

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение высшего образования «Уральский федеральный университет имени первого Президента России Б.Н. Ельцина» (УрФУ)

Институт фундаментального образования

Кафедра «Технической физики»

Оценка

Руководитель курсового

проектирования

Члены комиссии

Дата защиты

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

по теме: Создание телеграмм−бота для обучения сотрудников в сфере онлайн-продаж

по дисциплине: Информатика

Студент: Бузаньяров Эльдар Салаватович

(Подпись)

Группа: ФТ−140007

Екатеринбург

2025



Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение высшего образования «Уральский федеральный университет имени первого Президента России Б.Н. Ельцина» (УрФУ)

Институт фундаментального образования

Кафедра «Технической физики»

**Задание на курсовую работу**

Студент Бузаньяров Эльдар Салаватович

Группа ФТ−140007

специальность/направление подготовки 09.03.02 Информационные системы и технологии

1.Тема курсовой работы: Создание телеграмм−бота для обучения сотрудников в сфере онлайн–продаж.

2.Содержание работы, в том числе состав графических работ и расчетов: описание и реализация алгоритмов, блок−схемы алгоритмов, таблицы, листинг кода.

3.Дополнительные сведения: работа выполнена на языке программирования python и библиотеки pyTelegramBotAPI.

4. План выполнения курсовой работы

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование элементов  проектной работы | Сроки | Примечания | Отметка о выполнении |
| Изучение методов и библиотек | 6.01-8.01 |  |  |
| Знакомство со сферой онлайн−продаж, составление плана курса | 9.01 |  |  |
| Планирование | 10.01 |  |  |
| Реализация алгоритма | 10.01-13.01 |  |  |
| Предварительная проверка | 13.01 |  |  |
| Оформление пояснительной записки | 14.01-17.01 |  |  |
| Защита работы | 18.01 |  |  |

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/М.Н. Ятченко

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc187789308)

[1 Методология создания алгоритма 5](#_Toc187789309)

[1.1 Онлайн-продажи 5](#_Toc187789310)

[1.2 Создание бота 6](#_Toc187789311)

[1.2.1 Создание и обозначение блок−схем 6](#_Toc187789312)

[1.2.2 Приветствие 7](#_Toc187789313)

[1.2.3 Ввод имени 7](#_Toc187789314)

[1.2.4 Главное меню бота 8](#_Toc187789315)

[1.2.5 Аккаунт 9](#_Toc187789316)

[1.2.6 Выбор темы и урока 10](#_Toc187789317)

[1.2.7 Урок 10](#_Toc187789318)

[1.2.8 Прогресс 11](#_Toc187789319)

[1.3 Средства и методы создания системы 12](#_Toc187789320)

[1.3.1 Библиотеки и доп. средства 12](#_Toc187789321)

[1.3.2 Методы создания программным путем 12](#_Toc187789322)

[1.3.3 Реализация отдельных объектов 13](#_Toc187789323)

[1.3.4 Реализация приветственного сообщения 14](#_Toc187789324)

[1.3.5 Реализация создания аккаунта 14](#_Toc187789325)

[1.3.6 Реализация главного меню 15](#_Toc187789326)

[1.3.7 Реализация аккаунта 16](#_Toc187789327)

[1.3.8 Реализация навигации между темами и уроками 16](#_Toc187789328)

[1.3.9 Реализация уроков 17](#_Toc187789329)

[1.3.10 Реализация прогресса 18](#_Toc187789330)

[2 Реализация алгоритма 20](#_Toc187789331)

[2.1 Состав информационной системы 20](#_Toc187789332)

[2.2 Приветственное сообщение 20](#_Toc187789333)

[2.3 Создание аккаунта 20](#_Toc187789334)

[2.4 Главное меню 21](#_Toc187789335)

[2.5 Аккаунт 22](#_Toc187789336)

[2.6 Выбор темы и урока 22](#_Toc187789337)

[2.8 Урок 23](#_Toc187789338)

[2.9 Прогресс 24](#_Toc187789339)

[ЗАКЛЮЧЕНИЕ 25](#_Toc187789340)

[БИБЛИОГРАФИЧЕСКИЙ СПИСОК 26](#_Toc187789341)

[ПРИЛОЖЕНИЕ 27](#_Toc187789342)

# ВВЕДЕНИЕ

В условиях стремительного роста электронной коммерции и усиления конкуренции на рынке онлайн-продаж, обучение персонала приобретает первостепенное значение. Данный проект представляет собой реализацию телеграмм−бота, предоставляющего доступ к курсу по подготовке и повышению квалификации специалистов в этой области. Бот позволяет пользователям проходить обучение в интерактивном режиме, получая текстовую и мультимедийную информацию, а также взаимодействовать с системой посредством кнопок и обратных запросов. Также предусмотрен функционал для отслеживания личного прогресса, что мотивирует пользователей и позволяет им контролировать процесс обучения.

Проект способствует повышению профессиональной компетенции сотрудников, что, в свою очередь, ведет к увеличению эффективности продаж и улучшению качества обслуживания клиентов. Данная разработка может быть использована в любой организации, занимающейся онлайн-торговлей, и стать неотъемлемой частью процесса адаптации и развития персонала.

Целью данной работы является разработка телеграмм−бота, который предоставляет доступ к онлайн-курсу для обучения персонала в сфере онлайн-продаж.

Для достижения поставленной цели необходимо решить следующие задачи:

− Изучить сферу онлайн-продаж;

− Изучить инструменты и методы для создания телеграмм-ботов;

− Разработать телеграмм−бота;

− Реализовать телеграмм−бота;

− Протестировать телеграмм−бота.

# **1 Методология создания алгоритма**

## **1.1 Онлайн-продажи**

Онлайн-продажи представляют собой мощную и постоянно развивающуюся сферу, которая кардинально изменила принципы ведения бизнеса и потребительского поведения. За последние годы они превратились из нишевого явления в доминирующую силу в мировой экономике. Этот переход обусловлен рядом факторов, включая глобализацию, технологический прогресс и растущую популярность интернет-покупок. Сегодня для многих компаний онлайн-продажи являются не просто дополнительным каналом сбыта, а основным источником дохода и стратегическим направлением развития.

Преимущества онлайн-продаж очевидны: они позволяют бизнесу преодолевать географические ограничения и получать доступ к широкой, часто международной аудитории. Покупатели, в свою очередь, ценят удобство и доступность покупок в любое время и из любой точки мира. Развитие мобильных технологий еще больше усилило этот тренд, делая онлайн-шопинг еще более простым и доступным. Кроме того, электронная коммерция позволяет компаниям собирать и анализировать данные о поведении клиентов, что дает возможность персонализировать предложения и улучшать качество обслуживания[1].

Однако, стоит отметить, что динамичная природа онлайн-продаж также создает ряд вызовов. Конкуренция на этом рынке очень высока, и для успеха необходимо постоянное развитие и совершенствование. Успех в онлайн-продажах определяется не только качеством продукции и привлекательными ценами, но и профессионализмом персонала, который работает с клиентами в цифровой среде. Менеджеры по продажам должны обладать не только отличными навыками общения, но и глубоким пониманием потребностей клиентов, а также умением адаптироваться к постоянно меняющимся технологиям.

Работа в онлайн-продажах требует гибкости, креативности и постоянного обучения. Менеджеры должны уметь устанавливать контакт в цифровой среде, выявлять потребности и мотивации клиентов, представлять продукт таким образом, чтобы он был максимально привлекательным и решать возникающие вопросы и возражения. Постпродажное обслуживание также играет важную роль, поскольку от него зависит лояльность клиентов и их готовность к повторным покупкам. В конечном счете, онлайн-продажи — это не только про транзакции, но и про выстраивание долгосрочных отношений с клиентами, что требует постоянного совершенствования и развития навыков персонала.

