеси компіляції, тестування і розгортання.



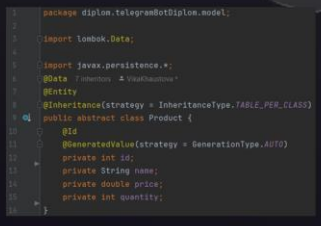
Файл application.properties використовується для налаштування конфігурації Spring Boot-додатка. Він містить різноманітні властивості, які визначають поведінку додатка, підключення до бази даних, параметри серверу та інші важливі налаштування.



Telegram Bots API: забезпечує інтеграцію з Telegram API
Spring Boot Starter: основний стартер для додатків на базі Spring Boot
Lombok: інструмент для автоматичної генерації стандартного коду
Spring Boot Starter Data JPA: забезпечує інтеграцію з JPA для роботи з базами даних.
MySQL Connector/J: JDBC-драйвер для підключення до бази даних MySQL
emoji-java: бібліотека для роботи з емодзі в Java-додатках

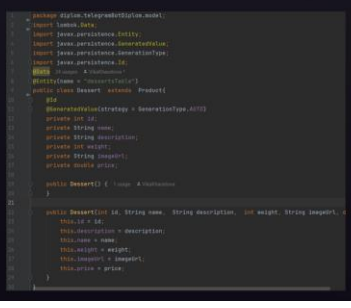
8

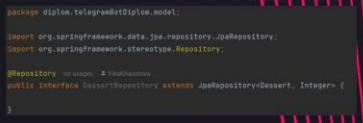
КЛАСИ-СУТНОСТІ СТРАВ І НАПОЇВ



Клас Product є абстрактним класом, який слугує базовою моделлю для різних продуктів. Він визначає загальні властивості, які є спільними для всіх продуктів, і використовується для створення конкретних підкласів (наприклад, Dessert, Tea, Coffee тощо), що представляють конкретні типи продуктів.

Клас Dessert є прикладом сутності, яка представляє десерти в БД. Аналогічно цьому класу створені всі інші класи, що представляють різні продукти, такі як піццу, доповнення, кавові напої, чаї, коктейлі, солодкі напої тощо.





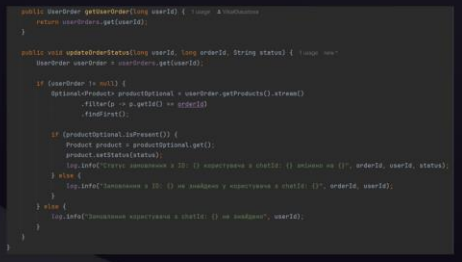
Клас DessertRepository і аналогічні йому репозиторії є інтерфейсами, що забезпечують доступ до БД для відповідних сутностей. Вони розширюють інтерфейс JpaRepository від Spring Data JPA, який надає набір стандартних CRUD-операцій (Create, Read, Update, Delete) для роботи з базою даних.

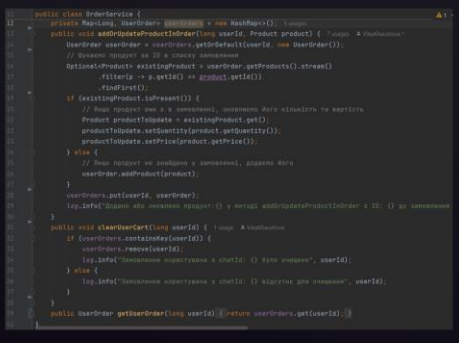
8

КЛАС, ЩО ВІДПОВІДАЄ ЗА ЗАМОВЛЕННЯ

Клас OrderService відіграє ключову роль у керуванні замовленнями користувачів у телеграм-боті. Його загальне призначення полягає у наданні функціоналу для додавання, оновлення та очищення замовлень користувачів.

Також у класі реалізовано можливість оновлення статусу замовлення і надсилання повідомлення про статус користувачу.

