

**ОТЧЕТ**

**по производственной практике ПП.02.01**

**по профессиональному модулю ПМ.02 «Разработка и администрирование баз данных»**

**специальность** **09.02.03 «Программирование в компьютерных системах»**

Выполнил студент группы П1-17

Лихторенко О.С.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (подпись)

Принял преподаватель

Родичкин П.Ф.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (оценка)

Королёв, 2021

**Содержание**

[**Общие сведения о МГОТУ** 3](#_Toc67495900)

[**Сфера деятельности МГОТУ, история появления и развития МГОТУ** 3](#_Toc67495901)

[**Структура предприятия, функции подразделения и взаимосвязь с другими подразделениями.** 6](#_Toc67495902)

[**Список задач подразделения в сфере программирование в компьютерных системах** 7](#_Toc67495903)

[**Введение** 10](#_Toc67495904)

[**Глава 1. Теоретическая часть** 11](#_Toc67495905)

[1.1 Изучение предметной области 11](#_Toc67495906)

[1.2 Изучение существующих разработок 13](#_Toc67495907)

[**Глава 2. Проектная часть** 16](#_Toc67495908)

[2.1 Построение диаграммы прецедентов 16](#_Toc67495909)

[2.2 Выбор инструментов 20](#_Toc67495910)

[2.3 Проектирование сценария 21](#_Toc67495911)

[2.4 Построение диаграммы классов 22](#_Toc67495912)

[2.5 Описание главного модуля 23](#_Toc67495913)

[2.6 Описание спецификаций к модулям 25](#_Toc67495914)

[2.7 Описание модулей 27](#_Toc67495915)

[2.8 Описание тестовых наборов модулей 28](#_Toc67495916)

[2.9 Описание применения средств отладки 30](#_Toc67495917)

[2.10 Анализ оптимальности использования памяти и быстродействия 31](#_Toc67495918)

[**Глава 3. Эксплуатационная часть** 35](#_Toc67495919)

[3.1. Руководство оператора 35](#_Toc67495920)

[3.1.2. Назначение программы 35](#_Toc67495921)

[3.1.3. Условия выполнения программы 35](#_Toc67495922)

[3.1.4. Выполнение программы 35](#_Toc67495923)

[**Заключение** 40](#_Toc67495924)

[**Список литературы и интернет-источников** 41](#_Toc67495925)

[**Приложения** 42](#_Toc67495926)

# **Общие сведения о МГОТУ**

# **Сфера деятельности МГОТУ, история появления и развития МГОТУ**

МГОТУ - учреждение высшего образования Московской области.

В МГОТУ производят образовательную и научную деятельность.

Университет образован 16 июля 1998 года в форме некоммерческой организации с названием: Негосударственное образовательное учреждение «Королевская академия управления, экономики и социологии».

27 апреля 1999 года наименование было изменено на Негосударственное образовательное учреждение «Королевский институт управления, экономики и социологии».

Постановлением Главы города Королёва от 13 июня 2001 года №1009 Королевский институт управления, экономики и социологии был преобразован в муниципальное образовательное учреждение.

26 декабря 2005 года постановлением Правительства Московской области № 964/52 вуз перешёл в собственность Московской области и получил наименование: Государственное образовательное учреждение высшего профессионального образования Московской области «Королевский институт управления, экономики и социологии». Этим же постановлением учредителем вуза было назначено Министерство образования Московской области.

Постановлением Правительства Московской области от 25.09.2007 года № 725/32 Государственное образовательное учреждение высшего профессионального образования Московской области «Королевский институт управления, экономики и социологии» передано в ведомственное подчинение Министерства финансов Московской области.

В конце декабря 2011 года Приказом Феде­ральной службы по надзору в сфере образования и науки КИУ­ЭС признан прошедшим проце­дуру государственной аккреди­тации с установлением статуса «академия». 12 апреля 2012 года постановлением Правительства Московской области № 484/13 вуз переименован в Государственное бюджетное образовательное учреждение высшего профессионального образования Московской области «Финансово-технологическая академия».