## **1.2 Создание бота**

### **1.2.1 Создание и обозначение блок**−**схем**

В данной работе для описания отдельных процессов и алгоритмов взаимодействия с графическим интерфейсом будут использоваться блок−схемы, построенные с использованием приложения draw.io в соответствии с обозначениями, представленными в таблице 1.

Таблица 1 – Обозначение блок−схем

|  |  |
| --- | --- |
| Блок | Значение |
|  | Начало/конец алгоритма,  переход в следующий/предыдущий раздел |
|  | Процесс/Действие |
|  | Ячейка данных |
|  | Ввод/вывод данных |
|  | Показывают направление потока выполнения алгоритма |

### **1.2.2 Приветствие**

При запуске бота пользователь получает приветственное сообщение, содержащее имя его аккаунта в мессенджере. Так же появляется возможность нажать кнопку для перехода к следующему этапу. Схема взаимодействия представлена на рисунке 1.



Рисунок 1 – Приветствие

### **1.2.3 Ввод имени**

После приветствия бот задает вопрос о том, как зовут пользователя. После получения ответа бот записывает имя пользователя в отдельный словарь и предоставляет возможность продолжить работу, нажав кнопку. После нажатия пользователя переносит в главное меню телеграмм бота. Схема взаимодействия представлена на рисунке 2.

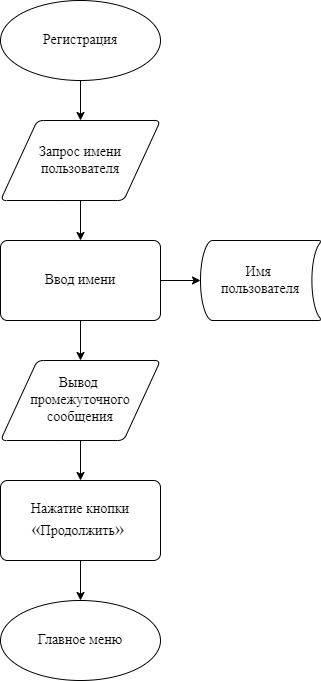


Рисунок 2 –Ввод имени

### **1.2.4 Главное меню бота**

В главном меню пользователю предоставляется возможность перейти в три раздела бота, а именно: аккаунт, выбор темы и прогресс. Переход в разделы осуществляется с помощью кнопок, с названиями соответствующих разделов. Схема взаимодействия представлена на рисунке 3.

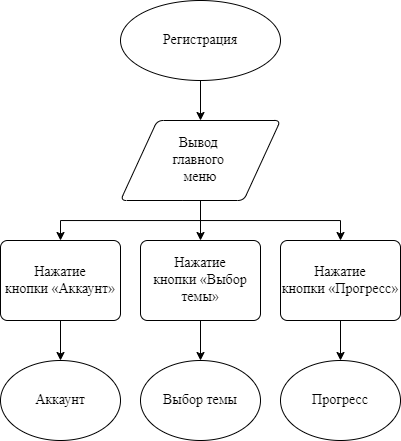


Рисунок 3 – Главное меню

### **1.2.5 Аккаунт**

В данном разделе пользователь получает данные об аккаунте, такие как имя пользователя и текущие курсы пользователя. Также пользователь может по нажатию кнопки вернуться в главное меню. Схема взаимодействия представлена на рисунке 4.

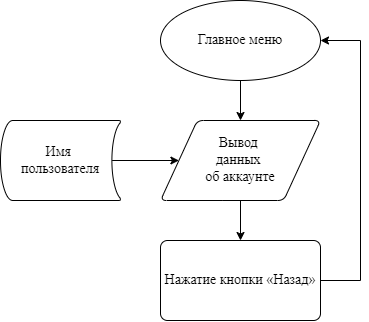


Рисунок 4 – Аккаунт

### **1.2.6 Выбор темы и урока**

Курс имеет 4 главные темы, каждая тема имеет 3 урока. В данном разделе пользователь имеет возможность перейти в любую тему, а из него в любой урок выбранной темы и получить соответствующую информацию. Также система кнопок даёт возможность возвращаться на уровень выше, позволяя пользователю выбрать другую тему или урок. Схема взаимодействия представлена на рисунке 5.

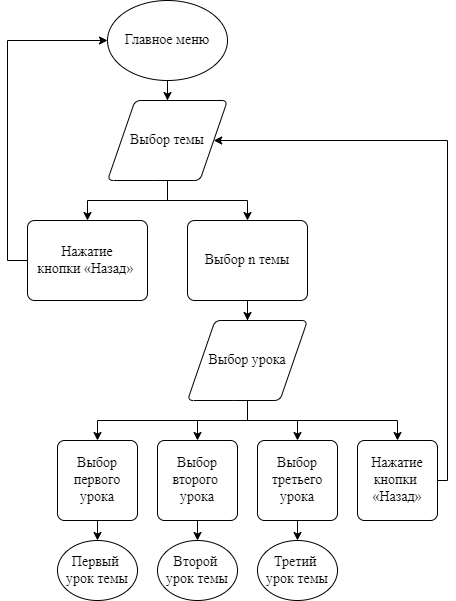


Рисунок 5 – Выбор темы

### **1.2.7 Урок**

После выбора определенного урока бот предоставляет пользователю информационные материалы по соответствующей теме, такие как:

– Текстовые материалы;

– Схемы;

– Изображения;

– Ссылки на статьи;

– Ссылки на видеоматериалы.

Так же можно отметить урок как пройденный, двигая прогресс прохождения курса или выйти и выбрать другой урок. Схема взаимодействия представлена на рисунке 6.

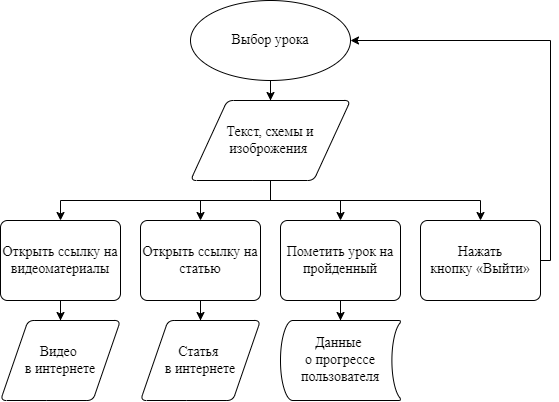


Рисунок 6 – Урок

### **1.2.8 Прогресс**

Данный раздел позволяет пользователю отслеживать свой прогресс прохождения курса и выйти обратно в меню. Здесь отображается следующая информация: количество пройденных уроков, общее количество уроков и их соотношение в процентах. Данные о прогрессе прохождения курса хранятся в отдельном словаре. Схема взаимодействия представлена на рисунке 7.

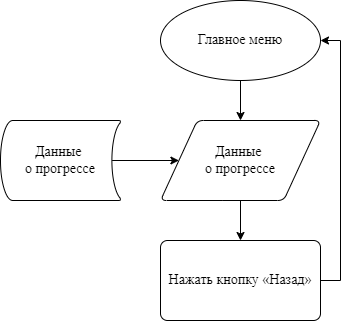


Рисунок 7 – Урок

## **1.3 Средства и методы создания системы**

### **1.3.1 Библиотеки и доп. средства**

Разработка телеграмм–бота для обучения сотрудников в сфере онлайн–продаж осуществляется с использованием языка программирования Python и библиотеки pyTelegramBotAPI.

pyTelegramBotAPI – библиотека Python для создания телеграмм–ботов. Она предоставляет возможность создания элементов интерфейса бота[1][2]. В данной работе с помощью данной библиотеки созданы:

− Текстовые сообщения;

− Сообщения с изображениями;

− Кнопки с ссылками.

