**СОДЕРЖАНИЕ**

Введение

ВВЕДЕНИЕ

В современном мире организация бизнеса в сфере ювелирных изделий требует использования инновационных подходов и технологий. В условиях постпандемической реальности спрос на онлайн-сервисы значительно вырос, и разработка такого приложения позволит компаниям адаптироваться к новым требованиям рынка.

На данный момент существуют следующие основные противоречия между необходимостью применения программного обеспечения для заказа ювелирных изделий и методами оформления заказов:

* заказ при личной встрече с клиентом или по телефону требует значительного времени на согласование дизайна и деталей, в то время как приложение позволяет ускорить этот процесс и сделать его более удобным;
* при ручном вводе возможны ошибки при заполнении бланков;
* необходимость содержать большее количество менеджеров для работы с клиентами, чем при использовании приложения;
* традиционные способы заказа ограничивают возможность детального анализа и понимания потребностей клиентов.

Разработка приложения для оформления заказов в магазине ювелирных изделий является крайне актуальной, так как оно позволит упростить и ускорить процесс заказа, позволит снизить процент ошибок при вводе данных, повысит доступность услуг магазина, увеличит количество клиентов.

Применение приложения также позволит оптимизировать работу ювелирного магазина за счет интеграции с внутренними системами управления. Оно обеспечит эффективное отслеживание статусов заказов и управление производственными процессами, сбор данных о заказах и анализ предпочтений клиентов для прогнозирования спроса. Это будет способствовать более эффективному управлению ресурсами, улучшению сервиса и увеличению прибыли магазина.

При разработке приложения для оформления заказов в ювелирном магазине можно выделить несколько ключевых проблем исследования, которые необходимо учесть для успешного внедрения и функционирования системы:

* понимание потребностей пользователей: какие функции и возможности они хотели бы видеть в новом приложении?
* разработка интуитивно понятного и удобного интерфейса, который будет доступен для пользователей с разным уровнем технической компетенции: какие элементы дизайна и навигации будут наиболее эффективными для упрощения процесса оформления заказа?
* интеграция с существующими программными системами и базами данных ювелирного магазина: как обеспечить бесшовную интеграцию, минимизируя при этом возможные сбои в работе?
* управление данными и безопасность: какие меры безопасности необходимо внедрить, чтобы гарантировать конфиденциальность и защиту информации?
* тестирование и отладка: какие тесты необходимо выполнить на различных этапах разработки приложения? Какие будут сценарии использования приложения в реальных условиях?
* адаптивность приложения для различных устройств (мобильные телефоны, планшеты, персональный компьютер) и масштабируемость его для будущих расширений функционала: как заложить такие характеристики в архитектуре приложения?
* оценка потребности в обучении сотрудников и клиентов при переходе на новое приложение: как разработать программу обучения, которая бы облегчила процесс адаптации пользователей к новым технологиям?

Каждый из этих аспектов требует детального исследования и тщательной проработки для успешного внедрения приложения для оформления заказов в ювелирном магазине, чтобы обеспечить его эффективность, удобство и безопасность.

Объект исследования – процесс разработки модулей программного обеспечения для компьютерных систем.

Предмет исследования – программное обеспечение для оформления заказов в ювелирном магазине.

Цель курсового проекта: разработать технический проект для приложения, которое позволит оптимизировать процесс оформления заказов в ювелирном магазине.

Задачи курсового проекта:

* провести исследование для определения конкретных потребностей и предпочтений целевой аудитории (как клиентов, так и сотрудников ювелирного магазина);
* спроектировать структуру и функциональность приложения, выбрать инструменты для разработки приложения.

Гипотеза исследования: автоматизация процесса оформления заказов в ювелирном магазине через приложение с удобным и интуитивно понятным интерфейсом уменьшит время, необходимое клиентам для завершения заказа, повысит точность данных и снизит количество ошибок в заказах по сравнению с текущими процессами, а также позволит ювелирному магазину получать аналитическую информацию о предпочтениях клиентов и динамике заказов, что поможет в дальнейшем развитии бизнеса.