11 июля 2012 года постановлением Правительства Московской области № 906/24 Академия была реорганизована в форме присоединения к ней государственного бюджетного образовательного учреждения среднего профессионального образования Московской области «Королёвский государственный техникум технологии и дизайна одежды» и государственного бюджетного образовательного учреждения среднего профессионального образования Московской области «Королёвский колледж космического машиностроения и технологии»

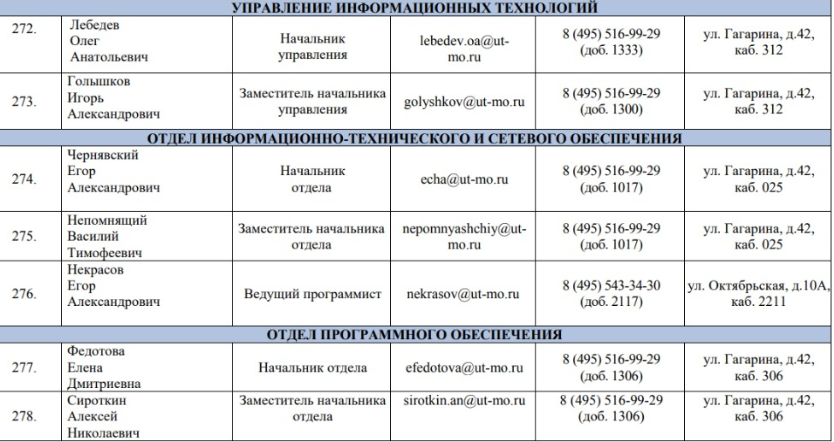
26 декабря 2013 года постановлением Правительства Московской области от 26.12.2013 № 1137/57 «О передаче Государственного бюджетного образовательного учреждения высшего профессионального образования Московской области «Финансово-технологическая академия» в ведомственное подчинение Министерству образования Московской области» Академия была передана в ведомственное подчинение Министерству образования Московской области.

20 января 2015 года постановлением Правительства Московской области от 20.01.2015 № 9/2 Академии присвоен статус «университета» и вуз переименован в Государственное бюджетное образовательное учреждение высшего образования Московской области «Технологический университет».

3 апреля 2015 года Технологическим университетом, Администрацией г.о. Королёв и ведущими предприятиями ракетно-космической отрасли РФ, работающими в наукограде Королёве, ряде городов Московской области и г. Москве, подписано соглашение о создании консорциума «Региональный научно-образовательный кластер «Северо-восток» с целью обеспечения потребностей наукоёмких высокотехнологичных предприятий квалифицированными кадрами.

22 ноября 2019 года постановлением Губернатора Московской области № 564-ПГ Технологическому университету присвоено имя дважды Героя Советского Союза, лётчика-космонавта А.А. Леонова.

**Подразделения МГОТУ**





# **Структура предприятия, функции подразделения и взаимосвязь с другими подразделениями.**

**Подразделения информационных технологий:**

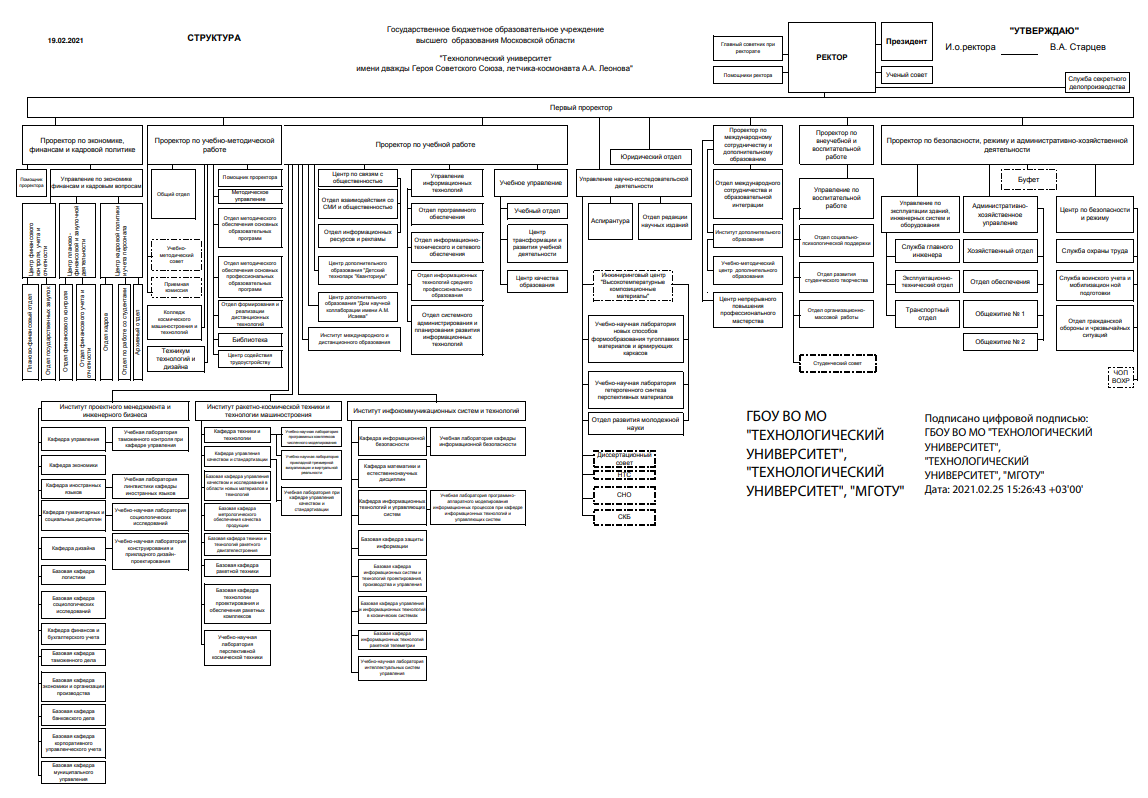
- Отдел управления информационных технологий