Регистрация бота и привязка API-ключа осуществляется с помощью телеграмм-бота BotFather, позволяющего создавать ботов и управлять ими. Он так же предоставляет возможности для настройки имени и фотографии профиля бота.

### **1.3.2 Методы создания программным путем**

Телеграмм–бот для обучения сотрудников в сфере онлайн–продаж реализуется программным методом с использованием линейных и разветвляющихся алгоритмов. Полный код программы представлен в приложении А.

Для создания клавиатуры и кнопок, отправки и удаления сообщений и картинок используются соответствующие классы и методы. Для обработки запросов, отправляемых при нажатии кнопок применяются декораторы, они позволяет вызывать разные функции при разных событиях, происходящих в чате. Пример − отправка сообщений, нажатие кнопок.

### **1.3.3 Реализация отдельных объектов**

Для создания самого бота используется класс TeleBot. При создании объекта бота в класс передается API-ключ, полученный с помощью BotFather:

from telebot import \*  
token = '7398398254:AAFa27\_AjW45tXGjZbXSlMou8YH0cnsAx8I'  
bot = telebot.TeleBot(token)

Для реализации кнопок необходимо создать клавиатуру и объекты с классом кнопки. После создания кнопок их необходимо добавить в клавиатуру:

goback\_keyboard = types.InlineKeyboardMarkup()  
goback\_keyboard.row\_width = 2  
article = types.InlineKeyboardButton('Статья', url=article\_link)  
video = types.InlineKeyboardButton('Видео', url=video\_link)  
complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_1b')  
goback = types.InlineKeyboardButton('Выйти', callback\_data='theme1')  
goback\_keyboard.add(article, video, complete, goback)

При нажатии кнопок может быть осуществлен переход по интернет-ссылке или отправлен запрос, который будет обработан ботом с помощью декоратора callback\_query\_handler:

@bot.callback\_query\_handler(func = lambda call: True)  
def answer(call):  
 option = call.data  
 umessage = call.message  
 match option:  
 case 'start':  
 bot.send\_message(umessage.chat.id, text='Как вас зовут')  
 bot.register\_next\_step\_handler(umessage, registration)

Для отправки сообщений используются методы send\_message и send\_photo в зависимости от того, что необходимо отправить. Метод send\_message позволяет отправить текст и привязать к сообщению заранее созданную клавиатуру с кнопками. Метод send\_photo позволяет отправить изображение с подписью. При отправке сообщения, предыдущее удаляется, улучшая восприятие интерфейса. Удаление происходит по индексу сообщения, которое было отправлено перед текущим:

message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
bot.delete\_message(umessage.chat.id, message\_id - 1)

### **1.3.4 Реализация приветственного сообщения**

Бот обрабатывает команду start от пользователя и запускает функцию, которая создает текст сообщения, включающий имя аккаунта пользователя. Для продолжения создается клавиатура и кнопка с соответствующим текстом. Далее бот отправляет приветственный текст с клавиатурой, привязанной к сообщению. После нажатия кнопки бот получает запрос start :

@bot.message\_handler(commands=['start'])  
def welcome\_message(message):  
 welcome\_message = (f'Здравствуйте, {message.from\_user.first\_name}! \n'  
 f'Приветствуем вас на платформе для обучения сотрудников в сфере онлайн-продаж')  
 start\_button\_keyboard = types.InlineKeyboardMarkup()  
 start\_button = types.InlineKeyboardButton('Начать', callback\_data='start')  
 start\_button\_keyboard.add(start\_button)  
 bot.send\_message(message.chat.id, welcome\_message, reply\_markup=start\_button\_keyboard)

### **1.3.5 Реализация создания аккаунта**

Заранее создаются словари, хранящие данные об именах пользователях и их прогрессе:

user\_data = {}

progress\_data = {}

Так же создается функция, сохраняющая данные о введенном имени и устанавливающая нулевой прогресс у нового пользователя. При нажатии кнопки «Продолжить» бот получает запрос main:

def registration(message):  
 user\_id = message.chat.id  
 user\_name = message.text  
 user\_data[user\_id] = user\_name  
 progress\_data[user\_id] = []  
 continue\_keyboard = types.InlineKeyboardMarkup()  
 continue\_button = types.InlineKeyboardButton('Продолжить', callback\_data='main')  
 continue\_keyboard.add(continue\_button)  
 bot.send\_message(message.chat.id, f'Приятно познакомиться, {user\_name}!', reply\_markup=continue\_keyboard)

При получении запроса start бот просит ввести имя пользователя и вызывает ранее созданную функцию, обрабатывающую введенные данные:

@bot.callback\_query\_handler(func = lambda call: True)  
def answer(call):  
 option = call.data  
 umessage = call.message  
 match option:  
 case 'start':  
 bot.send\_message(umessage.chat.id, text='Как вас зовут')  
 bot.register\_next\_step\_handler(umessage, registration)

### **1.3.6 Реализация главного меню**

При получении запроса main бот отправляет сообщение с клавиатурой, которая содержит кнопки, позволяющие перейти в разделы бота. Каждая кнопка отправляет разные запросы, отвечающие за свои разделы:

case 'main':  
 choose\_topic = types.InlineKeyboardMarkup()  
 account\_b = types.InlineKeyboardButton('Аккаунт', callback\_data='account')  
 theme\_b = types.InlineKeyboardButton('Выбор темы', callback\_data='course')  
 progress\_b = types.InlineKeyboardButton('Прогресс', callback\_data='progress')  
 choose\_topic.row\_width = 1  
 choose\_topic.add(account\_b, theme\_b, progress\_b)  
 message\_id = bot.send\_message(umessage.chat.id, 'Курс "Эффективные продажи в онлайн-магазине"', reply\_markup=choose\_topic).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)

### **1.3.7 Реализация аккаунта**

При получении запроса account бот отправляет сообщение с данными об аккаунте, в частности имя пользователя, которое было получено при регистрации, и кнопку, позволяющую вернуться в главное меню:

case 'account':  
 chatid = umessage.chat.id  
 username = user\_data[chatid]  
 account\_message = (f'{username} \n'  
 f'\n'  
 f'Курсы:\n'  
 f' "Эффективные продажи в онлайн-магазине"')  
 account\_keyboard = types.InlineKeyboardMarkup()  
 account\_button = types.InlineKeyboardButton('Назад', callback\_data='main')  
 account\_keyboard.add(account\_button)  
 message\_id = bot.send\_message(umessage.chat.id, account\_message, reply\_markup=account\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)