При изучении проблем разработки приложения для оформления заказов в ювелирном магазине будут применяться следующие методы исследования:

* опросы и анкетирование: проведение опросов среди потенциальных пользователей (клиентов и сотрудников ювелирных магазинов) с целью сбора информации о их потребностях, предпочтениях и текущих проблемах в процессе оформления заказа;
* анализ конкурентов: изучение существующих приложений на рынке, которые уже решают аналогичные задачи. Анализ их функциональности, пользовательского интерфейса и отзывов пользователей поможет выявить лучшие практики и возможные недостатки;
* пользовательское тестирование: создание прототипов и их тестирование на целевой аудитории. Получение обратной связи об удобстве использования и функциональности на ранних этапах разработки;
* моделирование сценариев использования: разработка сценариев, описывающих разные пути, которыми могут следовать пользователи в процессе оформления заказа, что позволит понять возможные проблемы и улучшить интерфейс;
* кросс-дисциплинарные исследования: вовлечение специалистов из различных областей (дизайнеров, разработчиков, маркетологов) в процессе исследования поможет обеспечить комплексный подход к разработке приложения;
* объектно-ориентированный анализ: проектирование архитектуры и функциональности приложения, построения моделей поведения системы;
* анализ современного инструментария для разработки программного обеспечения для компьютерных систем;
* методы математической статистики.
* алгоритмы применения инструментальных средств на различных этапах разработки программного обеспечения.

Практическая значимость курсового проекта заключается в том, что представленное в нем описание процесса проектирования и разработки приложения для заказа ювелирных изделий позволит проанализировать принятые на всех этапах решения и, при необходимости, внести изменения.

Основными пользователями разработанного приложения будут клиенты ювелирного магазина, а также менеджеры по работе с клиентами данной организации.

Аспекты проектирования программного продукта будут рассмотрены в первом разделе курсового проекта.

**1 ТЕХНИЧЕСКИЙ ПРОЕКТ**

**1.1 Анализ предметной области**

В ювелирной отрасли важнейшими аспектами успешного бизнеса являются удовлетворение потребностей клиентов и повышение эффективности работы сотрудников. Проблемы, с которыми сталкиваются ювелирные магазины, включают долгий процесс оформления заказов, риски ошибок при оформлении вручную и ограниченные возможности анализа предпочтений клиентов. Для решения этих задач необходимо внедрение специализированного приложения, которое позволит автоматизировать процесс оформления заказов и улучшить взаимодействие с клиентами.

Приложение для оформления заказов в ювелирном магазине станет важным инструментом для оптимизации бизнес-процессов, предлагая удобные функции для клиентов и сотрудников. Его внедрение способно значительно упростить обслуживание, ускорить обработку заказов и обеспечить удобный доступ к информации для анализа клиентских предпочтений.

Приложение должно обладать следующими характеристиками:

* удобство и простота использования – клиенты должны легко находить (просматривать каталог изделий, добавлять товары в корзину) и заказывать изделия, управлять своим заказом, а сотрудники – обрабатывать заявки с минимальными задержками. Простой и интуитивно понятный интерфейс поможет пользователям быстро освоить приложение и эффективно использовать его;
* централизованное управление – приложение должно позволять сотрудникам ювелирного магазина контролировать весь процесс заказа из одного интерфейса. Это включает ведение каталога товаров, управление заказами, отслеживание их статуса и работу с клиентскими запросами, что облегчит обработку заявок и улучшит качество обслуживания;
* автоматизация рутинных задач – рутинные операции, такие как подтверждение заказа, отправка уведомлений и обновление статусов, должны быть автоматизированы. Это снизит нагрузку на сотрудников и сократит вероятность ошибок, что особенно важно в ювелирном бизнесе, где важна каждая деталь;
* возможность сбора и анализа данных о заказах – приложение должно автоматизировать и упрощать выявление популярных изделий и общих тенденций спроса, что позволит управлять ассортиментом более обоснованно и эффективно;
* поддержка обратной связи – приложение должно позволять собирать отзывы и пожелания клиентов, помогая ювелирному магазину постоянно улучшать качество обслуживания;
* масштабируемость и адаптивность – приложение должно поддерживать работу на различных устройствах и легко адаптироваться к возможным будущим изменениям, таким как добавление новых категорий товаров или функций.

**1.2 Постановка задачи**

Необходимо разработать веб-приложение для автоматизации процесса оформления заказов в ювелирном магазине.

Приложение должно использовать данные, хранящиеся в базе данных. Информация о заказах, клиентах и прикрепленных файлах должна быть приведена к третьей нормальной форме (3НФ).

Приложение следует реализовать с использованием трехуровневой архитектуры, разработав API для обработки бизнес-логики.

