Белорусский государственный университет информатики и радиоэлектроники

Лабораторная работа №1

Техническое задание к программному продукту

«Аренда торговых площадей»

Выполнил: Ярош Г.И..

ст. гр. 353501

Проверила: Саттарова П. С.

ассистент кафедры информатики

Минск, 2016

**Содержание**

1. Общие положения
   1. Краткое описание предметной области
   2. Целевая аудитория
   3. Предназначение системы
2. Информационные сущности
3. Функциональные требования
   1. Возможности гостя
   2. Возможности пользователя
   3. Возможности администратора
   4. Хранимые процедуры
4. Требования к безопасности
5. **Общие положения**
   1. Краткое описание предметной области

Целью работы является организация структуры базы данных и работы для оптимизации функционирования кинотеатра.

В кинотеатре работает обслуживающий персонал. К сервису обращаются люди, желающие забронировать билет на киносеанс. Все они проходят обязательную регистрацию, при которой в базу данных заносятся стандартные анкетные данные (фамилия, имя, номер телефона).

Каждый пользователь может заказывать несколько билетов, как на один сеанс, так и на несколько различных. Все брони билетов фиксируются при этом изменяется плотность заполнения зала на определенный сеанс.

По наступлении времени окончания сеанса данные в связанных таблицах данных обновляются.

* 1. Целевая аудитория

Физические лица, посетители кинотеатра.

* 1. Предназначение системы

Автоматизация работы кинотеатра, операций заказа билета.

1. **Информационные сущности**

При анализе темы выделились следующие сущности:

* Фильм (называние, язык, продолжительность, возрастной лимит, дата релиза, страна;
* Работник, обслуживающий сеанс (имя, фамилия, номер телефона);
* Зал (номер, количество мест, количество рядов);
* Клиент (псевдоним в системе, имя, фамилия, номер телефона);
* Пользователь(логин, пароль);
* Роль(рядовой пользователь/администратор);
* Сеанс (время, дата, фильм, цена, номер зала, фильм, обслуживающий работник);
* Билет(держатель билета);
* Место(номер ряда, номер места, номер зала);

1. **Функциональные требования**

Пользовательский интерфейс системы “Кинотеатр” будет реализован на языке Python в виде оконного приложения.

* 1. Возможности гостя

1. Регистрация в системе
   1. Посетитель указывает пользовательские данные (псевдоним в системе, имя, фамилия, номер телефона, пароль.
2. Авторизация в системе
   1. Пользователь должен указать логин и пароль.
3. Просмотр информации о сеансах.
   1. Возможности пользователя
4. Выход из системы
   1. По требованию (по нажатию кнопки «Выход»).
   2. После 10-и минут неактивности происходит автоматический выход из системы.
5. Заказ билета
   1. Выбор сеанса. В случае наличия мест, происходит заказ.
   2. Предоставление возможности заказа нескольких билетов.
6. Мои билеты
   1. Предоставление возможности просмотра заказанных билетов.
   2. Предоставление возможности просмотра информации о заказанных ранее талонах на прошедшие сеансы.
   3. Предоставление возможности отмены заказанного талона.
   4. Возможности администратора
7. Управление пользователями (создание, редактирование, удаление)
   1. Для создания пользовательского аккаунта администратор должен корректно заполнить все поля, определяющие пользовательский аккаунт.
8. Возможность добавления информации о сеансах и фильмах.
9. Просмотр статистической информации
   1. Количество заказанных билетов на определенные сеансы.
   2. Просмотр информации о залах и работниках.
   3. Хранимые процедуры

В системе должны быть созданы и использованы следующие хранимые процедуры и функции в схемах Oracle:

* вывести список фильмов или сеансов по заданному критерию;
* вывод данных о сеансе;
* вывод данных о фильме;
* просмотр свободных мест;
* вывод наиболее востребованных за день фильмов;
* просмотр своей истории заказов;
* получение данных о работнике, обслуживающем сеанс
* заполнение данных о фильме, работниках, формирование сеансов(с проверкой непересечения с другими сеансами в этом зале)(для администратора).

1. **Требования к безопасности**
2. Аутентификация по комбинации логин + пароль.
3. Безопасное хранение пользовательских данных.
   1. Хранение хэшированных паролей вместо открытого текста.
   2. Ограничение доступа к БД.