**Аналитическая записка**

по теме «Информационная система для кофейни»

1. **Наименование проекта**

Информационная система для кофейни.

1. **Целевая аудитория**

Данной информационной системой могут пользоваться администраторы, сотрудники (баристы) и бухгалтер для контроля финансов и оптимизации процессов. Администраторы кофейни применяют её для учета запасов товаров, продаж и управлением информацией о сотрудниках. Бухгалтеры используют для ведения отчетности. Сотрудники применяют для осуществления продаж.

1. **Назначение проекта**

Информационная система призвана автоматизировать и оптимизировать операции, связанные с управлением кофейни. Она обеспечивает удобное управление товарными запасами, позволяя отслеживать наличие ингредиентов в реальном времени, генерирует финансовые отчеты и помогает отслеживать доходы и расходы. Главной целью системы является облегчение и автоматизация процессов внутри кофейни, что позволит более точно учитывать данные о продажах, а также повысит качество обслуживания клиентов.

1. **Описание взаимодействия с потенциальным пользователем**

Все пользовательские истории представлены в Таблице 1.

Таблица 1 – Пользовательские истории

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Роль** | **Действие** | **Цель** |
| 1 | Администратор | Добавление товара | Обновление списка товаров, поддержание актуальности списка имеющихся товаров. |
| 2 | Администратор | Добавление сотрудника | Обновление списка сотрудников. |
| 3 | Администратор | Добавление категорий | Обновление списка категорий, поддержание актуальности списка имеющихся категорий. |
| 4 | Администратор | Редактирование данных | Обновление информации для обеспечения актуальности данных. |
| 5 | Администратор | Удаление данных | Устранение устаревших или неверных данных для поддержания чистоты базы данных. |
| 6 | Администратор | Просмотр данных | Доступ к информации для анализа |
| 7 | Администратор | Формирование отчёта о продажах | Формирование отчётности по продажам за указанный период. |
| 8 | Бариста | Добавление продажи | Обновление списка продаж, поддержание актуальности списка продаж. |
| 9 | Бариста | Редактирование данных о продажах | Обновление информации о продажах. |
| 10 | Бухгалтер | Просмотр данных | Доступ к информации для анализа финансовой деятельности. |
| 11 | Бухгалтер | Формирование отчёта о продажах | Формирование отчётности по продажам для анализа доходов и расходов. |

Все пользовательские сценарии представлены в Таблице 2.