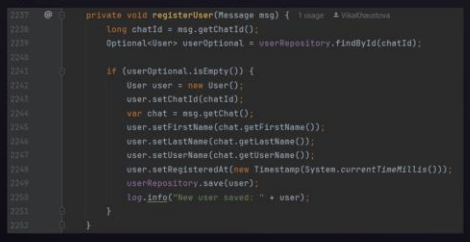




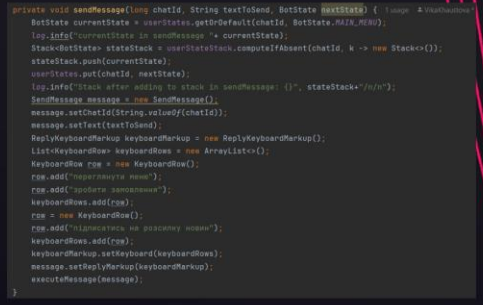
9

КЛАС «TELEGRAMBOT»

Метод registerUser() реєструє нового користувача, якщо він ще не збережений у базі даних. Використовується інформація з повідомлення для збереження даних користувача (ім'я, прізвище, ім'я користувача та час реєстрації).



Метод sendMessage() відповідає за надсилання повідомлення користувачу та оновлення стану бота. Він отримує поточний стан користувача, додає його до стека станів, після чого встановлює новий стан.



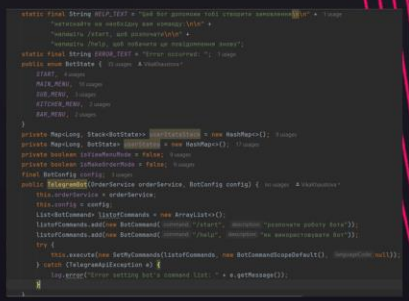
Метод startCommandReceived() надсилає вітальне повідомлення новому користувачу, очищає його кошик та видаляє інформацію про кількість замовлених товарів.

10

КЛАС «TELEGRAMBOT»

Метод executeMessage() безпосередньо надсилає повідомлення через бібліотеку Telegram API. Він приймає об'єкт SendMessage, що містить параметри повідомлення, і викликає метод execute для його відправлення. У випадку виникнення помилки метод фіксує її в логах.

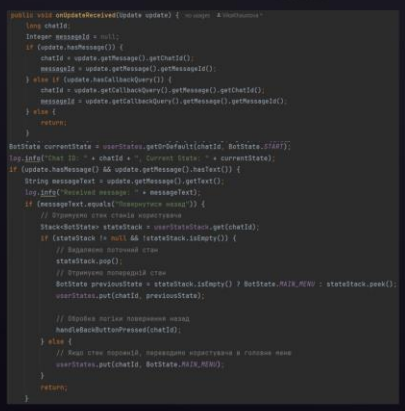
Метод prepareAndSendMessage() створює повідомлення для надсилання. Він ініціалізує об'єкт SendMessage, встановлює ідентифікатор чату та текст повідомлення, після чого викликає executeMessage для відправлення цього повідомлення користувачу.

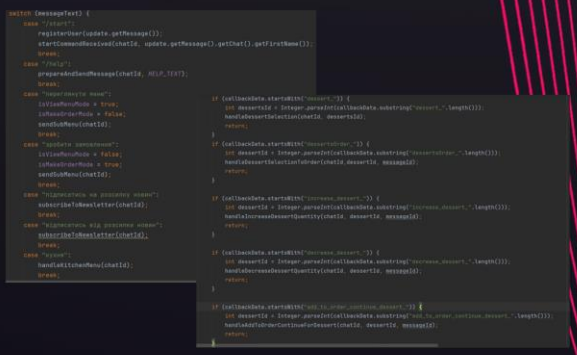


Частина коду, зображена на рисунку вище, налаштовує основні параметри та поведінку Telegram бота для створення замовлень.