### **1.3.8 Реализация навигации между темами и уроками**

При получении запроса course бот отправляет сообщение с клавиатурой, которая содержит кнопки, позволяющие перейти в одну из глав курса, и кнопку, позволяющую вернуться в главное меню. Каждая кнопка отправляет разные запросы, отвечающие за свои главы:

case 'course':  
 themes\_list = types.InlineKeyboardMarkup()  
 themes\_list.row\_width = 1  
 theme1 = types.InlineKeyboardButton('1 | Введение в онлайн-продажи', callback\_data='theme1')  
 theme2 = types.InlineKeyboardButton('2 | Работа с клиентами', callback\_data='theme2')  
 theme3 = types.InlineKeyboardButton('3 | Презентация товара и работа с возражениями', callback\_data='theme3')  
 theme4 = types.InlineKeyboardButton('4 | Завершение сделки и постпродажное обслуживание', callback\_data='theme4')  
 goback = types.InlineKeyboardButton('Назад', callback\_data='main')  
 themes\_list.add(theme1, theme2, theme3, theme4, goback)  
 message\_id = bot.send\_message(umessage.chat.id, 'Выберите тему:', reply\_markup=themes\_list).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id-1)

При получении запроса, соответствующего какой-либо главе бот отправляет сообщение с клавиатурой, которая содержит кнопки, позволяющие перейти в один из уроков главы, и кнопку, позволяющую вернуться к выбору главы. Каждая кнопка отправляет разные запросы, отвечающие за свои уроки:

case 'theme1':  
 themes\_list = types.InlineKeyboardMarkup()  
 themes\_list.row\_width = 1  
 theme\_a = types.InlineKeyboardButton('Особенности онлайн-продаж и их отличие от офлайн-продаж', callback\_data='theme1a')  
 theme\_b = types.InlineKeyboardButton('Основные понятия: конверсия, средний чек, лиды', callback\_data='theme1b')  
 theme\_c = types.InlineKeyboardButton('Роль менеджера по продажам в онлайн-магазине', callback\_data='theme1c')  
 goback = types.InlineKeyboardButton('Назад', callback\_data='course')  
 themes\_list.add(theme\_a, theme\_b, theme\_c, goback)  
 message\_id = bot.send\_message(umessage.chat.id, 'Выберите урок:', reply\_markup=themes\_list).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)

### **1.3.9 Реализация уроков**

При получении запроса, соответствующего какому-либо уроку бот отправляет сообщение с материалами по данному уроку и клавиатурой, которая содержит кнопки, позволяющие открыть видео или статью, и кнопки, позволяющие отметить урок как пройденный или вернуться к выбору урока:

case 'theme1a':  
 image = open('difference.png', 'rb')  
 text = ('Добро пожаловать в увлекательный мир онлайн-продаж! '  
 'Этот раздел заложит фундамент вашего понимания специфики этого канала. '  
 'В отличие от традиционных офлайн-магазинов, где взаимодействие происходит лицом к лицу, онлайн-продажи полагаются на цифровые инструменты и коммуникацию. '  
 'Главное преимущество — это скорость и доступность: клиенты могут совершать покупки в любое время дня и ночи, из любой точки мира, имея лишь доступ к интернету. '  
 'Также важна персонализация: онлайн позволяет собирать данные о клиентах и настраивать предложения под их конкретные потребности и предпочтения. '  
 'Это может быть история покупок, поисковые запросы или поведение на сайте. '  
 'Но, в отличие от офлайна, здесь нет прямого физического контакта. '  
 'Поэтому, мы должны уметь создавать доверительную атмосферу и эффективно общаться с клиентами в цифровом формате, используя текстовые сообщения, чаты, электронную почту и другие средства связи.')  
 article\_link = 'https://softservice-group.ru/online-vs-ofline/'  
 video\_link = 'https://www.youtube.com/watch?v=lLdMnMTYaf0'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_1a')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme1')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)

### **1.3.10 Реализация прогресса**

При прохождении урока, пользователь может отметить урока как пройденный, нажав на кнопку. Она вызывает функцию, передавая данные о пользователе и об уроке, который необходимо учесть как пройденный:

case 'progress\_plus\_1a':  
 progress\_plus(umessage, '1a')

Функция получает данные о пользователе, который пометил урок как пройденный, и индекс этого урока. Эти данные записываются в ранее созданный словарь. Так же производится проверка того, не пройден ли данный урок ранее:

def progress\_plus(message, theme):  
 user\_id = message.chat.id  
 completed\_themes = progress\_data[user\_id]  
 completed\_themes.append(theme)  
 progress\_data[user\_id] = list(set(completed\_themes))

При получении запроса progress из главного меню, бот отправляет сообщение, содержащее данные о прогрессе пользователя и кнопку, позволяющую вернуться в главное меню:

case 'progress':  
 progress\_id = umessage.chat.id  
 completed\_lessons = progress\_data[progress\_id]  
 persents = round(len(completed\_lessons)/12, 2)\*100  
 progress\_message = (f'Ваш прогресс\n'  
 f'\n'  
 f'Курсы:\n'  
 f'"Эффективные продажи в онлайн-магазине"\n'  
 f'{len(completed\_lessons)}/12 ({persents}%)')  
 progress\_keyboard = types.InlineKeyboardMarkup()  
 goback = types.InlineKeyboardButton('Назад', callback\_data='main')  
 progress\_keyboard.add(goback)  
 message\_id = bot.send\_message(umessage.chat.id, progress\_message, reply\_markup=progress\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)

# **2 Реализация алгоритма**

## **2.1 Состав информационной системы**

Разработанный телеграмм-бот предоставляет доступ к онлайн-курсу и включает в себя:

− Приветственное сообщение;

− Создание аккаунта;

− Главное меню;

− 4 главы;

− 12 уроков;

− Отслеживание прогресса.

## **2.2 Приветственное сообщение**

Приветственное окно встречает пользователя при запуске бота и предлагает начать работу, нажав на кнопку. Внешний вид приветственного сообщения представлен на рисунке 8.

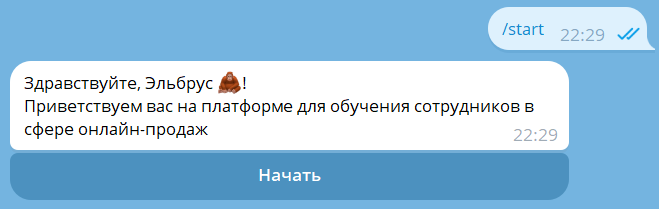


Рисунок 8 − Приветственное сообщение

## **2.3 Создание аккаунта**

После нажатия на кнопку под приветственным сообщением бот запрашивает имя пользователя. После получения ответа он предлагает продолжить работу, нажав на соответствующую кнопку. Внешний вид создания аккаунта представлен на рисунке 9.

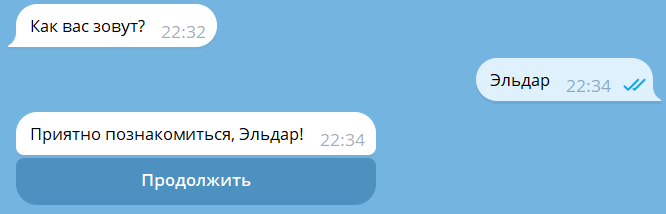


Рисунок 9 – Создание аккаунта

## **2.4 Главное меню**

В главном меню пользователю предоставляется возможность перейти в три раздела бота, а именно: аккаунт, выбор темы и прогресс. Переход в разделы осуществляется с помощью кнопок. Внешний вид главного меню представлен на рисунке 10.

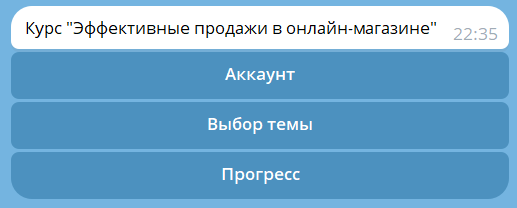


Рисунок 10 – Главное меню

## **2.5 Аккаунт**

В данном разделе пользователь может получить данные об аккаунте, такие как имя пользователя и его текущие курсы. Также он может по нажатию кнопки вернуться в главное меню. Внешний аккаунта представлен на рисунке 11.