Приложение для оформления заказов в ювелирном магазине должно обеспечивать следующие возможности:

* авторизация пользователей, имеющих доступ к приложению, для защиты данных и персонализированного опыта;
* изменение данных – функция для авторизованных пользователей, позволяющая редактировать профили, настройки и другие данные, связанные с заказами и учетной записью клиента;
* добавление данных – возможность для авторизованных пользователей добавлять новую информацию, например, создавать новые заказы, добавлять товары в каталог или обновлять их описание;
* удаление данных – функция, позволяющая удалять устаревшие или некорректные данные, поддерживая актуальность информации о товарах и заказах;
* получение информации – пользователи должны иметь доступ к актуальным данным о товарах, услугах и статусах заказов;
* поиск информации – возможность быстрого поиска нужных товаров, заказов или информации о клиентах, чтобы упростить процесс оформления заказа;
* обеспечение безопасности данных – приложение должно гарантировать защиту данных пользователей и заказов от несанкционированного доступа, обеспечивая высокий уровень безопасности.

Программный продукт должен включать в себя:

* API для бизнес-логики, обеспечивающий обработку и управление процессами оформления заказов и взаимодействие между клиентской и серверной частями;
* файловое хранилище для хранения изображений и документов, прикрепленных пользователем, таких как фотографии изделий и сертификаты;
* веб-приложение для управления данными ювелирного магазина, позволяющее добавлять, изменять и удалять информацию о товарах, управлять заказами и просматривать аналитику.

План реализации программного продукта включает:

* Исследование требований и проектирование архитектуры системы, включая структуру базы данных, API для взаимодействия с данными, а также интерфейсы для клиентов и сотрудников магазина.
* Создание API для доступа к информации о товарах, заказах, аутентификации пользователей и обеспечении безопасности данных.
* Реализация веб-приложения для администрирования данных в базе данных, включая добавление, изменение и удаление сведений о продуктах и заказах.
* Развертывание системы и обеспечение её стабильной работы.

Требования по входным данным:

* Запросы пользователей — данные, формулируемые клиентами и менеджерами для поиска информации о товарах, заказах и статусах.
* Данные администраторов — информация, необходимая для управления данными в приложении, в том числе добавление и редактирование информации о товарах и заказах.
* Данные аутентификации и учетной записи — информация, предоставляемая клиентами и администраторами при регистрации и входе в систему, включая имя пользователя, электронную почту и пароль для идентификации.

Требования по выходным данным:

* Информация о товарах – данные о доступных украшениях, включая описание, цену, материалы и изображения. Пользователи смогут просматривать карточки товаров и выбирать их для заказа.
* Детали заказов – информация о заказах, созданных пользователями, включая статус выполнения, номер заказа, перечень товаров и итоговую стоимость.
* Интерфейс администратора – административный интерфейс, предоставляющий функции для управления данными о товарах, заказами и пользователями. Администраторы смогут добавлять, редактировать и удалять информацию, связанную с каталогом товаров и заказами.
* Ответы на запросы от API – данные, предоставляемые внешним приложениям по API в формате JSON. Они содержат информацию о товарах, доступных для заказа, и статусах заказов.
* Данные учетных записей – данные о профилях клиентов и администраторов, включая контактную информацию и историю заказов, доступные для просмотра и редактирования в личном кабинете или административной панели.

Данные запросов должны поступать на API и отправляться внешним приложениями в формате JSON.

Данные должны храниться и передаваться в безопасной и защищенной форме. Доступ к данным администраторов должен быть строго контролируемым через аутентификацию. Данные администраторов должны храниться в зашифрованной форме, и должны быть приняты меры для обеспечения их конфиденциальности для предупреждения несанкционированного доступа к получению прав администратора.

**1.3 Проектирование функциональности веб-сайта**

Для описания взаимодействия пользователей и администраторов с ювелирным магазином воспользуемся методологией UML.

Рассмотрим на диаграмме вариантов использования взаимодействие пользователей с разрабатываемым приложением. Для этого построим диаграмму Use Case, в которой имеются:

* актеры («Пользователь», «Администратор», «API», «DB»);
* прецеденты («Авторизация», «Просмотр каталога товаров», «Поиск украшений», «Просмотр информации о товаре», «Добавление товара в корзину», «Удаление товара из корзины», «Оформление заказа», «Просмотр статуса заказа», «Обновление информации о товаре» (только для администратора)», «Обработка запроса API», «Обработка запроса БД»).

Диаграмма вариантов использования представлена в приложении А.

