**Аналитическая записка**

по теме «Информационная система для магазина по продаже ПО»

1. **Наименование проекта**

Информационная система для учета радиодеталей

1. **Целевая аудитория**

Цель данной информационной системы состоит в создании удобного и эффективного инструмента для управления процессами в магазине, специализирующемся на продаже программного обеспечения. Система разработана для использования как администраторами, так и клиентами.

Она направлена на оптимизацию бизнес-процессов, повышение производительности сотрудников и улучшение качества обслуживания клиентов.

1. **Назначение проекта**

Современные магазины радиодеталей, работающие офлайн, сталкиваются с растущей потребностью в эффективном управлении и учете товаров. Разработанная программа играет ключевую роль в этом контексте, нацеленная на автоматизацию и оптимизацию внутренних процессов магазина.

Программа предоставляет кассирам возможность быстрого и точного управления корзиной товаров. Эффективное взаимодействие между кассирами и сотрудниками склада обеспечивается моментальной передачей сигнала о необходимости выдачи товара после оплаты.

Отслеживание товаров от момента поступления на склад до момента реализации осуществляется с акцентом на минимизацию времени и максимизацию точности учета. Программа также предоставляет возможность генерации разнообразных отчетов, что упрощает анализ статистики продаж и динамики спроса.

1. **Описание взаимодействия с потенциальным пользователем**

Все пользовательские истории представлены в Таблице 1.

Таблица 1 – Пользовательские истории

|  |  |  |  |
| --- | --- | --- | --- |
| № | Роль | Действие | Цель |
| 1 | Администратор | Создание каталога товара | Создание нового каталога товаров для увеличения линеек продукции |
| 2 | Администратор | Создание отчетов | Создание отчетов для просмотра финансовых движений, отслеживания проданных товаров и товаров на складе, просмотра кто продал товар и в каком объёме и т.д. |
| 3 | Администратор | Поиск товаров | Поиск товара для быстрого доступа к информации о товаре |
| 4 | Администратор | Проверка наличия товаров | Чтобы быстро определить находится товар на складе или его нет |
| 5 | Администратор | Вход в систему | Вход в систему для доступа к информационной системе и дальнейшей работе с ней |
| 6 | Администратор | Регистрация | Добавление новых сотрудников |
| 7 | Администратор | Обновление данных | Поддержание актуальности списка товаров |
| 8 | Администратор | Добавление данных | Добавление новых товаров для поддержания актуальности |
| 9 | Администратор | Удаление данных | Удаление товаров для поддержания актуальности |
| 10 | Сотрудник | Вход в систему | Вход в систему для доступа к информационной системе и дальнейшей работе с ней |
| 11 | Сотрудник | Обновление данных | Поддержание актуальности списка товаров |
| 12 | Сотрудник | Добавление данных | Добавление новых товаров для поддержания актуальности |
| 13 | Сотрудник | Удаление данных | Удаление товаров для поддержания актуальности |
| 14 | Сотрудник | Поиск товаров | Поиск товара для быстрого доступа к информации о товаре |
| 15 | Сотрудник | Проверка наличия товаров | Чтобы быстро определить находится товар на складе или его нет |