- Отдел информационно-технического и сетевого обеспечения

- Отдел программного обеспечения

- Отдел системного администрирования и планирования развития информационных технологий

- Отдел информационных технологий среднего профессионального обеспечения



Структура «Технологического университета».

# **Список задач подразделения в сфере программирование в компьютерных системах**

**Отдел управления информационных технологий:**

* Анализ, внедрение и контроль развития информационных систем;
* Сопровождение информационных систем;
* Предоставление и контроль доступа к информационным ресурсам организации;
* Материально-техническое обеспечение организации средствами автоматизации и связи;
* Ведение технологической документации;

**Отдел информационно-технического и сетевого обеспечения:**

* Организацию электронного документооборота, в том числе применение средств электронной цифровой подписи или иных аналогов собственноручной подписи;
* применение средств антивирусной защиты;
* Ведение электронных журналов учета выполненных операций;
* Ограничение доступа к техническим средствам;
* Ежедневное копирование на резервный носитель содержащейся в [информационной системе](https://pandia.ru/text/category/informatcionnie_sistemi/) информации, обеспечивающее возможность ее восстановления;
* Контроль целостности размещенных на официальном сайте Ставропольского края сведений о контракте (его изменении) или сведения об исполнении (о прекращении действия) контракта и их защиту от несанкционированного изменения или уничтожения;
* Бесперебойную работу по ведению реестра контрактов;

**Отдел программного обеспечения:**

* Разработка и оснащение станков с ЧПУ управляющими программами обработки деталей выпускаемых предприятием изделий, а также новых видов изделий и технической оснастки;
* Разработка и реализация технических мероприятий по снижению всех видов затрат на единицу продукции;
* Расчет и внедрение в производство оптимальных норм расходования основных и вспомогательных материалов и обеспечение всех соответствующих служб нормами расхода основных и вспомогательных материалов;
* Обеспечение всех соответствующих служб технической документацией, хранящейся в архиве КТЦ;
* Обеспечение автоматизации рабочих мест конструкторов и технологов;

**Отдел системного администрирования и планирования развития информационных технологий:**

* Ремонт используемой техники;
* Обеспечение бесперебойной работы всех ПК и устранение неисправностей;
* Помощь штатным сотрудникам, работающим с ПК и офисной техникой, электронной почтой и т.д. в случае, возникновения трудностей;
* Обновление и закупка нужной техники и их комплектующих;
* Обеспечение нормальной работы используемых операционных систем и набор рабочих программ (ОС Windows, MS Office и т.п.);
* Установка и настройка ПО для корректной работы;
* Своевременное обновление необходимого ПО;
* Обеспечение информационной безопасности и защиты от хакерских атак и от спама;
* Создание резервных копий данных, удаление и их восстановление в случае необходимости;

**Отдел информационных технологий среднего профессионального обеспечения:**

* реализация концепции развития информационных систем;
* обеспечение требуемого уровня информационной безопасности;
* разработка стандартов на использование вычислительной техники и программного обеспечения;
* обеспечение информационной и технической поддержки средств вычислительной техники и программного обеспечения;
* проведение работ по оптимизации использования информационно технических ресурсов;

# **Введение**

Целью практики является написание программы «Кинотеатр» для кинотеатра второго экрана вида миниплекс. Данная программа позволит облегчить работу сотрудникам, главным образом букеру, путем систематизации данных о сеансах, проекторов, количества мест и учетов лицензионных ключей.