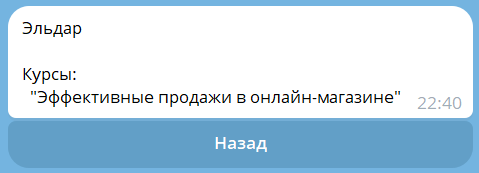


Рисунок 11 – Аккаунт

## **2.6 Выбор темы и урока**

Курс имеет 4 главные темы, каждая тема имеет 3 урока. В данном разделе пользователь имеет возможность перейти в любую тему, а из него в любой урок выбранной темы и получить соответствующую информацию. Также система кнопок даёт возможность возвращаться на уровень выше, позволяя пользователю выбрать другую тему или урок. Внешний вид выбора темы представлен на рисунке 12.

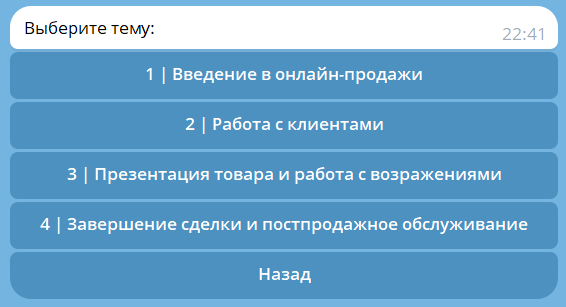


Рисунок 12 –Выбор темы

Внешний вид выбора урока представлен на рисунке 13.

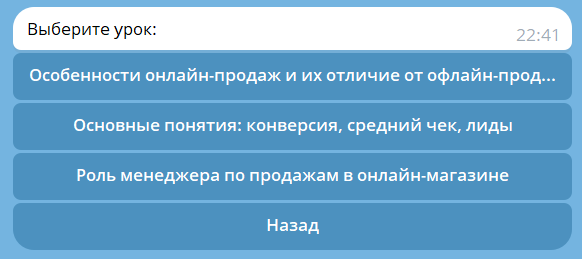


Рисунок 13 –Выбор урока

## **2.8 Урок**

## 

После выбора определенного урока бот предоставляет пользователю информационные материалы по соответствующей теме, такие как: тексты, схемы, изображения, ссылки на статьи и видеоматериалы. Так же можно пометить урок как пройденный или выйти и выбрать другой урок. Внешний вид урока представлен на рисунке 14.

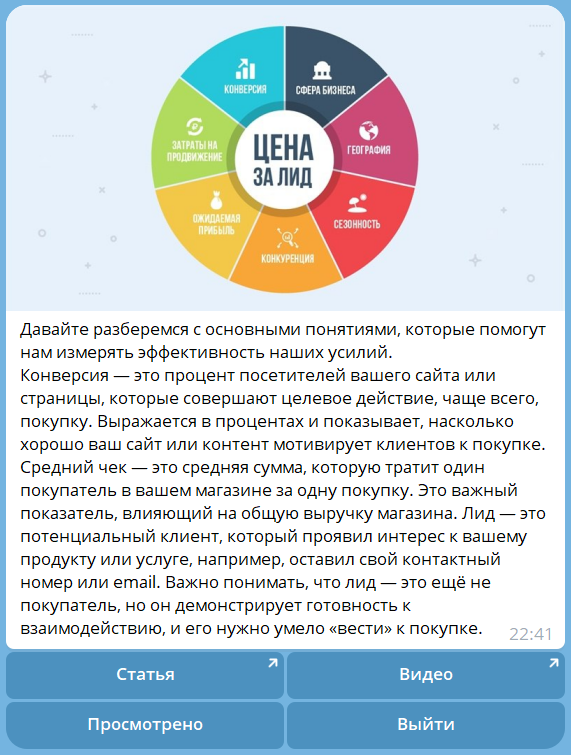


Рисунок 14 – Урок

**2.9 Прогресс**

Данный раздел позволяет пользователю отслеживать свой прогресс прохождения курса и выйти обратно в меню. Здесь отображается следующая информация: количество пройденных уроков, общее количество уроков и их соотношение в процентах. Внешний вид отслеживания прогресса представлен на рисунке 15.

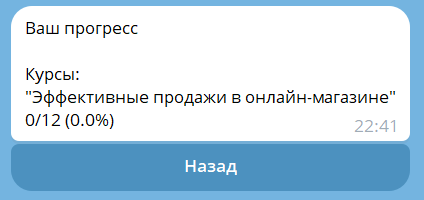


Рисунок 15 – Прогресс

# ЗАКЛЮЧЕНИЕ

В заключение следует отметить что, в рамках данной работы был разработан обучающий телеграмм-бот, является ответом на возрастающую потребность в квалифицированных специалистах в сфере онлайн-продаж, обусловленную стремительным ростом электронной коммерции и усилением конкуренции. Разработанный бот предоставляет интерактивную платформу для подготовки и повышения квалификации персонала, предлагая пользователям доступ к обучающим материалам в текстовом и мультимедийном форматах, а также функционал отслеживания личного прогресса.

Поставленная цель, заключавшаяся в разработке телеграмм-бота для онлайн-обучения, была достигнута посредством решения ряда последовательных задач:

− Изучена сфера онлайн-продаж:

− Изучение инструментов для создания ботов;

− Разработан курс по обучению персонала;

− Реализован телеграмм-бот;

− Проведено тестирование бота.

В результате, был создан инструмент, предоставляющий сотрудникам возможность получать необходимые знания и навыки в удобном и доступном формате, что, в конечном итоге, способствует повышению их профессиональной компетентности и, как следствие, улучшению показателей работы. Таким образом, проект не только демонстрирует возможность использования современных технологий в образовательных целях, но и вносит вклад в развитие квалифицированных кадров в сфере электронной коммерции.

# БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Сходства и различия оффлайн и онлайн торговли [Электронный ресурс]. Режим доступа: <https://www.synapse-studio.ru/blog/skhodstva-i-razlichiya-offlayn-i-onlayn-torgovli>. Дата обращения 10.10.2024

2. pyTelegramBotAPI 4.26.0 [Электронный ресурс]. Режим доступа: [https://pypi.org/project/pyTelegramBotAPI. Дата обращения 10.10.2024](https://pypi.org/project/pyTelegramBotAPI.%20Дата%20обращения%2010.10.2024)

3. telebot быстро и понятно. Телеграмм-бот [Электронный ресурс]. Режим доступа: [<https://habr.com/ru/articles/580408>](https://pypi.org/project/pyTelegramBotAPI). Дата обращения 10.10.2024

# ПРИЛОЖЕНИЕ

**ПРИЛОЖЕНИЕ А**

from telebot import \*  
  
token = '7398398254:AAFa27\_AjW45tXGjZbXSlMou8YH0cnsAx8I'  
bot = telebot.TeleBot(token)  
user\_data = {}  
progress\_data = {}  
  
