**АНАЛИТИЧЕСКАЯ ЗАПИСКА**

**«Информационная система для кинотеатра»**

1. **Наименование проекта**

Информационная система для кинотеатра.

1. **Целевая аудитория**

Основная целевая аудитория проекта:

* индивидуальные предприниматели;
* юридические лица;
* сотрудники кинотеатра: директор, администратор, менеджер;
* будущие клиенты.

1. **Назначение проекта**

Назначение разработки состоит в создании и внедрении информационной системы, предназначенной для автоматизации управления покупки билета на фильм. Система позволит автоматизировать работу кинотеатра, автоматизировать процессы учета продажи билетов.

1. **Описание взаимодействия с потенциальным пользователем**

Пользовательские истории представлены в Таблице 1.

Таблица 1 – Пользовательские истории

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Роль** | **Действие** | **Цель** |
|  |  |  |  |
| 1 | Администратор кинотеатра | Внесение изменений в БД | Обновление списка сеансов для поддержания его актуальности |
| 2 | Администратор кинотеатра | Изменение списка фильмов | Обновление списка сотрудников для поддержания актуальной информации о фильмах. |
| 3 | Администратор кинотеатра | Удаление фильмов | Поддержание актуальности списка категорий, удаление устаревших фильмов. |
| 4 | Администратор кинотеатра | Добавление фильмов | Обеспечение актуальности данных путем добавления новых фильмов. |
| 5 | Администратор  кинотеатра | Редактирование фильмов | Устранение устаревших или неверных данных для поддержания чистоты базы данных. |
| 6 | Менеджер | Просмотр финансовой отчетности | Анализ финансовой деятельности на основе отчетности о продаже билетов. |
| 7 | Клиент | Поиск фильма | Поиск фильма для формирования отчетности по продажам за определенный период. |
| 8 | Клиент | Изменение даты сеанса фильма | Обновление содержимого корзины для поддержания актуальности информации о фильмах. |
| 9 | Клиент | Удаление фильма из корзины | Поддержание актуальности информации о сеансах и категориях при изменении даты. |
| 10 | Клиент | Добавление фильмов в корзину | Обновление информации о добавленных фильмах и датах. |
| 11 | Клиент | Бронирование места | Бронирование места для упорядочивания данных в системе. |
| 12 | Клиент | Оплата данных | Анализ финансовой деятельности через оплату. |
| 13 | Клиент | Просмотр истории покупки билетов | Анализ доходов и расходов с помощью просмотра истории покупки билетов. |
| 14 | Клиент | Вход в систему | Авторизация в системе для доступа к акциям и уведомлениям о новых фильмах. |

Все пользовательские сценарии представлены в Таблице 2.