11

КЛАС «TELEGRAMBOT»





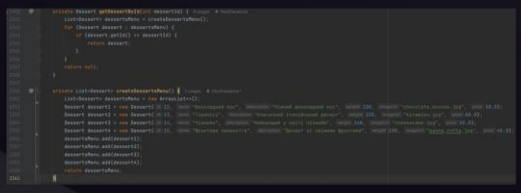
Метод onUpdateReceived() обробляє всі оновлення, які надходять до Telegram-бота. Метод обробляє різні типи повідомлень та запити зворотного виклику відповідно до поточного стану користувача.

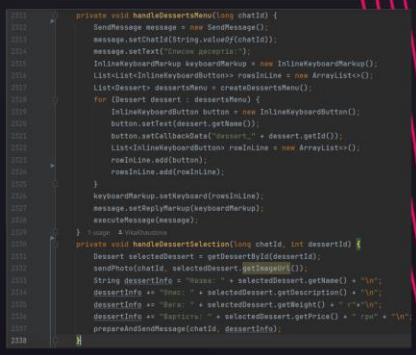
12

КЛАС «TELEGRAMBOT»

Здійснює пошук десерту у списку десертів за заданим ідентифікатором метод getDessertById(). Він повертає об'єкт десерту, якщо ідентифікатор знайдено, або null, якщо десерт з таким ідентифікатором не знайдено.

createDessertsMenu() - цей метод створює та повертає список десертів. Він ініціалізує кілька об'єктів десертів із відповідними властивостями, такими як назва, опис, вага, зображення та вартість, і додає їх до списку, який повертається.

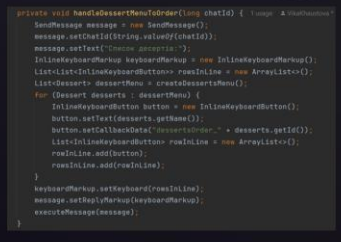




За відображення списку десертів користувачеві відповідає метод handleDessertsMenu(). Він створює повідомлення з текстом "Список десертів:" та налаштовує клавіатуру з кнопками для кожного десерту, що дозволяє користувачу вибрати конкретний десерт.
Метод handleDessertSelection() виконується після вибору десерту користувачем.

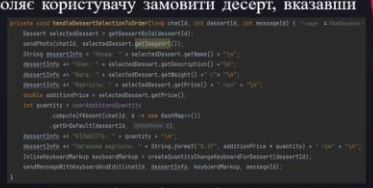
13

КЛАС «TELEGRAMBOT»

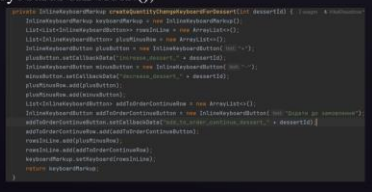


Метод handleDessertMenuToOrder() дуже схожий на handleDessertsMenu(). Обидва методи призначені для відображення меню десертів, але handleDessertsMenu() більше орієнтований на загальний перегляд десертів, тоді як handleDessertMenuToOrder() орієнтований на замовлення десертів з можливістю вибору кількості.

Аналогічна ситуація у методів handleDessertSelection() і handleDessertSelectionToOrder(). Обидва методи обробляють вибір десерту, але з різними цілями: перший показує інформацію про десерт, а другий дозволяє користувачу замовити десерт, вказавши його кількість.

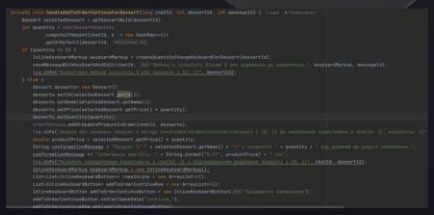


Для створення клавіатури для зміни кількості десерту а також додавати його до замовлення – використовується метод createQuantityChangeKeyboardForDessert(),



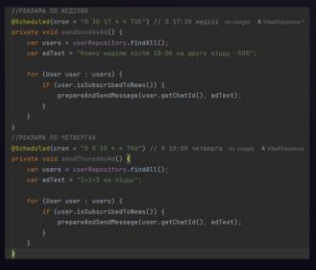
14

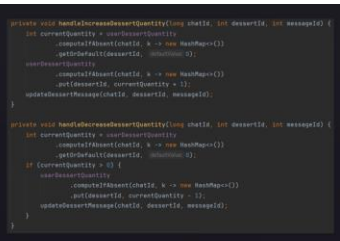
КЛАС «TELEGRAMBOT»



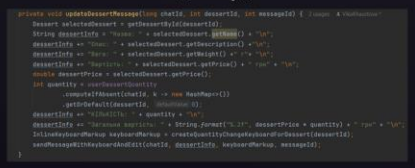
handleAddToOrderContinueForDessert() - цей метод обробляє підтвердження додавання десерту до замовлення.