  
@bot.message\_handler(commands=['start'])  
def welcome\_message(message):  
 welcome\_message = (f'Здравствуйте, {message.from\_user.first\_name}! \n'  
 f'Приветствуем вас на платформе для обучения сотрудников в сфере онлайн-продаж')  
 start\_button\_keyboard = types.InlineKeyboardMarkup()  
 start\_button = types.InlineKeyboardButton('Начать', callback\_data='start')  
 start\_button\_keyboard.add(start\_button)  
 bot.send\_message(message.chat.id, welcome\_message, reply\_markup=start\_button\_keyboard)  
  
def registration(message):  
 user\_id = message.chat.id  
 user\_name = message.text  
 user\_data[user\_id] = user\_name  
 progress\_data[user\_id] = []  
 continue\_keyboard = types.InlineKeyboardMarkup()  
 continue\_button = types.InlineKeyboardButton('Продолжить', callback\_data='main')  
 continue\_keyboard.add(continue\_button)  
 bot.send\_message(message.chat.id, f'Приятно познакомиться, {user\_name}!', reply\_markup=continue\_keyboard)  
  
def progress\_plus(message, theme):  
 user\_id = message.chat.id  
 completed\_themes = progress\_data[user\_id]  
 completed\_themes.append(theme)  
 progress\_data[user\_id] = list(set(completed\_themes))  
  
@bot.callback\_query\_handler(func = lambda call: True)  
def answer(call):  
 option = call.data  
 umessage = call.message  
 match option:  
 case 'start':  
 bot.send\_message(umessage.chat.id, text='Как вас зовут?')  
 bot.register\_next\_step\_handler(umessage, registration)

Продолжение **ПРИЛОЖЕНИЯ А**

case 'main':  
 choose\_topic = types.InlineKeyboardMarkup()  
 account\_b = types.InlineKeyboardButton('Аккаунт', callback\_data='account')  
 theme\_b = types.InlineKeyboardButton('Выбор темы', callback\_data='course')  
 progress\_b = types.InlineKeyboardButton('Прогресс', callback\_data='progress')  
 choose\_topic.row\_width = 1  
 choose\_topic.add(account\_b, theme\_b, progress\_b)  
 message\_id = bot.send\_message(umessage.chat.id, 'Курс "Эффективные продажи в онлайн-магазине"', reply\_markup=choose\_topic).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
  
 case 'account':  
 chatid = umessage.chat.id  
 username = user\_data[chatid]  
 account\_message = (f'{username} \n'  
 f'\n'  
 f'Курсы:\n'  
 f' "Эффективные продажи в онлайн-магазине"')  
 account\_keyboard = types.InlineKeyboardMarkup()  
 account\_button = types.InlineKeyboardButton('Назад', callback\_data='main')  
 account\_keyboard.add(account\_button)  
 message\_id = bot.send\_message(umessage.chat.id, account\_message, reply\_markup=account\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
  
 case 'course':  
 themes\_list = types.InlineKeyboardMarkup()  
 themes\_list.row\_width = 1  
 theme1 = types.InlineKeyboardButton('1 | Введение в онлайн-продажи', callback\_data='theme1')  
 theme2 = types.InlineKeyboardButton('2 | Работа с клиентами', callback\_data='theme2')  
 theme3 = types.InlineKeyboardButton('3 | Презентация товара и работа с возражениями', callback\_data='theme3')  
 theme4 = types.InlineKeyboardButton('4 | Завершение сделки и постпродажное обслуживание', callback\_data='theme4')  
 goback = types.InlineKeyboardButton('Назад', callback\_data='main')  
 themes\_list.add(theme1, theme2, theme3, theme4, goback)  
 message\_id = bot.send\_message(umessage.chat.id, 'Выберите тему:', reply\_markup=themes\_list).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id-1)

Продолжение **ПРИЛОЖЕНИЯ А**

case 'theme1':  
 themes\_list = types.InlineKeyboardMarkup()  
 themes\_list.row\_width = 1  
 theme\_a = types.InlineKeyboardButton('Особенности онлайн-продаж и их отличие от офлайн-продаж', callback\_data='theme1a')  
 theme\_b = types.InlineKeyboardButton('Основные понятия: конверсия, средний чек, лиды', callback\_data='theme1b')  
 theme\_c = types.InlineKeyboardButton('Роль менеджера по продажам в онлайн-магазине', callback\_data='theme1c')  
 goback = types.InlineKeyboardButton('Назад', callback\_data='course')  
 themes\_list.add(theme\_a, theme\_b, theme\_c, goback)  
 message\_id = bot.send\_message(umessage.chat.id, 'Выберите урок:', reply\_markup=themes\_list).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
  
 case 'theme2':  
 themes\_list = types.InlineKeyboardMarkup()  
 themes\_list.row\_width = 1  
 theme\_a = types.InlineKeyboardButton('Техники установления контакта с клиентом в чате', callback\_data='theme2a')  
 theme\_b = types.InlineKeyboardButton('Выявление потребностей и мотивации покупателя', callback\_data='theme2b')  
 theme\_c = types.InlineKeyboardButton('Активное слушание и умение задавать вопросы.', callback\_data='theme2c')  
 goback = types.InlineKeyboardButton('Назад', callback\_data='course')  
 themes\_list.add(theme\_a, theme\_b, theme\_c, goback)  
 message\_id = bot.send\_message(umessage.chat.id, 'Выберите урок:', reply\_markup=themes\_list).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
  
 case 'theme3':  
 themes\_list = types.InlineKeyboardMarkup()  
 themes\_list.row\_width = 1  
 theme\_a = types.InlineKeyboardButton('Как эффективно презентовать товар онлайн', callback\_data='theme3a')  
 theme\_b = types.InlineKeyboardButton('Описание преимуществ и выгод для клиента', callback\_data='theme3b')  
 theme\_c = types.InlineKeyboardButton('Работа с распространенными возражениями клиентов', callback\_data='theme3c')  
 goback = types.InlineKeyboardButton('Назад', callback\_data='course')  
 themes\_list.add(theme\_a, theme\_b, theme\_c, goback)  
 message\_id = bot.send\_message(umessage.chat.id, 'Выберите урок:', reply\_markup=themes\_list).message\_id

Продолжение **ПРИЛОЖЕНИЯ А**

bot.delete\_message(umessage.chat.id, message\_id - 1)  
  
 case 'theme4':  
 themes\_list = types.InlineKeyboardMarkup()  
 themes\_list.row\_width = 1  
 theme\_a = types.InlineKeyboardButton('Техники завершения сделки', callback\_data='theme4a')  
 theme\_b = types.InlineKeyboardButton('Сопровождение клиента после покупки', callback\_data='theme4b')  
 theme\_c = types.InlineKeyboardButton('Сбор обратной связи и повышение лояльности', callback\_data='theme4c')  
 goback = types.InlineKeyboardButton('Назад', callback\_data='course')  
 themes\_list.add(theme\_a, theme\_b, theme\_c, goback)  
 message\_id = bot.send\_message(umessage.chat.id, 'Выберите урок:', reply\_markup=themes\_list).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
  
 case 'theme1a':  
 image = open('difference.png', 'rb')  
 text = ('Добро пожаловать в увлекательный мир онлайн-продаж! '  
 'Этот раздел заложит фундамент вашего понимания специфики этого канала. '  
 'В отличие от традиционных офлайн-магазинов, где взаимодействие происходит лицом к лицу, онлайн-продажи полагаются на цифровые инструменты и коммуникацию. '  
 'Главное преимущество — это скорость и доступность: клиенты могут совершать покупки в любое время дня и ночи, из любой точки мира, имея лишь доступ к интернету. '  
 'Также важна персонализация: онлайн позволяет собирать данные о клиентах и настраивать предложения под их конкретные потребности и предпочтения. '  
 'Это может быть история покупок, поисковые запросы или поведение на сайте. '  
 'Но, в отличие от офлайна, здесь нет прямого физического контакта. '  
 'Поэтому, мы должны уметь создавать доверительную атмосферу и эффективно общаться с клиентами в цифровом формате, используя текстовые сообщения, чаты, электронную почту и другие средства связи.')  
 article\_link = 'https://softservice-group.ru/online-vs-ofline/'  
 video\_link = 'https://www.youtube.com/watch?v=lLdMnMTYaf0'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео',