Таблица 2 – Пользовательские сценарии

|  |  |  |
| --- | --- | --- |
| **№** | **Пользователь** | **Система** |
| 1 | Добавление товара | Приложение открывает соединение с базой данных, создает запрос на добавление товара в базу данных, используя язык запросов SQL, отправляет запрос на добавление товара в базу данных, база данных выполняет запрос и добавляют новую запись в таблицу. База данных обновляет общий список товаров, чтобы отражать добавленный товар, обновляет список товаров, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 2 | Редактирование товара | Приложение открывает соединение с базой данных, получает идентификатор товара для редактирования, создает запрос на редактирование товара в базе данных, используя язык запросов SQL, отправляет запрос на редактирование товара в базу данных, база данных выполняет запрос и обновляет соответствующую запись в таблице. База данных обновляет общий список товаров, чтобы отражать изменения в товаре, обновляет список товаров, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 3 | Просмотр списка товаров | Приложение открывает соединение с базой данных, создает запрос на получение списка товаров, используя язык запросов SQL, отправляет запрос на получение списка товаров в базу данных, база данных выполняет запрос и возвращает список товаров. Приложение обрабатывает и форматирует список товаров, отображает список товаров пользователю. Приложение закрывает соединение с базой данных. |
| 4 | Удаление товара | Приложение открывает соединение с базой данных, создает запрос на удаление товара, используя язык запросов SQL, отправляет запрос на удаление товара в базу данных, база данных выполняет запрос и удаляет товар. База данных обновляет общий список товаров, обновляет список товаров, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 5 | Добавление продажи | Приложение открывает соединение с базой данных, создает запрос на добавление записи о продаже в базу данных, используя язык запросов SQL, отправляет запрос на добавление записи о продаже в базу данных, база данных выполняет запрос и добавляют новую запись в таблицу. База данных обновляет общий список продаж, чтобы отражать добавленную продажу, обновляет список продаж, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 6 | Редактирование продажи | Приложение открывает соединение с базой данных, получает идентификатор продажи для редактирования, создает запрос на редактирование продажи в базе данных, используя язык запросов SQL, отправляет запрос на редактирование продажи в базу данных, база данных выполняет запрос и обновляет соответствующую запись в таблице. База данных обновляет общий список продаж, чтобы отражать изменения в продаже, обновляет список продаж, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 7 | Просмотр списка продаж | Приложение открывает соединение с базой данных, создает запрос на получение списка продаж, используя язык запросов SQL, отправляет запрос на получение списка продаж в базу данных, база данных выполняет запрос и возвращает список продаж. Приложение обрабатывает и форматирует список продаж, отображает список продаж пользователю. Приложение закрывает соединение с базой данных. |
| 8 | Удаление продажи | Приложение открывает соединение с базой данных, создает запрос на удаление продажи, используя язык запросов SQL, отправляет запрос на удаление продажи в базу данных, база данных выполняет запрос и удаляет продажу. База данных обновляет общий список продаж, обновляет список продаж, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 9 | Добавление сотрудника | Приложение открывает соединение с базой данных, создает запрос на добавление сотрудника в базу данных, используя язык запросов SQL, отправляет запрос на добавление сотрудника в базу данных, база данных выполняет запрос и добавляют новую запись в таблицу. База данных обновляет общий список сотрудников, чтобы отражать добавленного сотрудника, обновляет список сотрудников, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 10 | Редактирование сотрудника | Приложение открывает соединение с базой данных, получает идентификатор сотрудника для редактирования, создает запрос на редактирование сотрудника в базе данных, используя язык запросов SQL, отправляет запрос на редактирование сотрудника в базу данных, база данных выполняет запрос и обновляет соответствующую запись в таблице. База данных обновляет общий список сотрудников, чтобы отражать изменения в сотруднике, обновляет список сотрудников, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 11 | Просмотр списка сотрудников | Приложение открывает соединение с базой данных, создает запрос на получение списка сотрудников, используя язык запросов SQL, отправляет запрос на получение списка сотрудников в базу данных, база данных выполняет запрос и возвращает список сотрудников. Приложение обрабатывает и форматирует список сотрудников, отображает список сотрудников пользователю. Приложение закрывает соединение с базой данных. |
| 12 | Удаление сотрудника | Приложение открывает соединение с базой данных, создает запрос на удаление сотрудника, используя язык запросов SQL, отправляет запрос на удаление сотрудника в базу данных, база данных выполняет запрос и удаляет сотрудника. База данных обновляет общий список сотрудников, обновляет список сотрудников, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 13 | Добавление категории | Приложение открывает соединение с базой данных, создает запрос на добавление категории в базу данных, используя язык запросов SQL, отправляет запрос на добавление категории в базу данных, база данных выполняет запрос и добавляют новую запись в таблицу. База данных обновляет общий список категорий, чтобы отражать добавленную категорию, обновляет список категорий, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 14 | Просмотр списка категорий | Приложение открывает соединение с базой данных, создает запрос на получение списка категорий, используя язык запросов SQL, отправляет запрос на получение списка категорий в базу данных, база данных выполняет запрос и возвращает список категорий. Приложение обрабатывает и форматирует список категорий, отображает список категорий пользователю. Приложение закрывает соединение с базой данных. |
| 15 | Удаление категории | Приложение открывает соединение с базой данных, создает запрос на удаление категории, используя язык запросов SQL, отправляет запрос на удаление категории в базу данных, база данных выполняет запрос и удаляет категорию. База данных обновляет общий список категорий, обновляет список категорий, доступный для пользователей. Приложение закрывает соединение с базой данных. |
| 16 | Создание отчёта по продажам | Приложение открывает соединение с базой данных, создает запрос на получение данных о продажах за указанный пользователем период, используя язык запросов SQL. Отправляет запрос на получение данных о продажах в базу данных, база данных выполняет запрос и возвращает необходимые данные о продажах. Приложение обрабатывает и форматирует полученные данные о продажах, группируя их по необходимым параметрам, таким как дата. Затем приложение генерирует отчет по продажам в виде таблицы. Приложение отображает отчет по продажам пользователю. Приложение закрывает соединение с базой данных. |

Диаграмма Use-Case представлена на рисунке 1.