Таблица 2 – Пользовательские сценарии

|  |  |  |
| --- | --- | --- |
| **№** | **Пользователь** | **Система** |
| 1 | Внесение изменений в БД | Приложение открывает соединение с базой данных, создает запрос на обновление или добавление данных в таблицы, используя язык SQL. После выполнения запроса база данных обновляет соответствующие записи. После завершения операции приложение закрывает соединение с базой данных. |
| 2 | Изменение списка фильмов | Приложение открывает соединение с базой данных, формирует SQL-запрос на обновление данных о товаре (например, изменение цены, описания или статуса показа). Отправляет запрос, и база данных обновляет соответствующую запись. Приложение закрывает соединение с базой данных и обновляет отображаемый список фильмов для пользователя. |
| 3 | Удаление фильма | Приложение открывает соединение с базой данных, создает SQL-запрос на удаление фильма из таблицы фильмов, отправляет запрос. База данных выполняет запрос, удаляя запись о фильме. После выполнения операции приложение закрывает соединение с базой данных и обновляет список доступных фильмов для пользователей. |
| 4 | Добавление фильма | Приложение открывает соединение с базой данных, создает запрос на добавление нового фильма в базу данных с использованием SQL, отправляет запрос. База данных выполняет запрос, добавляя новую запись в таблицу. После выполнения операции приложение закрывает соединение с базой данных и обновляет отображаемый список фильмов. |
| 5 | Редактирование фильма | Приложение открывает соединение с базой данных, формирует запрос на изменение данных фильма (например, изменение цены или описания), используя SQL. Отправляет запрос, и база данных вносит изменения в соответствующую запись. Приложение закрывает соединение с базой данных и обновляет информацию о фильме, отображаемую пользователю. |
| 6 | Просмотр финансовой отчетности | Приложение открывает соединение с базой данных, создает запрос на выбор данных из таблиц с финансовой отчетностью, используя SQL, отправляет запрос. База данных возвращает отчетные данные. Приложение обрабатывает данные и отображает их пользователю в удобном формате. После этого соединение с базой данных закрывается. |
| 7 | Поиск фильма | Приложение открывает соединение с базой данных, создает SQL-запрос для поиска фильма по заданным критериям (например, по названию или дате показа), отправляет запрос. База данных возвращает результаты поиска, которые приложение отображает пользователю. После завершения операции соединение с базой данных закрывается. |
| 8 | Изменение даты сеанса фильма | Приложение открывает соединение с базой данных, создает SQL-запрос для изменения даты сеанса фильма (например, изменение количества билетов или удаление фильма), отправляет запрос. База данных обновляет запись даты для пользователя. После завершения операции приложение закрывает соединение с базой данных. |
| 9 | Удаление фильма из корзины | Приложение открывает соединение с базой данных, создает запрос на удаление фильма из таблицы дата пользователя, используя SQL, и отправляет его. База данных выполняет запрос и удаляет соответствующие записи. Приложение закрывает соединение с базой данных и обновляет отображение корзины. |
| 10 | Добавление фильма в корзину | Приложение открывает соединение с базой данных, формирует SQL-запрос на добавление выбранного фильма в корзину пользователя, отправляет запрос. База данных добавляет запись о товаре в таблицу корзины. После выполнения операции приложение закрывает соединение с базой данных и обновляет отображение содержимого корзины. |
| 11 | Бронирование места | Приложение открывает соединение с базой данных, создает запрос на добавление бронированного места и на основе содержимого корзины пользователя, отправляет запрос. База данных создает запись о бронировании места, фиксируя детали фильма и данные пользователя. Приложение закрывает соединение с базой данных и отображает информацию о созданном билете. |
| 12 | Оплата данных | Приложение открывает соединение с базой данных, формирует запрос на обновление статуса оплаты билета, отправляет запрос с информацией об успешной транзакции. База данных обновляет запись о билете, устанавливая статус «оплачено». Приложение закрывает соединение с базой данных и подтверждает пользователю успешное выполнение оплаты. |
| 13 | Просмотр истории покупки билетов | Приложение открывает соединение с базой данных, создает SQL-запрос для выборки истории покупки билетов конкретного пользователя, отправляет запрос. База данных возвращает информацию о прошлых покупках билетов, которую приложение отображает в виде списка или таблицы. После завершения операции соединение с базой данных закрывается. |
| 14 | Вход в систему | Приложение открывает соединение с базой данных, создает запрос на проверку учетных данных пользователя (например, логина и пароля), отправляет запрос. База данных сверяет введенные данные с сохраненными записями и возвращает результат. Если данные совпадают, приложение позволяет пользователю войти в систему. После выполнения операции соединение с базой данных закрывается. |

Диаграмма Use-Case представлена на рисунке 1.

****

Рисунок 1 – Диаграмма Use-Case

Диаграмма Use-Case для информационной системы кинотеатра отображает взаимодействие между системой и тремя основными категориями пользователей: Администратор, Менеджер и Пользователь. Каждый из акторов имеет доступ к различным функциям системы, в зависимости от своей роли.

Основные участники (акторы):

* Директор кинотеатра – отвечает за управление кинотеатром, контролирует работу всей информационной системы в целом, просматривает сводные отчеты, отвечает за добавление новых сотрудников, назначение прав доступа.
* Администратор – отвечает за добавление новых сотрудников, назначение прав доступа, добавляет информацию о фильмах.
* Менеджер – занимается просмотром финансовой отчетности на основе продажи билетов.
* Клиент – имеет возможность просматривать сеансы на определенную дату, выбрать фильм и произвести покупку билета.

Основные варианты использования Use-Case:

Просмотр сводных отчетов – директор анализирует сводные данные о продаже билетов, посещениях клиентов.