Продолжение **ПРИЛОЖЕНИЯ А**

url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_1a')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme1')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_1a':  
 progress\_plus(umessage, '1a')  
  
 case 'theme1b':  
 image = open('leads.jpg', 'rb')  
 text = ('Давайте разберемся с основными понятиями, которые помогут нам измерять эффективность наших усилий. \n'  
 'Конверсия — это процент посетителей вашего сайта или страницы, которые совершают целевое действие, чаще всего, покупку. '  
 'Выражается в процентах и показывает, насколько хорошо ваш сайт или контент мотивирует клиентов к покупке. '  
 ''  
 'Средний чек — это средняя сумма, которую тратит один покупатель в вашем магазине за одну покупку. '  
 'Это важный показатель, влияющий на общую выручку магазина. '  
 ''  
 'Лид — это потенциальный клиент, который проявил интерес к вашему продукту или услуге, например, оставил свой контактный номер или email. '  
 'Важно понимать, что лид — это ещё не покупатель, но он демонстрирует готовность к взаимодействию, и его нужно умело «вести» к покупке.')  
 article\_link = 'https://www.advantshop.net/blog/analitika/kpi-v-onlain-torgovle-kak-izmerit-uspeshnost-raboty-biznesa'  
 video\_link = 'https://www.youtube.com/watch?v=C36kGAVloQc'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_1b')  
 goback = types.InlineKeyboardButton('Выйти',

Продолжение **ПРИЛОЖЕНИЯ А**

callback\_data='theme1')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_1b':  
 progress\_plus(umessage, '1b')  
  
 case 'theme1c':  
 text = ('Роль менеджера по продажам в онлайн-магазине выходит за рамки простого приема заказов. '  
 'Вы - ключевая фигура, которая формирует восприятие компании в цифровом пространстве. '  
 'Ваша работа включает не только помощь в выборе товара и оформление заказа, но и создание позитивного опыта для каждого клиента. '  
 'Вы должны быть не только отличными продавцами, но и экспертами в товаре, а также консультантами, способными понять потребности клиента и предложить наилучшее решение. '  
 'Важны коммуникабельность, эмпатия, умение быстро реагировать на запросы, а также техническая грамотность, поскольку вся работа происходит в цифровой среде. '  
 'Вы — лицо компании в онлайн-пространстве, и именно от вас зависит, вернется ли клиент к нам снова.')  
 article\_link = 'https://prime-ltd.su/blog/rol-menedzhera-v-internet-magazine/'  
 video\_link = 'https://www.youtube.com/watch?v=Eu9HBNh71uo'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_1c')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme1')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_message(umessage.chat.id, text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_1c':  
 progress\_plus(umessage, '1c')  
  
 case 'theme2a':  
 image = open('communication.jpg', 'rb')

Продолжение **ПРИЛОЖЕНИЯ А**

text = ('Успешное установление контакта — это фундамент для продуктивного диалога и успешной продажи. '  
 'Первое сообщение должно быть вежливым, быстрым и персонализированным, насколько это возможно. '  
 'Забудьте про шаблонные фразы - старайтесь обращаться к клиенту по имени, если это возможно, и покажите, что вы готовы помочь. '  
 'Внимательно следите за временем ответа; долгие задержки могут отпугнуть клиента. '  
 'Используйте приветственные фразы, которые создают позитивный настрой и показывают вашу готовность к диалогу. '  
 'Проявите искренний интерес к запросу клиента, чтобы он почувствовал вашу заботу и вовлеченность. '  
 'Начните разговор с открытого вопроса, который поможет клиенту описать свою потребность.')  
 article\_link = 'https://www.kom-dir.ru/article/3580-ustanovlenie-kontakta-v-prodajah'  
 video\_link = 'https://www.youtube.com/watch?v=plYGH-P5QQA'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_2a')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme2')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_2a':  
 progress\_plus(umessage, '2a')  
  
 case 'theme2b':  
 image = open('needs.png', 'rb')  
 text = ('Чтобы предложить клиенту именно то, что ему нужно, необходимо научиться правильно выявлять его потребности и мотивацию. '  
 'Задавайте открытые вопросы, требующие развернутого ответа, а не просто «да» или «нет». '  
 'Внимательно слушайте ответы, обращайте внимание на детали и ключевые слова, которые помогут вам понять истинные намерения клиента. '  
 'Постарайтесь понять, какую проблему клиент хочет решить с помощью вашего продукта или услуги, какие

Продолжение **ПРИЛОЖЕНИЯ А**

характеристики и качества для него наиболее важны. '  
 'Используйте технику «5 почему», если это необходимо, чтобы добраться до самой сути проблемы. '  
 'Важно понять не только, что клиент хочет, но и почему он этого хочет, что его мотивирует к покупке.')  
 article\_link = 'https://kontur.ru/articles/6237'  
 video\_link = 'https://www.youtube.com/watch?v=bP9niNRcycE'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_2b')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme2')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_2b':  
 progress\_plus(umessage, '2b')  
  
 case 'theme2c':  
 image = open('listening.png', 'rb')  
 text = ('Активное слушание — это не просто молчание во время разговора, а сознательное усилие понять собеседника. '  
 'Оно заключается в том, чтобы внимательно слушать, что говорит клиент, не перебивая, и показывать, что вы вовлечены в разговор. '  
 'Перефразируйте слова клиента, чтобы убедиться, что вы правильно его поняли, и задавайте уточняющие вопросы, чтобы получить более точную информацию. '  
 'Используйте эмпатию, чтобы показать клиенту, что вы понимаете его чувства и переживания. '  
 'Активное слушание и правильные вопросы не только помогут вам получить необходимую информацию, но и помогут клиенту почувствовать себя ценным и важным, что увеличит его лояльность и готовность к покупке.')  
 article\_link = 'https://kontur.ru/talk/spravka/52358-aktivnoe\_slushanie\_v\_kakih\_situaciyah\_prigoditsya'  
 video\_link = 'https://www.youtube.com/watch?v=GHAg1AXPZx8'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья',

Продолжение **ПРИЛОЖЕНИЯ А**

url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_2c')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme2')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_2c':  
 progress\_plus(umessage, '2c')  
  
 case 'theme3a':  
 image = open('AIDA.png', 'rb')  
 text = ('Презентация товара в онлайн-магазине должна быть максимально привлекательной и информативной. '  
 'Используйте качественные фотографии и видео, которые показывают товар с разных ракурсов и в разных ситуациях использования. '  
 'Подробное и точное описание характеристик и особенностей товара — обязательное условие. '  
 'Выделите ключевые преимущества и покажите, как они могут решить проблемы клиента или улучшить его жизнь. '  
 'Не забудьте про призыв к действию, например, "Добавить в корзину", "Купить сейчас" или "Узнать больше". '  
 'Важно не просто перечислить характеристики, а показать выгоды для клиента, используя эмоциональные описания, которые вызывают интерес и желание приобрести товар. '  
 'Создайте такое впечатление, что товар уже находится в руках у клиента.')  
 article\_link = 'https://www.advantshop.net/blog/start/kak-effektivno-prezentovat-tovar-v-internet-magazine'  
 video\_link = 'https://www.youtube.com/watch?v=SDdmX9tEjX4'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_3a')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme3')  
 goback\_keyboard.add(article, video, complete,