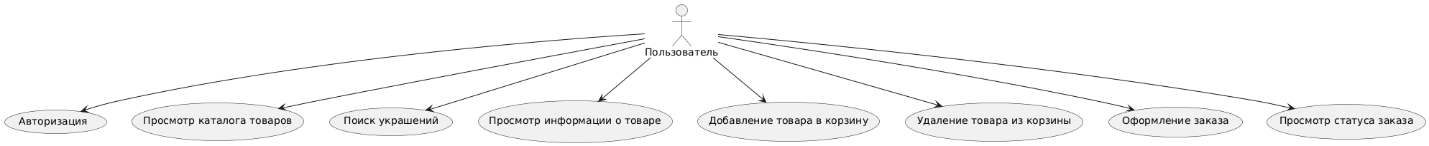
Рассмотрим ключевые моменты диаграммы:

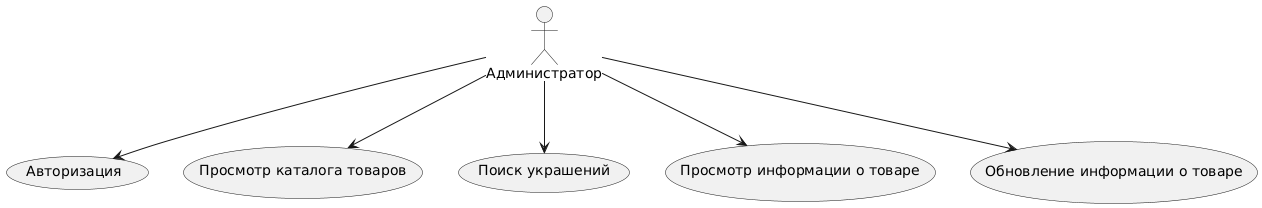
* Пользователь может выполнять прецеденты «Авторизация», «Просмотр каталога товаров», «Поиск украшений», «Просмотр информации о товаре», «Добавление товара в корзину», «Удаление товара из корзины», «Оформление заказа», «Просмотр статуса заказа».
* Администратор может выполнять прецеденты «Авторизация», «Обновление информации о товаре» и имеет доступ к прецедентам «Просмотр каталога товаров», «Поиск украшений», «Просмотр информации о товаре» для контроля информации о товарах.
* API обрабатывает запросы от актеров (например, проверка статуса заказа) и взаимодействует с базой данных.
* DB обрабатывает запросы от API, предоставляя доступ к данным, включая информацию о товарах, статусах заказов.

При взаимодействии с системой Администратор расширяет актера Пользователь и, таким образом, может также выполнять базовые прецеденты, доступные пользователю, такие как просмотр каталога товаров и поиск украшений.

Прецеденты «Добавление товара в корзину» и «Удаление товара из корзины» выполняются после прецедента «Авторизация», чтобы обеспечить безопасность данных пользователя.

Прецедент «Обработка запроса API» включает прецедент «Обработка запроса БД» для доступа к нужным данным и их обработки на стороне базы данных.

Рисунок 1 – use case пользователя

Рисунок 2 – use case администратора

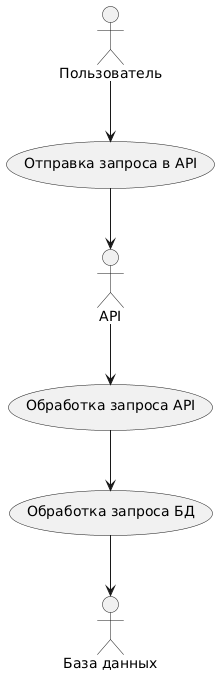


Рисунок 3 – use case «Процесс обработки пользовательских запросов»

**1.4 Проектирование структуры программного продукта**

С целью удостоверения предъявляемых к программе требований, приведенных в разделах 1.1-1.3, при реализации системы будет использоваться трехзвенная архитектура. Данная архитектура представляет собой организацию приложения на трех уровнях, каждый из которых выполняет определенные функции:

* представление – уровень отвечает за отображение пользовательского интерфейса и взаимодействие с пользователем. Он включает в себя компоненты, отвечающие за визуализацию данных и обработку пользовательских действий [10]. В этот уровень будет входить веб-приложение;
* логика приложения – на этом уровне происходит обработка бизнес-логики, связанной с функциональностью приложения для оформления заказов. Здесь располагаются компоненты, отвечающие за обработку запросов пользователей, проверку прав доступа, управление заказами, а также взаимодействие с базой данных. На этом уровне архитектуры будет находиться модуль обработки заказов, модуль аутентификации пользователей, модуль управления товарами;
* слой данных – уровень отвечает за хранение и доступ к данным. Здесь находится база данных, содержащая информацию, и компоненты, реализующие операции чтения и записи данных [10]. Этот слой будет представлять СУБД и компоненты, работающие с ней для получения и записи данных.