Далее будут описаны цели программы, ее применение и сама программа. отчет разделен на три главы.

В первой главе будет рассмотрена предметная область данной тем, а также приведены примеры уже существующих разработок в данной области, их описание, характеристики и отзывы пользователей.

Во второй части будут рассмотрены инструменты и модули, которые были разработаны, структура программной части и листинги ключевых частей программных модулей.

В третьей части будет рассмотрено руководство для пользователей, а также изображение графического интерфейса программы.

В результате этих трех главах будет полностью описана разработка, ее суть и назначения, а также сама область применения. Что полностью опишет всю разработку.

# **Глава 1. Теоретическая часть**

## Изучение предметной области

Первый кинотеатр появился в конце 1890-х годов вместе с зарождением [кинематографа](https://ru.wikipedia.org/wiki/%D0%9A%D0%B8%D0%BD%D0%B5%D0%BC%D0%B0%D1%82%D0%BE%D0%B3%D1%80%D0%B0%D1%84).

Кинотеатры подразделяются на: круглогодичные (многозальные, однозальные) и сезонные (летние открытые и летние закрытые).

Цифровые копии, в том числе и 3D фильмы. Приходят в специальных кейсах, на переносных HDD дисках, либо по защищенной сети или спутниковый ретранслятор. Копии приходят в специальном формате DCP (Digital Cinema Package). Фильм в среднем весит от 230 до 350 гигабайт. Эта технология практически вытеснила традиционную пленочную благодаря многим преимуществам, например возможность проведения мировых премьер одновременно в разных точках планеты.

Для того, чтобы показать картину, кинотеатру необходимо получить лицензию на показ и цифровой ключ (KDM, Key Delivering Message), в котором зашифровано количество сеансов и их время. Также в ключе указываются серийные номера конкретных серверов, которыми можно показать кинокартину, а с 2009 года и номера кинопроекторов. В случае поломки сервера его замена должна согласовываться с компанией-дистрибьютором, у которой запрашивается новый ключ KDM. Та же процедура обязательна при смене расписания киносеансов, в противном случае показ будет невозможен вследствие блокировки сервера лицензионным ключом.

На данный момент существуют следующие виды кинотеатров:

* Моноэкран — кинотеатр с одним экраном;
* Миниплекс — комплекс с количеством экранов от 2 до 8;
* Мультиплекс — кинотеатр от 9 до 15 залов;
* Мегаплекс — от 16 залов;
* Киноплекс — кинотеатр, расположенный в торговом или торгово-развлекательном центре;
* Кинотеатры «первого экрана» — выпускают фильмы в день премьеры по всей стране;
* Кинотеатры «второго экрана» — прокатывают фильмы через две-три недели после премьеры на «первом экране».

## Изучение существующих разработок

Рассмотрим приложение для работников и гостей кинотеатра компании ООО «Киноплан». Все материалы взяты с их официального сайта.

Реализована автоматическая отчётность перед дистрибьюторами, налоговыми и государственными службами, что упрощает работу кассира и букера.

Функция планирование репертуара (рис. 1), так же облегчает работу букера и менеджера кинотеатра.

