Московский государственный технический
Университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»
Отчет по лабораторной работе №4
«Разработка бота на основе конечного автомата для Telegram с использованием языка Python»

Выполнил:
Власов Александр,
ИУ5-33Б

Проверил:
Гапанюк Е.Ю.

2023 г.

Задание

Разработайте бота для Telegram. Бот должен реализовывать конечный автомат из нескольких состояний.

Текст программы

```
from aiogram import Bot, Dispatcher, executor, types
from aiogram.contrib.fsm_storage.memory import MemoryStorage
from aiogram.dispatcher import FSMContext
from aiogram.dispatcher.filters.state import StatesGroup, State
from aiogram.types import ReplyKeyboardMarkup, KeyboardButton



class ClientState(StatesGroup):
    START_ORDER = State()
    CITY_SELECTED = State()
    RESTAURANT_SELECTED = State()
    DISH_SELECTED = State()
    DRINK_SELECTED = State()
    PROCCESS_ORDER = State()



bot = Bot(token='6087539215:AAHJo7sGhkCmzfMPWYiXRd7wkuZ0JV1xF9g')

storage = MemoryStorage()
dp = Dispatcher(bot, storage=storage)



@dp.message_handler(commands=['Go'])
async def start_proccess(message: types.Message, state: FSMContext) -> None:
    msg = '''Привет! 👋🤖 Я бот доставки еды! В каком ты городе?'''
```

```
    msk_btn = KeyboardButton('Москва')
    spb_btn = KeyboardButton('Санкт-Петербург')
    voronezh_btn = KeyboardButton('Воронеж')
    lipetsk_btn = KeyboardButton('Липецк')

    markup = ReplyKeyboardMarkup(resize_keyboard=True)
    markup.row(msk_btn, spb_btn)
    markup.row(voronezh_btn, lipetsk_btn)

    await message.answer(msg, reply_markup=markup)
    await state.set_state(ClientState.START_ORDER)


@dp.message_handler(state=ClientState.START_ORDER)
async def choose_restoraunts_process(message: types.Message,
                                     state: FSMContext):
    user_msg = message.text
    await state.update_data(CITY=user_msg)

    dragon_rest_btn = KeyboardButton('Китайский дракон')
    pylounge_rest_btn = KeyboardButton('PyLounge')

    markup = ReplyKeyboardMarkup(resize_keyboard=True)
    markup.row(dragon_rest_btn, pylounge_rest_btn)

    await message.answer('Выберите заведение', reply_markup=markup)
    await state.set_state(ClientState.CITY_SELECTED)


@dp.message_handler(state=ClientState.CITY_SELECTED)
async def dish_process(message: types.Message, state: FSMContext):
    user_msg = message.text
```

```
    await state.update_data(RESTAURANT=user_msg)

    soup_menu_btn = KeyboardButton('Суп')
    nosoup_menu_btn = KeyboardButton('Не суп')

    markup = ReplyKeyboardMarkup(resize_keyboard=True)
    markup.row(soup_menu_btn, nosoup_menu_btn)

    await message.answer('Выберите блюдо', reply_markup=markup)
    await state.set_state(ClientState.RESTAURANT_SELECTED)


@dp.message_handler(state=ClientState.RESTAURANT_SELECTED)
async def drink_process(message: types.Message, state: FSMContext):
    user_msg = message.text
    await state.update_data(DISH=user_msg)

    cola_menu_btn = KeyboardButton('Кола')
    more_cool_cola_menu_btn = KeyboardButton('РашнКола')

    markup = ReplyKeyboardMarkup(resize_keyboard=True)
    markup.row(cola_menu_btn, more_cool_cola_menu_btn)

    await message.answer('Выберите напиток', reply_markup=markup)
    await state.set_state(ClientState.DISH_SELECTED)


@dp.message_handler(state=ClientState.DISH_SELECTED)
async def order_process(message: types.Message, state: FSMContext):
    user_msg = message.text
    await state.update_data(DRINK=user_msg)
```

```
    proccess_btn = KeyboardButton('Оформить заказ')
    cancel_btn = KeyboardButton('Отмена')

    markup = ReplyKeyboardMarkup(resize_keyboard=True)
    markup.row(proccess_btn, cancel_btn)

    await message.answer('Мы почти закончили', reply_markup=markup)
    await state.set_state(ClientState.DRINK_SELECTED)


@dp.message_handler(state=ClientState.DRINK_SELECTED)
async def finish_process(message: types.Message, state: FSMContext):
    user_msg = message.text
    if user_msg == 'Оформить заказ':
        user_state_data = await state.get_data()
        city = user_state_data['CITY']
        rest = user_state_data['RESTAURANT']
        dish = user_state_data['DISH']
        drink = user_state_data['DRINK']
        msg = f'''Ваш заказ: {dish} {drink} из {rest} ({city})
ОФОРМЛЕН!!!'''
        await message.answer(msg)
    else:
        await message.answer('Пока(')
    await state.finish()


if __name__ == '__main__':
    executor.start_polling(dp, skip_updates=True)
```

Пример выполнения программы