Диаграмма развертывания

Разрабатываемое веб-приложение будет состоять из следующих составляющих:

* страница аутентификации и авторизации – форма входа для аутентификации пользователей и администраторов, обеспечивающая контроль доступа к различным функциям приложения;
* главная страница – центральная страница, на которой пользователи могут просматривать доступные ювелирные изделия и переходить к оформлению заказа;
* страница для просмотра и управления заказами – позволяет пользователям просматривать историю своих заказов, статусы доставки и управлять текущими заказами (например, изменять статус или отменять заказ);
* страница управления товарами – обеспечивает интерфейс для добавления, редактирования и удаления ювелирных изделий, позволяющий администраторам обновлять информацию о наличии, ценах и описаниях товаров;
* страница управление пользователями – обеспечивает возможность добавления, редактирования и удаления учетных записей пользователей, а также управление их правами доступа.

Схема взаимодействия страниц

**1.5 Проектирование базы данных**

Для реализации полнофункционального приложения для оформления заказов в ювелирном магазине необходимо использовать базу данных.

В ходе исследования предметной области был выявлен набор сущностей, информацию о которых необходимо хранить в базе данных:

* Пользователь (id, username, password\_hash, emailAddress, phoneNumber, firstName, lastName, birthDate, permissionLvl, avatarUrl);
* Товар (id, name, description, price, brand, model, imageUrl, stockQuantity, categoryId);
* Заказ (id, userId, status, createdAt, updatedAt);
* Элемент заказа (id, orderId, productId, quantity, price);
* Категория (id, name, description);
* Корзина (id, userId);
* Элемент корзины (id, cartId, productId, quantity);
* Адрес (id, userId, country, city, street, homeNumber, flat);

Каждая таблица будет отвечать за определённый аспект функциональности приложения, позволяя эффективно управлять данными о пользователях, товарах и заказах.

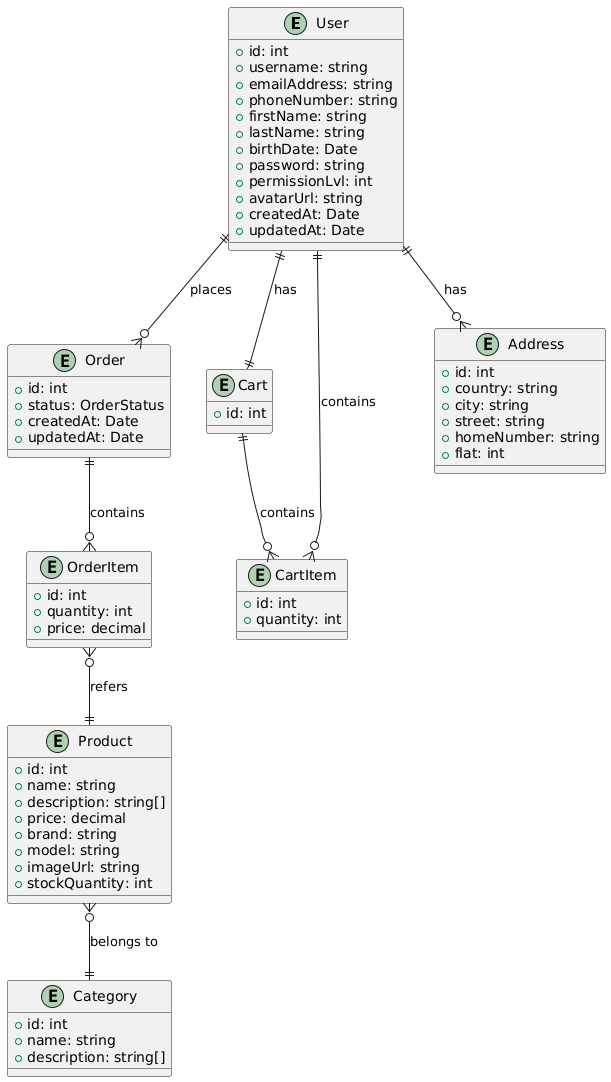


Рисунок 4 – Концептуальная модель БД

Словарь данных

**1.6 Проектирование пользовательского интерфейса**

Для реализации полнофункционального приложения для оформления заказов в ювелирном магазине необходимо использовать базу данных.