Все пользовательские сценарии представлены в таблице 2

Таблица 2 – Пользовательские

|  |  |  |
| --- | --- | --- |
| **№** | **Сценарий** | **Система** |
| 1 | Добавление данных | Приложение устанавливает соединение с базой данных и формирует SQL-запрос для добавления информации о товаре, такого как цена, описание или статус наличия. После отправки запроса база данных обновляет соответствующую запись. Затем приложение завершает соединение с базой данных и обновляет список товаров, отображаемый пользователю. |
| 2 | Удаления данных | Приложение устанавливает соединение с базой данных и формирует SQL-запрос для удаления информации о товаре, такого как цена, описание или статус наличия. После отправки запроса база данных обновляет соответствующую запись. Затем приложение завершает соединение с базой данных и обновляет список товаров, отображаемый пользователю. |
| 3 | Обновление данных | Приложение устанавливает соединение с базой данных и формирует SQL-запрос для обновления либо редактирования информации о товаре, такого как цена, описание или статус наличия. После отправки запроса база данных обновляет соответствующую запись. Затем приложение завершает соединение с базой данных и обновляет список товаров, отображаемый пользователю. |
| 4 | Вход в систему | Приложение устанавливает соединение с базой данных и формирует запрос для проверки учетных данных пользователя, таких как логин и пароль. После отправки запроса база данных сопоставляет введенные данные с сохраненными записями и возвращает результат. Если данные совпадают, пользователь получает доступ к системе. По завершении операции соединение с базой данных закрывается. |
| 5 | Проверка наличия товаров | Приложение устанавливает соединение с базой данных и формирует запрос для проверки количества товара на складе. После отправки запроса база данных сопоставляет наименование товара с со всем списком товаров. Если данные совпадают и количество больше нуля, то выводится наименование и количество товара на складе. По завершении операции соединение с базой данных закрывается. |
| 6 | Поиск товаров | Приложение предоставляет пользователю возможность найти товары по заданным критериям. Пользователь вводит параметры поиска, и приложение формирует запрос к базе данных. База данных выполняет поиск по соответствующим записям и возвращает список товаров, соответствующих введенным параметрам. Затем приложение отображает результаты пользователю, позволяя ему просмотреть доступные товары. |
| 7 | Создание отчётов | Приложение позволяет пользователю генерировать отчёты на основе данных из базы. Пользователь выбирает тип отчёта и параметры, на которых он будет основан. Приложение формирует запрос к базе данных для извлечения необходимой информации, а затем обрабатывает полученные данные и создает отчет в удобном формате. По завершении операции пользователю предоставляется доступ к сгенерированному отчёту. |
| 8 | Регистрация | Приложение предоставляет возможность регистрации нового пользователя. Пользователь вносит необходимые данные, такие как имя, номер телефона, адрес электронной почты и пароль. После этого приложение создает запрос для проверки уникальности введенных данных и записи новой учетной записи в базу данных. Если регистрация проходит успешно, приложение уведомляет пользователя о завершении процесса и предоставляет возможность войти в систему. |
| 9 | Создание каталога товаров | Приложение позволяет пользователю создавать каталог товаров, добавляя в него новые позиции. Пользователь вводит информацию о каждом товаре, включая название, описание, цену и другие характеристики. После этого приложение формирует запрос для сохранения данных в базе и добавляет запись о каждом товаре в каталог. По завершении операции пользователю предоставляется доступ к обновленному каталогу товаров. |

Диаграмма Use-Case представлена на рисунке 1.

**Изображение выглядит как текст, диаграмма, зарисовка, линия

Автоматически созданное описание**Диаграмма use-case для приложения описывает взаимодействие пользователей и системы в процессе использования ИС. Основные элементы диаграммы включают актеров (пользователей системы), варианты использования (функции системы) и связи между ними.

Рисунок 1 – Use case диаграмма

Описание диаграммы use-case:

|  |  |
| --- | --- |
| **Акторы** | **Краткое описание** |
| Администратор | Сотрудник, который имеет доступ ко всем функциям информационной системы. Имеет право добавлять, удалять и редактировать сотрудников, товары, каталоги и вообще всю информацию. Так же является ответственным за эту информацию, но не имеет право менять отчёты |
| Сотрудник | Сотрудник, который имеет ограниченное количество прав, которых хватит на добавление новых товаров (но не каталогов), удаление товаров (проще говоря продажа), поиск и просмотр наличия товаров |

1. **Описание функционала**

Все функциональные требования представлены в Таблице 3.

Таблица 3 - Функциональные требования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Название функции** | **Входные параметры** | **Выходные параметры** | **Описание функции** |
| 1 | Добавление данных | Наименование, фото, количество, цена (и другое в зависимости от данных) | Успешное добавление | Функция создаёт новую строчку в требуемом списке. |
| 2 | Удаление данных | ID (и другое в зависимости от данных) | Успешное удаление | Функция удаляет строчку в требуемом списке. |
| 3 | Обновление данных | ID (и другое в зависимости от данных) | Успешное обновление | Функция обновляет строчку в требуемом списке. |
| 4 | Вход в систему | Логин и пароль | Успешный вход в систему | Функция сравнивает вводные данные и данные в бд. |
| 5 | Проверка наличия товара | Наименование товара | Интересующий товар | Функция сравнивает вводные данные и данные в бд и выводит ответ |
| 6 | Поиск товара | Наименование товара либо ID товара | Интересующий товар | Функция сравнивает вводные данные и данные в бд и выводит ответ |
| 7 | Регистрация | Данные необходимые для регистрации (телефон, фио и др.) | Успешная регистрация | Функция добавляет новые данные о пользователе в бд |
| 8 | Создание отчёта | Данные о продажах, покупках, потерях и прочее | Успешно созданный отчет | Функция собирает необходимые данные и формирует отчёт |
| 9 | Создание каталога товаров | Наименование, описание и др. | Новый каталог | Функция собирает необходимые данные и формирует новый каталог |

