

**Московский государственный технический  
университет им. Н.Э. Баумана.**

Кафедра «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»  
Отчет по лабораторной работе №6.2  
На тему «Разработка бота на основе конечного автомата для Telegram с  
использованием языка Python»

Выполнил:

студент группы ИУ5-31Б  
Мишакин А.О.

Подпись и дата:

Проверил:

преподаватель каф. ИУ5  
Гапанюк Ю.Е.

Подпись и дата:

Москва, 2024

**Цель работы:** изучение разработки ботов в Telegram.

**Задание:** Разработайте бота для Telegram.

**Описание:** здесь написана вторая часть кода ТГ бота из 5ого отчёта.

**Текст программы:**

```
def send_notification(chat_id, topic):
    bot.send_message(chat_id, f"Напоминание: {topic}")

def show_tasks(message):
    chat_id = message.chat.id
    if chat_id not in user_tasks or not user_tasks[chat_id]:
        bot.send_message(chat_id, "У вас нет задач.")
    else:
        tasks = "\n".join([f"{i+1}. {topic} - {time}" for i, (topic, time) in
            enumerate(user_tasks[chat_id])])
        bot.send_message(chat_id, f"Ваши задачи:\n{tasks}")

def start_quiz(message):
    question = "Кто был первым президентом США?"
    options = ["Авраам Линкольн", "Джордж Вашингтон", "Томас Джефферсон"]

    markup = telebot.types.ReplyKeyboardMarkup(resize_keyboard=True)
    for option in options:
        markup.add(KeyboardButton(option))

    bot.send_message(message.chat.id, question, reply_markup=markup)
    bot.register_next_step_handler(message, handle_quiz_response)

def handle_quiz_response(message):
    correct_answer = "Джордж Вашингтон"
    if message.text == correct_answer:
        bot.send_message(message.chat.id, "Правильно!",
            reply_markup=start_keyboard())
    else:
        bot.send_message(message.chat.id, "Неправильно, попробуйте ещё раз.",
            reply_markup=start_keyboard())

def start_keyboard():
    markup = ReplyKeyboardMarkup(resize_keyboard=True)
    markup.add(KeyboardButton("1. Добавить задачу"), KeyboardButton("2.
Показать задачи"))
    markup.add(KeyboardButton("3. Начать квиз"), KeyboardButton("4.
Справка"))
    return markup

def run_scheduler():
    while True:
        schedule.run_pending()
        time.sleep(1)

thread = threading.Thread(target=run_scheduler, daemon=True)
thread.start()

def auxiliary_help(message):
    bot.send_message(message.chat.id,
```

```

"Привет! Я Ассистент по обучению. Сейчас я расскажу тебе, что я умею и что
вы сможете делать:\n"
"Вы можете:\n"
"1. Добавлять задачи😊\n"
"2. Смотреть расписание смотреть расписание своих задач!😊\n"
"3. Начать квиз???\n"
"4. Можете посмотреть эту справку) 📄 📄 📄\n"
"Что могу я👉👉👉\n"
"1. Я буду напоминать вам о ваших делах\n"
"2. Буду собирать список ваших дел. Я его никогда не забуду)\n"
"3. Для того, чтобы вы не забывали тему, я могу кидать вам КВИЗЫ\n"
"4. Чем могу помочь?", reply_markup=start_keyboard())

bot.polling(none_stop=True)

```