Добавление нового сотрудника – администратор и директор регистрируют новых сотрудников в системе.

Назначение прав доступа – администратор и директор и назначают права доступа сотрудникам в информационной системе.

Добавление информации о фильмах – администратор вносит в информационную систему данные о фильмах.

Просмотр сеансов – клиент просматривает сеансы фильмов на определенную дату.

Просмотр фильмов – клиент смотрит на какой фильм хочет сходить, в выбранную дату сеанса.

Покупка билета – клиент покупает билет на выбранную дату и фильм.

Акторы связаны с соответствующими случаями использования посредством линий, что отражает их взаимодействие с системой.

Диаграмма иллюстрирует, какие роли играют акторы, как они взаимодействуют с системой, и каковы ключевые функции, выполняемые системой для обеспечения эффективной работы кинотеатра.

1. **Основной функционал (функциональные требования)**

Функциональные требования представлены в таблице 3.

Таблица 3 – Функциональные требования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Название функции** | **Входные параметры** | **Выходные параметры** | **Описание функции** |
| 1 | Вывод сводных отчетов | Данные из базы данных, | Вывод отчетности в формате «.pdf» | Функция предназначена для сохранения, вывода следующих отчетов в информационную систему: отчет о поступлении клиентах |
| 2 | Добавление нового пользователя в систему | Для клиента: почта, пароль, для сотрудника: почта, пароль, паспорт, трудовая книжка, СНИЛС, документы воинского учёта, ФИО | Запись в базе данных в таблице «Пользователь» | Функция формирует запись в базе данных с данными клиента, сотрудника, определяет роль сотрудника, ограничивает доступ в соответствии с ролью |
| 3 | Формирование отчетности | Необходимые данные для отчетности из базы данных, шаблоны в формате «.pdf» | Сформированный, сгенерированный отчет | Функция генерирует отчет, позволяет сохранить отчет в необходимую директорию |
| 4 | Внесение информации о фильме | Данные о фильме | Запись в базе данных в таблице «Фильм» | Функция формирует запись в базе данных о фильме |

1. **Возможные аналоги решения**

**Vista Cinema –** Комплексное решение для управления кинотеатрами, включающее продажу билетов, управление сеансами, маркетинг и аналитику.



Рисунок 2 – Страница Vista Cinema

**Преимущества:**

Поддержка всех аспектов управления кинотеатром, включая продажу билетов, маркетинг и аналитику. Мощные инструменты для анализа данных о продажах и посещаемости. Возможность создания и управления программами лояльности.

**Недостатки:**

Высокая стоимость лицензии и внедрения. Требует обучения персонала для эффективного использования.