Продолжение **ПРИЛОЖЕНИЯ А**

goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_3a':  
 progress\_plus(umessage, '3a')  
  
 case 'theme3b':  
 image = open('advantages.jpg', 'rb')  
 text = ('Ключ к успешной презентации — это умение трансформировать характеристики товара в конкретные выгоды для клиента. '  
 'Не зацикливайтесь на технических деталях, покажите, как эти детали улучшают жизнь клиента. '  
 'Например, если у телефона есть мощный аккумулятор, не просто скажите, что он "мощный", а объясните, что это означает, что клиенту не придется постоянно беспокоиться о зарядке в течение дня. '  
 'Используйте язык выгод, который понятен и близок клиенту. '  
 'Старайтесь понять, что именно для него важно, и подчеркивайте именно эти аспекты. '  
 'Не забывайте, что клиенты покупают не товары, а решения своих проблем и удовлетворение своих потребностей.')  
 article\_link = 'https://www.calltouch.ru/blog/hpv-v-prodazhah-kak-ispolzovat-tehniku-harakteristika-preimushhestvo-vygoda-dlya-uvelicheniya-prodazh/'  
 video\_link = 'https://www.youtube.com/watch?v=CHxNjex9x2M'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_3b')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme3')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_3b':  
 progress\_plus(umessage, '3b')  
  
 case 'theme3c':  
 image = open('objection.webp', 'rb')

Продолжение **ПРИЛОЖЕНИЯ А**

text = ('Возражения клиентов — это неотъемлемая часть процесса продаж. '  
 'Не стоит их бояться, скорее, наоборот, рассматривать как возможность убедить клиента в ценности вашего товара и отработать его сомнения. '  
 'Типичные возражения, такие как «дорого» или «я подумаю», должны быть отработаны с уверенностью и профессионализмом. '  
 'Вместо того, чтобы спорить с клиентом, используйте эмпатию, чтобы понять его точку зрения. '  
 'Предлагайте альтернативные варианты, если цена вызывает вопросы, или дайте клиенту время подумать, но обязательно предложите вашу помощь в случае возникновения вопросов. '  
 'Главное — оставаться позитивным и уверенным, не теряя настойчивости.')  
 article\_link = 'https://www.insales.ru/blogs/university/kak-rabotat-s-vozrazheniyami'  
 video\_link = 'https://www.youtube.com/watch?v=il54IlwQBKQ'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_3c')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme3')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_3c':  
 progress\_plus(umessage, '3c')  
  
 case 'theme4a':  
 image = open('finish.jpg', 'rb')  
 text = ('Завершение сделки — это момент, когда вы мягко, но уверенно подталкиваете клиента к покупке. '  
 'Используйте техники, которые помогут вам перейти от диалога к действию. \n'  
 'Техника "альтернативный выбор" — это предложение клиенту выбора между несколькими вариантами товара, что помогает принять решение. \n'  
 'Техника "резюме" — это краткое повторение всех преимуществ и выгод выбранного товара, чтобы клиент еще раз

Продолжение **ПРИЛОЖЕНИЯ А**

убедился в своем выборе. \n'  
 'Техника "ограниченное предложение" — это создание срочности, например, через скидки или акции, которые действуют только ограниченное время. \n'  
 'Важно не давить на клиента, а мягко подвести его к завершению сделки, показав, что вы готовы помочь ему сделать правильный выбор.')  
 article\_link = 'https://aspro.cloud/crm/docs/zavershenie-sdelki/'  
 video\_link = 'https://www.youtube.com/watch?v=nSFwoiNuXfk'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_4a')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme4')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_4a':  
 progress\_plus(umessage, '4a')  
  
 case 'theme4b':  
 text = ('Сопровождение клиента после покупки — это не менее важный этап, чем сама продажа. '  
 'Не забывайте о клиенте после того, как он совершил покупку. '  
 'Отправьте ему благодарственное письмо или сообщение, предложите свою помощь в случае возникновения вопросов или проблем. '  
 'Поддерживайте связь с клиентом, узнавайте его впечатления от товара, предлагайте специальные условия для повторных покупок. '  
 'Положительный опыт после покупки повышает лояльность клиентов и увеличивает вероятность того, что он вернется к вам снова и посоветует ваш магазин своим друзьям. '  
 'Важно показать клиенту, что вы цените его доверие и готовы поддержать его на всех этапах взаимодействия.')  
 article\_link = 'https://www.mango-office.ru/journal/for-marketing/prodazhi/posleprodazhnoe-obsluzhivanie/'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2

Продолжение **ПРИЛОЖЕНИЯ А**

article = types.InlineKeyboardButton('Статья', url=article\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_4b')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme4')  
 goback\_keyboard.add(article, complete, goback)  
 message\_id = bot.send\_message(umessage.chat.id, text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
 case 'progress\_plus\_4b':  
 progress\_plus(umessage, '4b')  
  
 case 'theme4c':  
 image = open('feedback.jpeg', 'rb')  
 text = ('Сбор обратной связи — это не просто возможность услышать мнение клиентов, но и мощный инструмент для улучшения вашей работы. '  
 'Предлагайте клиентам оставлять отзывы о товарах и о качестве обслуживания. '  
 'Прислушивайтесь к их мнению, анализируйте их замечания и используйте эту информацию для оптимизации процессов. '  
 'Важно не только собирать обратную связь, но и активно реагировать на нее, отвечая на вопросы и решая проблемы клиентов. '  
 'Своевременная реакция на обратную связь показывает, что вы заботитесь о клиентах и делаете все возможное, чтобы обеспечить их удовлетворенность. '  
 'Это один из лучших способов повысить лояльность клиентов и построить с ними долгосрочные отношения.')  
 article\_link = 'https://habr.com/ru/companies/planado/articles/311482/'  
 video\_link = 'https://www.youtube.com/watch?v=fCZFTI6hpmw'  
 goback\_keyboard = types.InlineKeyboardMarkup()  
 goback\_keyboard.row\_width = 2  
 article = types.InlineKeyboardButton('Статья', url=article\_link)  
 video = types.InlineKeyboardButton('Видео', url=video\_link)  
 complete = types.InlineKeyboardButton('Просмотрено', callback\_data='progress\_plus\_4c')  
 goback = types.InlineKeyboardButton('Выйти', callback\_data='theme4')  
 goback\_keyboard.add(article, video, complete, goback)  
 message\_id = bot.send\_photo(umessage.chat.id, image, caption=text, reply\_markup=goback\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)

Продолжение **ПРИЛОЖЕНИЯ А**

case 'progress\_plus\_4c':  
 progress\_plus(umessage, '4c')  
  
 case 'progress':  
 progress\_id = umessage.chat.id  
 completed\_lessons = progress\_data[progress\_id]  
 persents = round(len(completed\_lessons)/12, 2)\*100  
 progress\_message = (f'Ваш прогресс\n'  
 f'\n'  
 f'Курсы:\n'  
 f'"Эффективные продажи в онлайн-магазине"\n'  
 f'{len(completed\_lessons)}/12 ({persents}%)')  
 progress\_keyboard = types.InlineKeyboardMarkup()  
 goback = types.InlineKeyboardButton('Назад', callback\_data='main')  
 progress\_keyboard.add(goback)  
 message\_id = bot.send\_message(umessage.chat.id, progress\_message, reply\_markup=progress\_keyboard).message\_id  
 bot.delete\_message(umessage.chat.id, message\_id - 1)  
  
bot.polling()