Методи sendSundayAd() і sendThursdayAd() використовуються для автоматичної розсилки рекламних повідомлень користувачам, які підписані на новини, у визначений день та час.





Два методи, які збільшують та зменшують кількість обраного десерту для конкретного користувача та оновлюють відповідне повідомлення - handleIncreaseDessertQuantity() і handleIncreaseDessertQuantity().

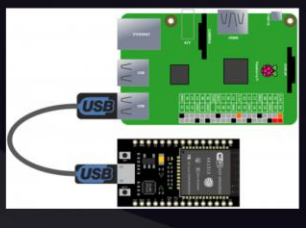


За оновлення повідомлення з інформацією про десерт та його кількість відповідає метод updateDessertMessage().

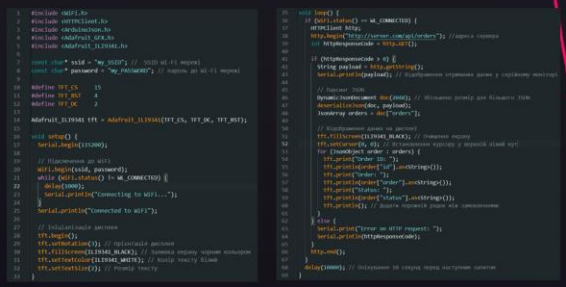
ВЗАЄМОДІЯ ЗІ СПІВРОБІТНИКАМИ КУХНІ

Структурна схема системи

Підключення ESP-32 до Raspberry Pi



Код для ESP-32, який буде взаємодіяти з сервером і відображати дані на дисплеї з використанням HTTP запиту

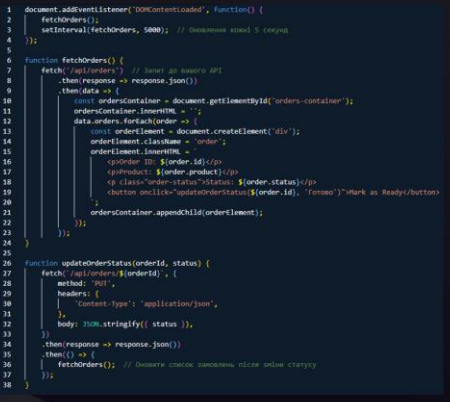


Приклад отриманих від сервера даних у форматі JSON

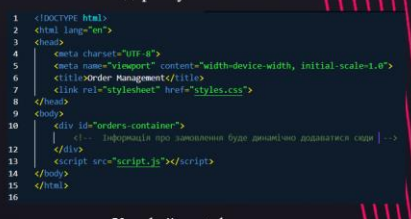
```
{
  "orders": [
    {"id": 1, "status": "Готується", "order": "Піцца Маргарита і чорний чай"},
    {"id": 2, "status": "Готово", "order": "Чізкейк і капучино"},
    {"id": 3, "status": "Готово", "order": "Лимонад полуниця/чизкейк і піцца Капрічоза"}
  ]
}
```