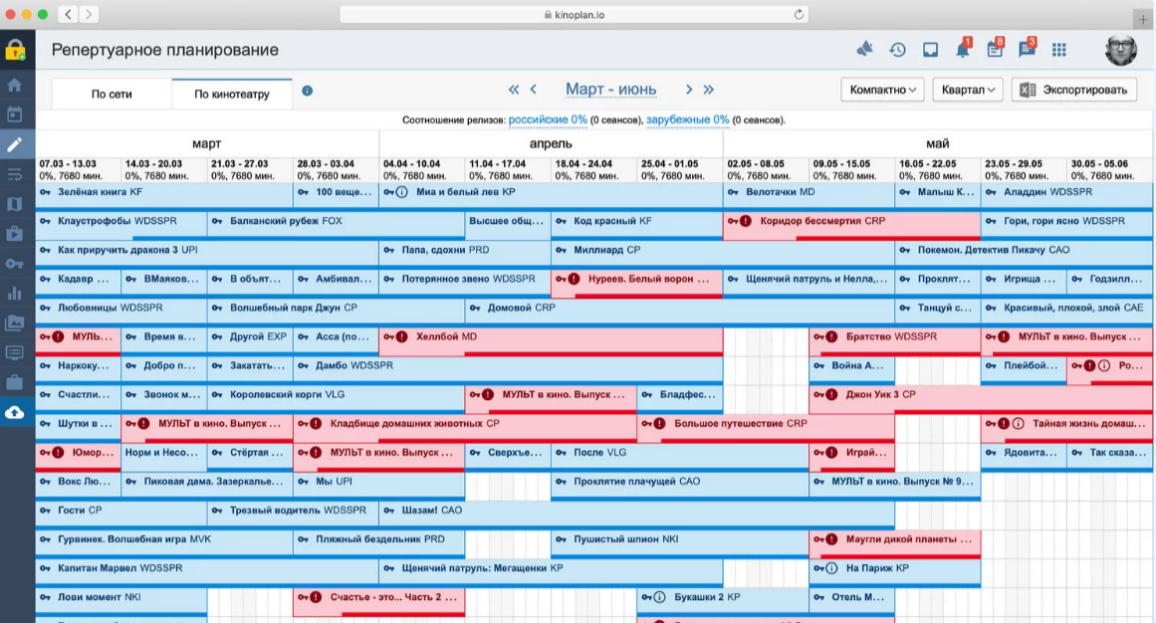


Рисунок 1. Планирование репертуара

Автоматическое расписание сеансов (рис. 2), позволяет экономить время и заранее планировать долгосрочные премьеры и показы.

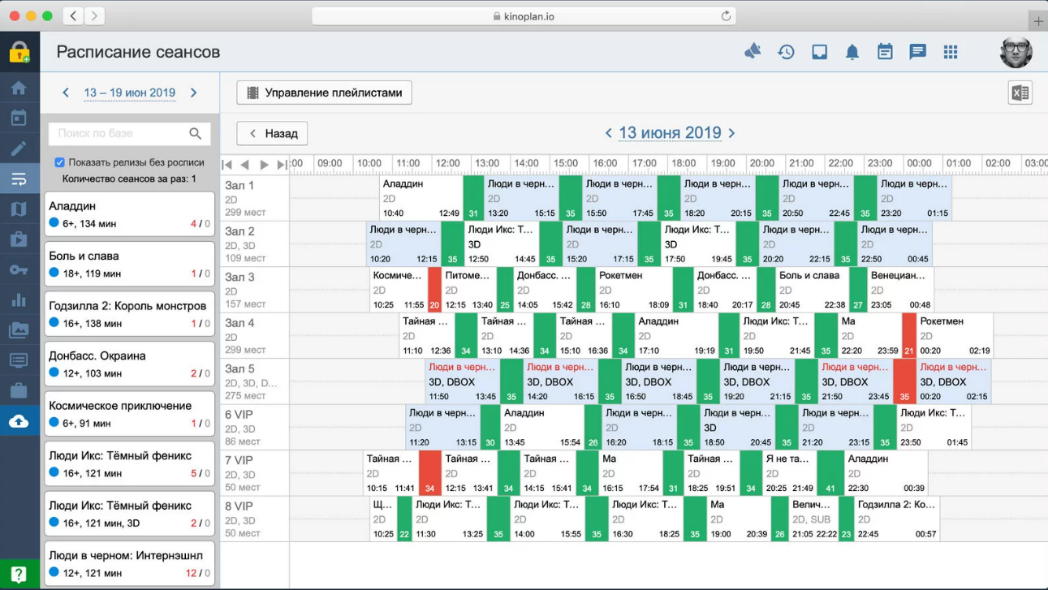


Рисунок 2. Расписание

Функция управление KDM ключами (рис. 3), позволяет сократить персонал и рабочие часы.