**Comscore –** Решения для анализа и управления кинопрокатом, оптимизации расписания и улучшения посещаемости.

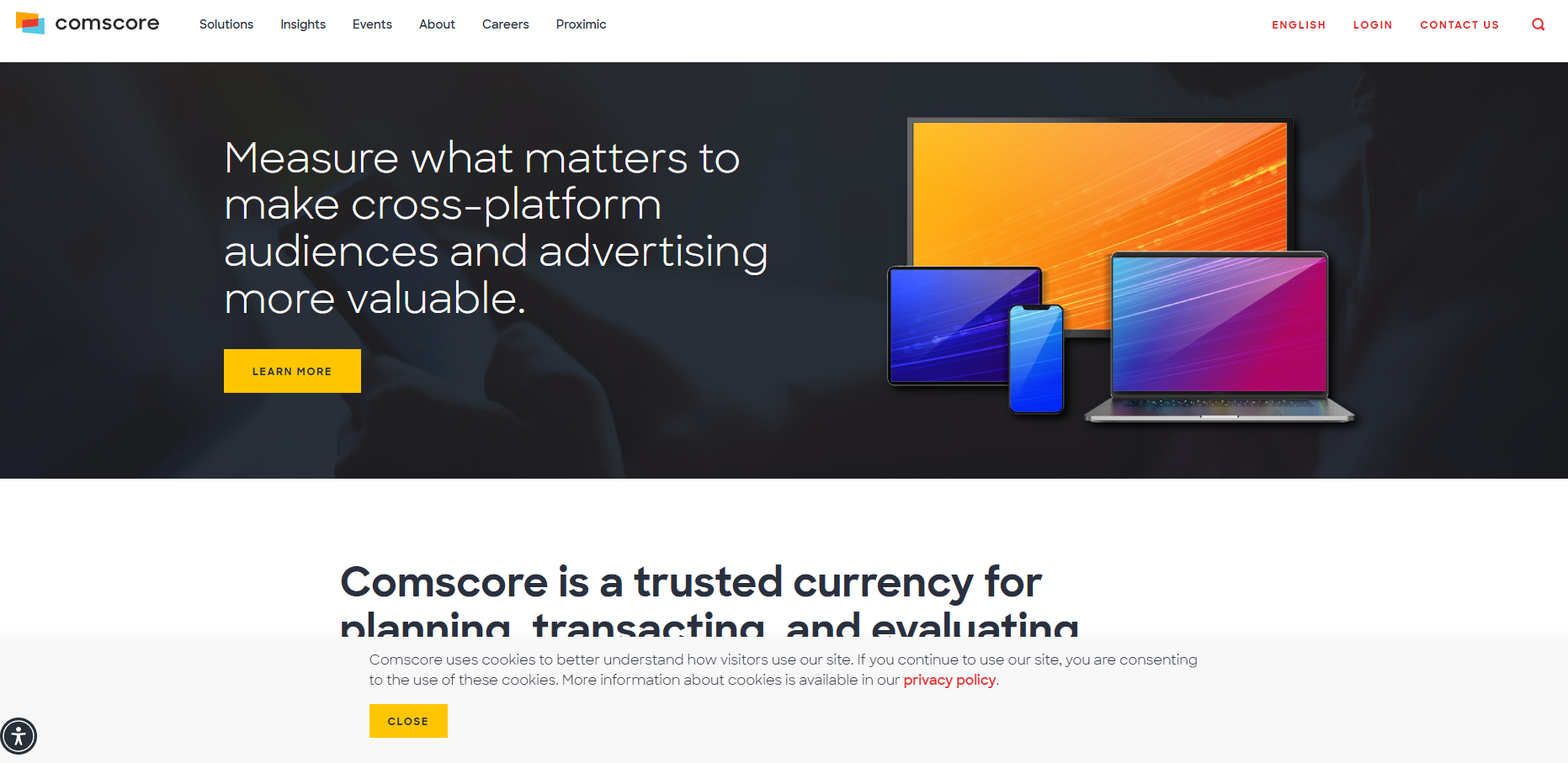


Рисунок 2 – Страница Comscore

**Преимущества:**

Подробные отчеты и визуализации данных. Совместимость с различными системами управления кинотеатрами.

**Недостатки:**

Меньше функций для непосредственного управления продажами и сеансами. Высокая стоимость услуг аналитики.

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

Для создания информационной системы для кинотеатра можно использовать различные технологии и модели.

Пример технологий для базы данных: для хранения структурированных данных можно использовать реляционные базы данных, например, MySQL, PostgreSQL, для хранения неструктурированных данных, например, фотографий можно использовать MongoDB, CouchDB.

Пример моделей для базы данных: для представления структуры базы данных можно использовать ER-моделирование, ERD-диаграммы, JSON файлы.

Пример технологий для создания веб интерфейса: для создания пользовательского интерфейса можно использовать React, Angular, для обработки запросов и взаимодействия с базой данных можно использовать Node.js, Django, Flask.

Пример моделей для создания веб интерфейса: MVC моделирование.

Пример технологий для написания мобильного приложения: для использования всех возможностей устройства можно использовать Kotlin, Flutter, React Native.

Пример моделей для написания мобильного приложения: MVVM моделирование.

Пример технологий для проектирования интерфейса пользователя: для проектирования интерфейса приложения, сайта, можно использовать Figma, draw.io.

Модели для проектирования интерфейса пользователя: для определения основных задач и потребностей пользователя можно использовать пользовательские сценарии, для проверки удобства и функциональности интерфейса, прототипирование и тестирование.

Данные технологии и модели помогут создать эффективную и функциональную информационную систему. Выбор конкретных инструментов и подходов зависит от специфики задач.