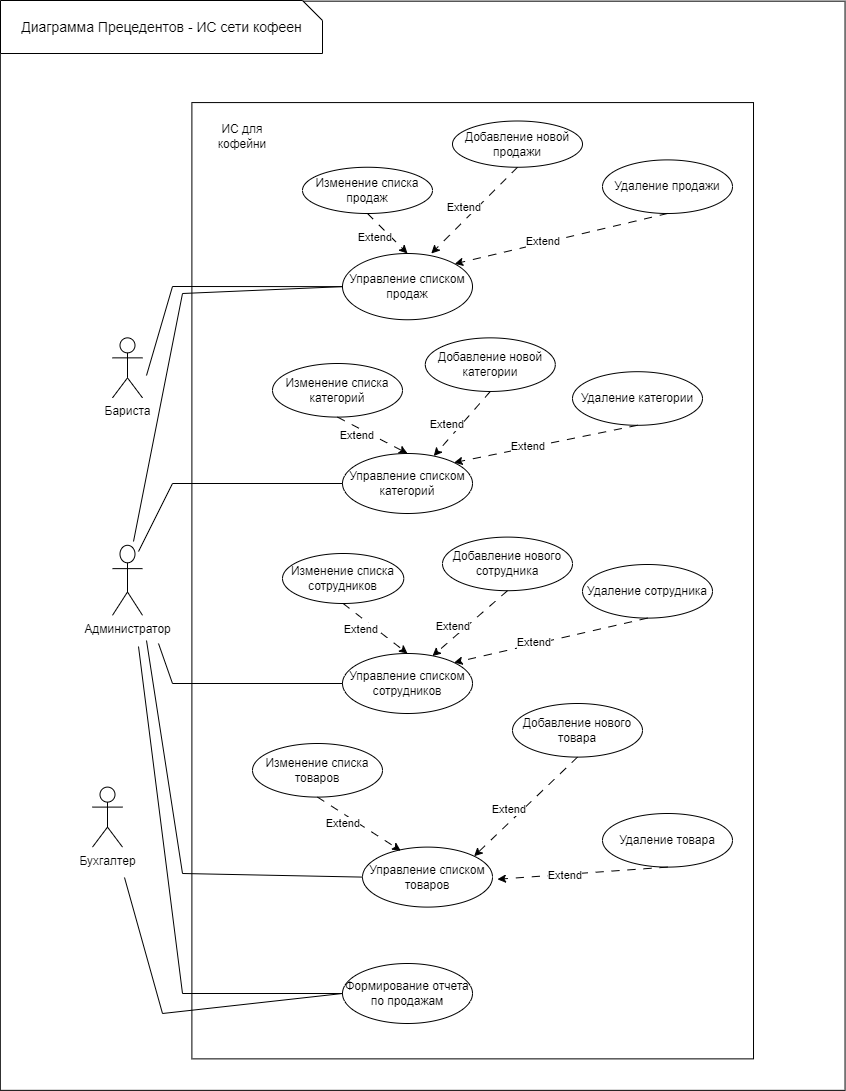


Рисунок 1 – Use case диаграмма

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

Описание диаграммы use-case:

Таблица 3 – Описание акторов

|  |  |
| --- | --- |
| **Акторы** | **Краткое описание** |
| Менеджер | Ответственное лицо, которое управляет всеми аспектами системы: добавляет, изменяет и удаляет любые данные. |
| Бариста | Сотрудник, который осуществляет продажи, добавляет и редактирует данные о товарах, категориях и продажах, но не имеет доступа к управлению сотрудниками и отчетам о продажах. |
| Бухгалтер | Сотрудник, который просматривает данные о продажах и формирует финансовую отчётность. |

Таблица 4 – Описание прецедентов

|  |  |
| --- | --- |
| **Прецеденты** | **Описание** |
| Управление списком категорий | Запускается непосредственно администратором и баристой. Позволяет дополнять существующий список категорий новыми записями, редактировать определенную категорию из списка, удалять выбранную категорию. |
| Управление списком товаров | Запускается непосредственно администратором и баристой. Позволяет дополнять существующий список товаров новыми записями, редактировать определенный товар из списка, удалять выбранный товар. |
| Управление списком сотрудников | Запускается непосредственно администратором. Позволяет дополнять существующий список сотрудников новыми записями, редактировать определенного сотрудника из списка, удалять выбранного сотрудника. |
| Управление списком продаж | Запускается непосредственно администратором. Позволяет дополнять существующий список продаж новыми записями, редактировать определенную продажу из списка, удалять выбранную продажу. |
| Формирование отчета по продажам | Запускается администратором и бухгалтером. Позволяет сформировать отчет по продажам за определенный период по запросу пользователя. |

1. **Описание функционала**

Все функциональные требования представлены в Таблице 3.