ВЕБ ДОДАТОК ДЛЯ RASPBERRY PI

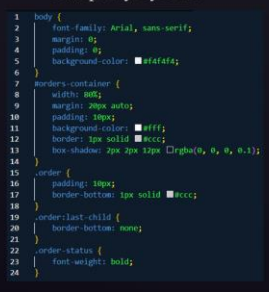
Код файлу script.js



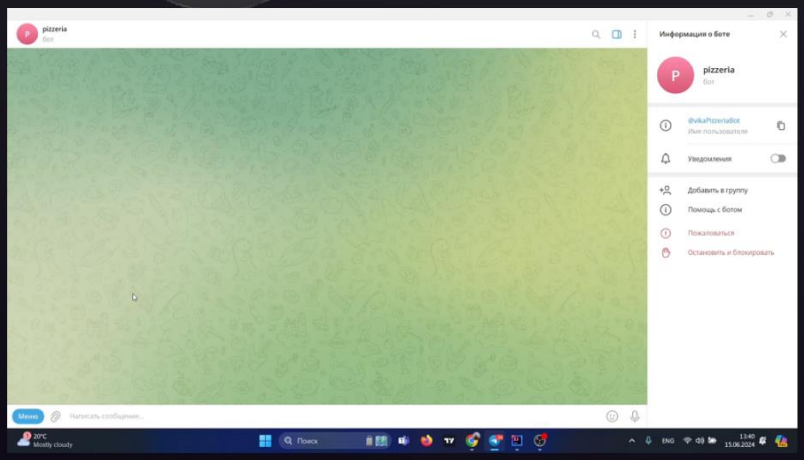
Код файлу index.html



Код файлу styles.css



ДЕМОНСТРАЦІЯ РОБОТИ



18

ВИСНОВКИ

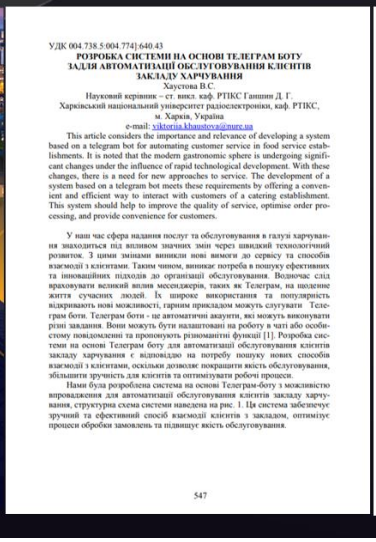
Система включає в себе кілька ключових компонентів, що працюють у тісній взаємодії. Telegram бот виступає основним інтерфейсом для клієнтів, надаючи їм можливість переглядати меню, робити замовлення та отримувати інформацію про статус своїх замовлень.

Результати дають зрозуміти, що автоматизована система обслуговування клієнтів є ефективним рішенням для закладів харчування. Вона дозволяє зменшити час обробки замовлень, покращити точність даних, підвищити рівень задоволеності клієнтів і забезпечити зручність для персоналу. Інтеграція мікроконтролера ESP-32 з системою додатково спрощує процеси на кухні, що позитивно впливає на загальну продуктивність.

На підставі отриманих результатів можна зробити висновок, що використання сучасних технологій, таких як Telegram бот та мікроконтролери, є доцільним та ефективним для автоматизації процесів у сфері обслуговування. Подальший розвиток даної системи може включати впровадження нових функцій, таких як автоматизоване управління запасами, інтеграція з платіжними системами та розширення можливостей аналітики.

19

ПУБЛІКАЦІЯ ЗА ТЕМОЮ РОБОТИ



УДК 004.738.5:004.774]:640.43

РОЗРОБКА СИСТЕМИ НА ОСНОВІ ТЕЛЕГРАМ БОТУ ЗАДЛЯ АВТОМАТИЗАЦІЇ ОБСЛУГОВУВАННЯ КЛІЄНТІВ ЗАКЛАДУ ХАРЧУВАННЯ

Хаустова В.С.
Науковий керівник – ст. викл. каф. РТКС Ганшин Д. Г.
Харківський національний університет радіоелектроніки, каф. РТКС,
м. Харків, Україна
e-mail: viktoriia.khaustova@nure.ua

This article considers the importance and relevance of developing a system based on a telegram bot for automating customer service in food service establishments. It is noted that the modern gastronomic sphere is undergoing significant changes under the influence of rapid technological development. With these changes, there is a need for new approaches to service. The development of a system based on a telegram bot meets these requirements by offering a convenient and efficient way to interact with customers of a catering establishment. This system should help to improve the quality of service, optimise order processing, and provide convenience for customers.