1. **Возможные аналоги решения**

«PartsBox»

PartsBox — это приложение для учета компонентов и создания электронных проектов.

Ссылка для доступа к аналогу: <https://partsbox.com/>

Интерфейс сервиса представлен на рисунке 1.

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 2 - Partsbox

Пользователи могут легко управлять своим инвентарем, добавлять новые компоненты, создавать проекты с указанием используемых деталей. Поддерживает автоматическое получение данных о компонентах из различных источников, что упрощает процесс ввода информации.

Плюсы:

Имеет удобный интерфейс для управления компонентами и создания проектов.

Поддерживает сканирование штрихкодов для быстрого ввода данных.

Интегрируется с различными источниками данных для автоматического получения информации о компонентах.

Минусы:

Цена может быть довольно высокой, особенно для больших организаций.

Интерфейс может быть не настолько интуитивным для новых пользователей.

«Ciiva»

Ciiva — облачное приложение для управления компонентами и инвентарем. Оно предоставляет обширную базу данных компонентов с возможностью получения подробной информации, включая даташиты и схемы. Помимо учета, Ciiva также позволяет отслеживать заказы, что полезно при управлении производственными потоками.

Интерфейс сервиса представлен на рисунке 3.

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок 3 - Интерфейс сервиса “Ciiva”

Плюсы:

Облачное приложение, что обеспечивает доступность данных из любого места.

Предоставляет обширную информацию о компонентах из различных источников.

Имеет функционал для учета инвентаря и заказов.

Минусы:

Возможны сложности с пользовательским интерфейсом для некоторых пользователей.

В зависимости от объема данных могут возникнуть проблемы с производительностью.

«PartKeepr»

PartKeepr — это бесплатное и открытое программное обеспечение для учета компонентов. Он предоставляет простой, но функциональный интерфейс для добавления и управления компонентами. Пользователи могут категорировать компоненты, отслеживать количество на складе и следить за использованием в проектах.

Интерфейс сервиса представлен на рисунке 4.

Изображение выглядит как текст, снимок экрана, число, программное обеспечение

Автоматически созданное описание

Рисунок 4 – Интерфейс сервиса «PartKeepr»

Плюсы:

Бесплатное и открытое программное обеспечение, что может быть привлекательным для малых предприятий и любителей.

Простой интерфейс и легкость использования.

Поддерживает учет компонентов и инвентаря.

Минусы:

Меньше функциональности по сравнению с некоторыми коммерческими аналогами.

Поддержка может быть ограниченной, так как это проект с открытым исходным кодом.

1. **Предполагаемые к использованию технологии и модели**

Предлагаемое решение заключается в создании базы данных с использованием СУБД MySQL, которая будет эффективно хранить и управлять необходимыми данными. Данная база данных обеспечит надежное управление всеми записями, что значительно упростит работу сотрудников.

Для взаимодействия с базой данных будет разработано Backend-приложение на фреймворке FastAPI. Часть которой будет пользоваться администратор и сотрудник будет реализована на языке Python, что обеспечит высокую производительность и удобство в использовании. В рамках данного приложения администраторы, включая сотрудников, ответственных за управление данными, смогут легко добавлять, редактировать и удалять информацию о категориях, товарах и продажах. Кроме того, будет доступна функция формирования отчетов о продажах, что упростит аналитическую деятельность.

Программа предоставит возможность удобного управления информацией и выполнения всех необходимых операций без необходимости прибегать к сторонним инструментам для обработки данных. Это значительно повысит эффективность работы сотрудников и сократит время на выполнение рутинных задач.