Таблица 5 - Функциональные требования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Название функции** | **Входные параметры** | **Выходные параметры** | **Описание функции** |
| 1 | Добавление товара | Код, наименование, категория, цена, количество. | Новая строка в списке товаров. | Функция сохраняет информацию о товаре в базу данных, проверяет уникальность кода товара, создаёт новую запись в таблице товаров и обновляет индекс для быстрого поиска в будущем. |
| 2 | Редактирование товара | Код, наименование, категория, цена, количество. | Обновленная информация о товаре. | Функция находит запись о товаре по его коду, проверяет возможность изменения (например, отсутствие зависимых процессов), обновляет поля и вносит изменения в базу данных. |
| 3 | Удаление товара | Код товара. | Успешное удаление. | Функция находит запись по коду товара, проверяет наличие зависимых данных (например, активных продаж или резервов), удаляет товар и очищает связанные данные для поддержания целостности. |
| 4 | Добавление продажи | Номер продажи, сумма продажи, дата, код, количество, ID\_Сотрудника. | Новая строка в списке продаж. | Функция проверяет наличие товара на складе, фиксирует продажу, уменьшает количество доступных товаров, создаёт новую запись в таблице продаж и привязывает её к сотруднику. |
| 5 | Редактирование продажи | Номер продажи, сумма продажи, дата, код, количество, ID\_Сотрудника. | Обновленная информация о продаже. | Функция находит запись о продаже, проверяет, можно ли внести изменения (например, по срокам или статусу), обновляет данные о сумме, товаре и сотруднике, корректирует складские запасы. |
| 6 | Удаление продажи | Номер продажи. | Успешное удаление. | Функция проверяет, завершена ли продажа (доставлена или оплачена), удаляет запись о продаже, возвращает товары на склад (если применимо) и корректирует финансовые данные. |
| 7 | Добавление сотрудника | ID, ФИО, контактные данные, адрес проживания. | Новая строка в списке сотрудников. | Функция сохраняет информацию о сотруднике, проверяет уникальность ID, добавляет данные в таблицу сотрудников, создаёт учётную запись для доступа к системе. |
| 8 | Редактирование сотрудника | ID, ФИО, контактные данные, адрес проживания. | Обновленная информация о сотруднике. | Функция находит сотрудника по ID, проверяет его активность (например, наличие текущих задач), вносит изменения и синхронизирует их с системой безопасности для актуализации прав доступа. |
| 9 | Удаление сотрудника | ID сотрудника. | Успешное удаление. | Функция проверяет, завершены ли все задачи сотрудника (продажи, отчёты), удаляет запись и отключает доступ к системе, аннулируя связанные данные. |
| 10 | Добавление категории | Код, наименование. | Новая строка в списке категорий. | Функция создаёт новую категорию в таблице категорий, проверяет уникальность наименования и кода, и обновляет систему классификации для связи с товарами. |
| 11 | Удаление категории | Код, наименование. | Успешное удаление. | Функция проверяет наличие товаров в категории, если категория пустая, удаляет её из базы данных и обновляет классификацию для исключения из связанных процессов. |
| 12 | Формирование отчёта | Параметры запроса пользователя: период. | Сформированный отчёт с данными по запросу. | Функция анализирует введённые параметры, обращается к соответствующим таблицам в базе данных, формирует отчёт, который может включать графики, таблицы, и экспортирует в нужный формат. |

1. **Возможные аналоги решения**

Для определения проблематики рассмотрим существующий аналог - КонтурМаркет - https://kontur.ru/market/obshepit.