У наш час сфера надання послуг та обслуговування в галузі харчування знаходиться під впливом значних змін через швидкий технологічний розвиток. З цими змінами виникли нові вимоги до сервісу та способів взаємодії з клієнтами. Таким чином, виникає потреба в пошуку ефективних та інноваційних підходів до організації обслуговування. Водночас слід враховувати великий вплив месенджерів, таких як Телеграм, на щоденне життя сучасних людей. Їх широке використання та популярність відкривають нові можливості, гарним прикладом можуть слугувати Телеграм боти. Телеграм боти – це автоматичні акаунти, які можуть виконувати різні завдання. Вони можуть бути налаштовані на роботу в чаті або особистому повідомленні та пропонують різноманітні функції [1]. Розробка системи на основі Телеграм боту для автоматизації обслуговування клієнтів закладу харчування є відповідно на потребу пошуку нових способів взаємодії з клієнтами, оскільки дозволяє покращити якість обслуговування, збільшити зручність для клієнтів та оптимізувати робочі процеси.

Нами була розроблена система на основі Телеграм-боту з можливістю впровадження для автоматизації обслуговування клієнтів закладу харчування, структурна схема системи наведена на рис. 1. Ця система забезпечує зручний та ефективний спосіб взаємодії клієнтів з закладом, оптимізує процеси обробки замовлень та підвищує якість обслуговування.

547

Рисунок 1 – Структурна схема системи

Для автоматизації обробки замовлень у закладі харчування через Телеграм бота, першим кроком є розробка та налаштування бота для клієнтів, що дозволить їм робити замовлення. Телеграм бот розроблено на основі мови програмування Java. Java – це об'єктно-орієнтована мова програмування загального призначення з простим і зрозумілим синтаксисом, яка підходить для різних платформ [2]. В проекті використовується низка фреймворків та бібліотек. Spring Boot використовується для створення серверної частини, яка обробляє запити від клієнтів і взаємодіє з базою даних. Telegram Bot API дозволяє створювати та взаємодіяти з Телеграм ботами. Lombok спрощує розробку за допомогою автоматичної генерації коду. Emoji Java використовується для роботи з емоджі в тексті. MySQL Connector/J забезпечує підключення до бази даних MySQL. JPA (Java Persistence API) використовується для взаємодії з об'єктами бази даних у Java-додатках. Maven є інструментом для автоматизації управління залежностями та збірки проекту.

Другим кроком є створення серверу, який приймає та обробляє дані від бота, зберігаючи їх у базі даних. Сервер - це апаратне забезпечення, виділене і спеціалізоване для виконання на ньому сервісного програмного забезпечення і зберігання цієї інформації [3]. Крім того, використано мікроконтролер ESP-32 для взаємодії з дисплеєм Raspberry Pi. ESP32 — це серія мікроконтролерів типу «система на кристалі», що мають інтегровані контролери Wi-Fi і Bluetooth, низьке енергоспоживання і невисоку ціну [4]. Така система дозволяє співробітникам кухні легко взаємодіяти з процесом готування та оновлювати статуси замовлень, що автоматично відображаються клієнтам через Телеграм бот.

548

ДОДАТОК Д

(Обов'язковий)

ВІДОМІСТЬ КВАЛІФІКАЦІЙНОЇ РОБОТ

№ п/п	Позначення	Найменування	Додаткові відомості
		Текстові документи	
1		Пояснювальна записка	100 с.

Зм	Арк.	№ докум.	Підп.	Дата	ГЮІК.ХХХХХХ.000Д2		
Розроб.		Хаустова			Система обробки замовлень в закладах харчування. Відомість кваліфікаційної роботи.	Літера	Аркуш 1 / Аркушів 1
Перевір.		Ганшин				У	
Н. контр.		Бітченко				ХНУРЕ Кафедра РТІКС	
Затв.		Зарудний					