|  |  |  |
| --- | --- | --- |
|  |  | УТВЕРЖДАЮ |
|  |  | Руководитель  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.  М.П. |

ТЕХНИЧЕСКИЙ ПРОЕКТ

Онлайн-ассистент для интернет-магазина

2024

Содержание

[1 Уточнение структур данных 2](#_Toc182672833)

[2 Функциональные требования 3](#_Toc182672834)

[3 Не функциональные требования 4](#_Toc182672835)

[4 Архитектура системы 5](#_Toc182672836)

[5 Этапы разработки 6](#_Toc182672837)

[6 Оценка рисков 7](#_Toc182672838)

[7 Будущие возможности 8](#_Toc182672839)

# Уточнение структур данных

Для эффективной работы онлайн-ассистента интернет-магазина необходимо определить несколько ключевых структур данных:

* Пользователь: Включает уникальный идентификатор, имя, электронную почту, номер телефона, дату регистрации, историю покупок и предпочтения (язык и валюта).
* Товар: Хранит уникальный идентификатор, название, описание, категорию, цену, количество на складе, характеристики (цвет, размер, бренд), изображения и информацию о рейтинге (средний рейтинг и количество отзывов).
* Заказ: Содержит уникальный идентификатор, информацию о пользователе, дату заказа, статус, общую сумму, список товаров в заказе, адрес доставки и информацию о платеже (метод, идентификатор транзакции и статус платежа).
* Взаимодействие с пользователем: Включает уникальный идентификатор взаимодействия, информацию о пользователе, время взаимодействия, сообщение и ответ ассистента, а также контекст (идентификатор сессии, намерение пользователя и упомянутые сущности).

Эти структуры данных помогут ассистенту эффективно управлять информацией и улучшать взаимодействие с клиентами.

## Функциональные требования

* 1. **Основные функции ассистента**
     1. Поиск товаров:
* Пользователь может задать вопрос о товаре (например, "Покажи мне смартфоны").
* Ассистент должен предоставлять список товаров, соответствующих запросу.
  + 1. Информация о товаре:
* Пользователь может запросить информацию о конкретном товаре (например, "Расскажи о характеристиках iPhone 13").
* Ассистент должен предоставить описание, характеристики и цену товара.
  + 1. Помощь с заказом:
* Пользователь может запросить помощь с оформлением заказа (например, "Как оформить заказ?").
* Ассистент должен предоставить пошаговую инструкцию по оформлению заказа.
  + 1. Часто задаваемые вопросы (FAQ):
* Ассистент должен отвечать на часто задаваемые вопросы о доставке, возврате, оплате и т.д.
  + 1. Поддержка пользователей:
* Ассистент должен предоставлять возможность связаться с оператором в случае сложных вопросов.
  1. **Дополнительные функции**
     1. Персонализированные рекомендации:
* Ассистент может рекомендовать товары на основе истории покупок пользователя.
  + 1. Интеграция с CRM:
* Ассистент должен иметь возможность интеграции с системой управления клиентами (CRM) для отслеживания взаимодействий с пользователями.

# Не функциональные требования

* 1. Производительность:
* Время отклика ассистента должно составлять не более 2 секунд.
  1. Безопасность:
* Все данные пользователей должны быть защищены, и необходимо обеспечить соответствие GDPR.
  1. Доступность:
* Ассистент должен быть доступен 24/7.
  1. Масштабируемость:
* Система должна быть способна обрабатывать до 1000 одновременных пользователей.

# Архитектура системы

* 1. Компоненты системы
     1. Frontend:
* Веб-интерфейс для взаимодействия с пользователями.
* Чат-бот, встроенный в веб-сайт интернет-магазина.
  + 1. Backend:
* Серверная часть, обрабатывающая запросы от пользователей.
* База данных для хранения информации о товарах, пользователях и взаимодействиях.
  + 1. API:
* RESTful API для взаимодействия между фронтендом и бэкендом.
  + 1. Машинное обучение:
* Модели для обработки естественного языка (NLP) для понимания запросов пользователей.
  1. Технологический стек
     1. Frontend:
* HTML, CSS, JavaScript
* Фреймворк: React или Vue.js
  + 1. Backend:
* Язык программирования: Python или Node.js
* Фреймворк: Django или Express.js
  + 1. База данных:
* Реляционная СУБД: PostgreSQL или MySQL
  + 1. Машинное обучение:
* Библиотеки: TensorFlow, PyTorch, spaCy
  + 1. Хостинг:
* Облачные платформы: AWS, Google Cloud или Azure

# Этапы разработки

* 1. Анализ требований:
* Сбор и анализ требований от заинтересованных сторон.
  1. Проектирование архитектуры:
* Разработка архитектуры системы и выбор технологий.
  1. Разработка:
* Создание фронтенда и бэкенда.
* Интеграция с базой данных и API.
  1. Тестирование:
* Проведение функционального и нагрузочного тестирования.
  1. Развертывание:
* Развертывание системы на облачной платформе.
  1. Поддержка и обновление:
* Обеспечение технической поддержки и регулярных обновлений.

# Оценка рисков

1. Технические риски:

* Возможные проблемы с интеграцией различных компонентов.

1. Риски безопасности:

* Утечка данных пользователей.

1. Риски производительности:

* Невозможность обработки большого объема запросов в пиковые часы, что может привести к снижению качества обслуживания клиентов.

1. Риски пользовательского опыта:

* Неправильное понимание запросов пользователей может привести к недовольству и потере клиентов.

1. Риски изменения требований:

* Изменения в бизнес-требованиях или законодательстве могут потребовать внесения изменений в проект.

# Будущие возможности

* 1. Интеграция с другими каналами:
* Расширение функциональности ассистента для работы с мессенджерами (например, WhatsApp, Facebook Messenger) и голосовыми помощниками (например, Alexa, Google Assistant).
  1. Использование искусственного интеллекта:
* Внедрение более сложных алгоритмов машинного обучения для улучшения качества обработки естественного языка и персонализации рекомендаций.
  1. Расширение функциональности:
* Добавление новых функций, таких как отслеживание заказов, управление возвратами и обратная связь от пользователей.
  1. Адаптация к новым рынкам:
* Модификация ассистента для работы на новых рынках с учетом культурных и языковых особенностей.