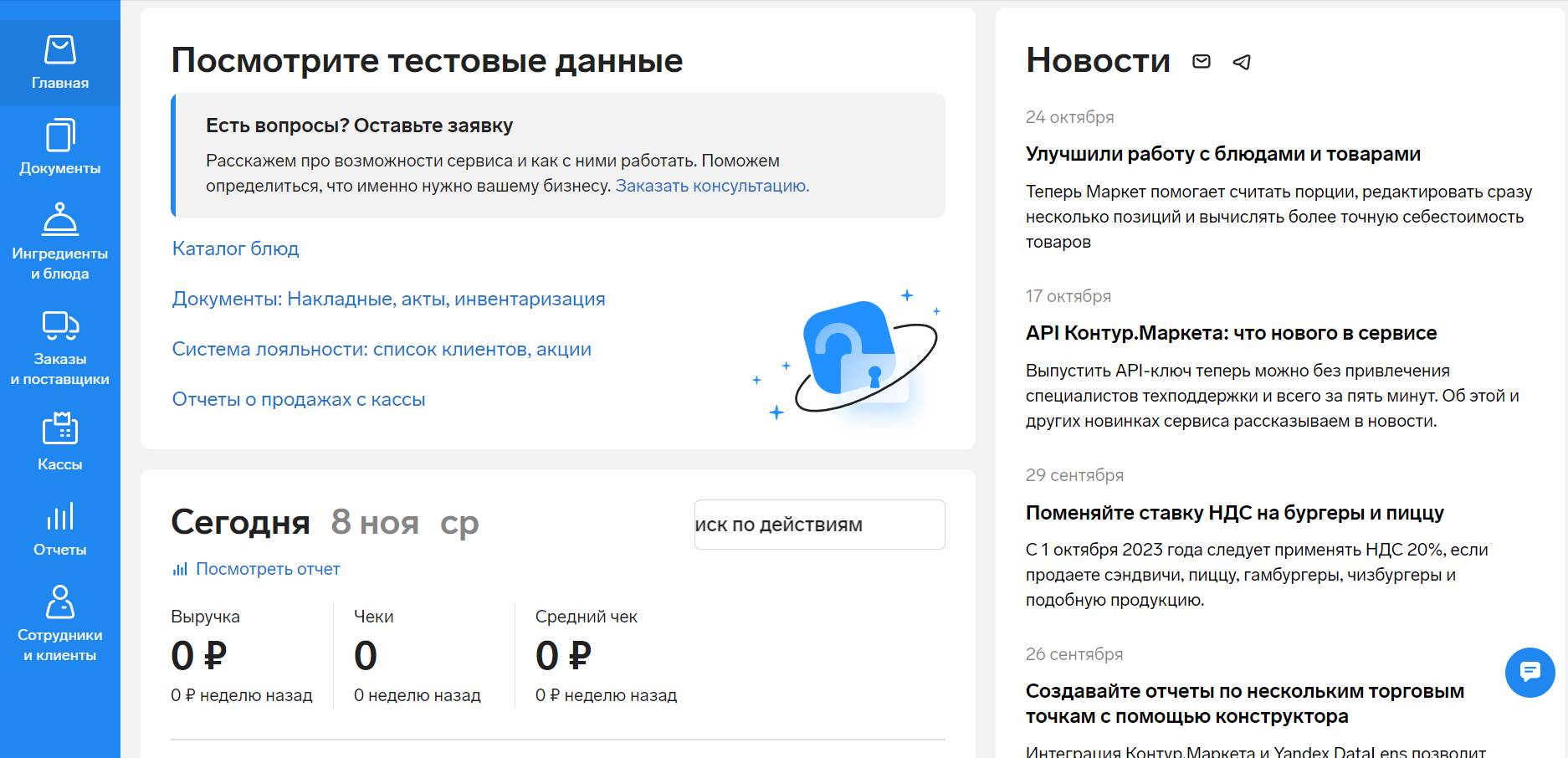


Рисунок 2 – Интерфейс КонтурМаркета

Данный аналог представляет собой систему для учета в заведении.

К преимуществам можно отнести:

* Управление продукцией, заказами, категориями и отслеживание остатков.
* Интеграция с онлайн-магазинами и платежными системами.

К недостаткам можно отнести:

* Некоторые функции могут быть сложны в использовании для новых пользователей.
* Система не подстроена конкретно под кофейню.

Рассмотрев аналог можно сделать вывод, что разрабатываемая информационная система должна быть точной, без ошибок и хранить в себе большой объем данных, подстроена под кофейню и обладать простым и понятным пользователю интерфейсом.

Тогда целью производственной практики является проектирование и разработка информационной системы, предназначенной для хранения большого объема записей в базе данных и создание графического интерфейса для упрощения процесса заполнения БД, которая хранит информацию для учета товаров, сотрудников и продаж.

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

Предлагаемое решение – создание базы данных при помощи СУБД SQLite, которая будет хранить данные о плакате. Эта база данных обеспечит эффективное управление всеми данными. Для работы с базой данных будет разработано приложение на языке Python с использованием библиотеки PyQt 5. В рамках приложения пользователи смогут добавлять и редактировать, удалять данные о категориях, товарах, сотрудниках и продажах, а также формировать отчет о продажах. Интерфейс будет интуитивно понятным и адаптированным. Программа обеспечит возможность удобного управления информацией и выполнения всех необходимых операций без необходимости использования внешних инструментов для обработки данных.