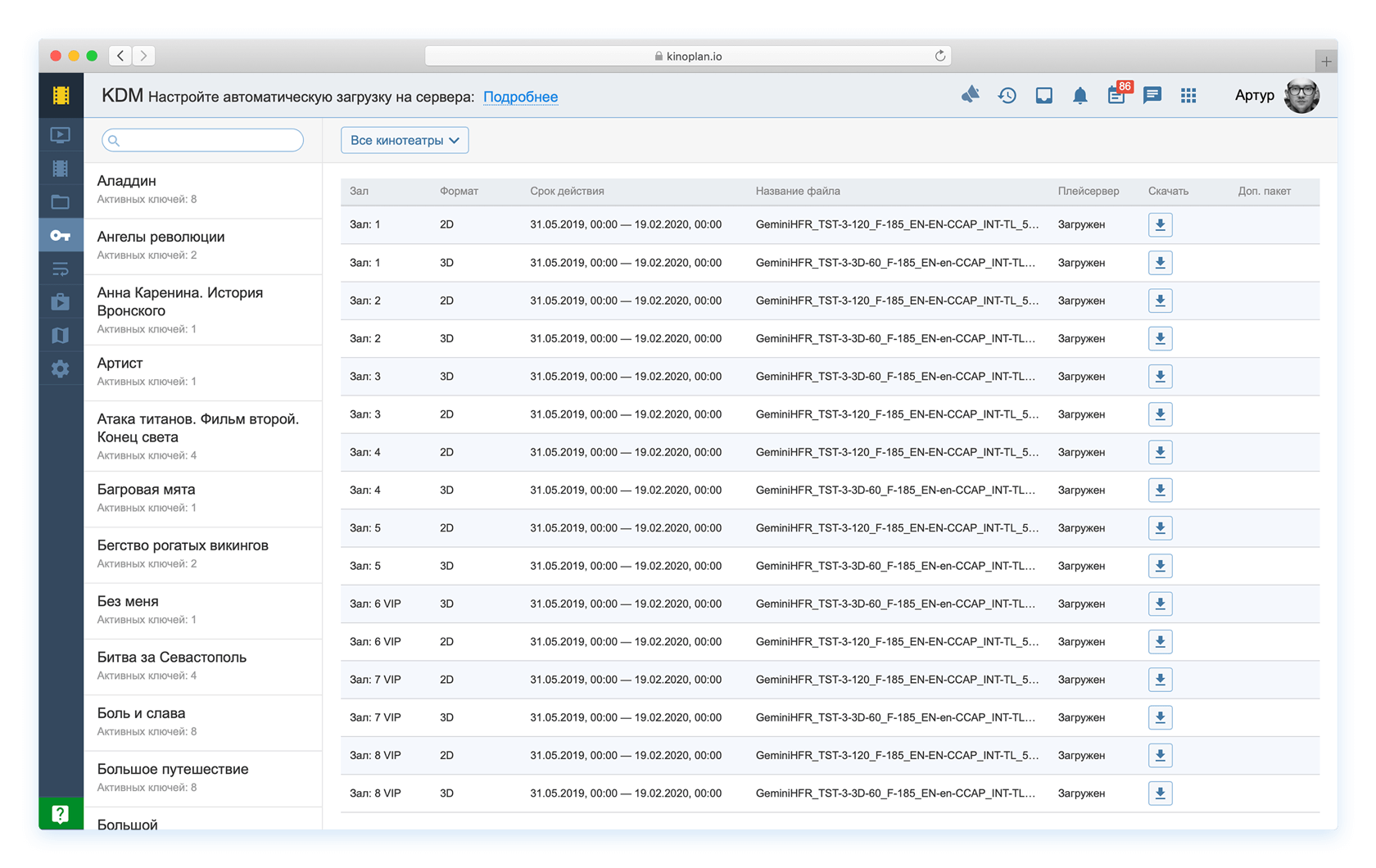


Рисунок 3. KDM ключи

# **Глава 2. Проектная часть**

## Построение диаграммы прецедентов

Были построены диаграммыпрецедентов для «Администратора», «Букера», «Киномеханика», «Кассира» и «Клиента»

На данных диаграммах мы можем увидеть какой пользователь за какое действие отвечает. Далее вы можете рассмотреть общую диаграмму прецедентов (рис. 4).

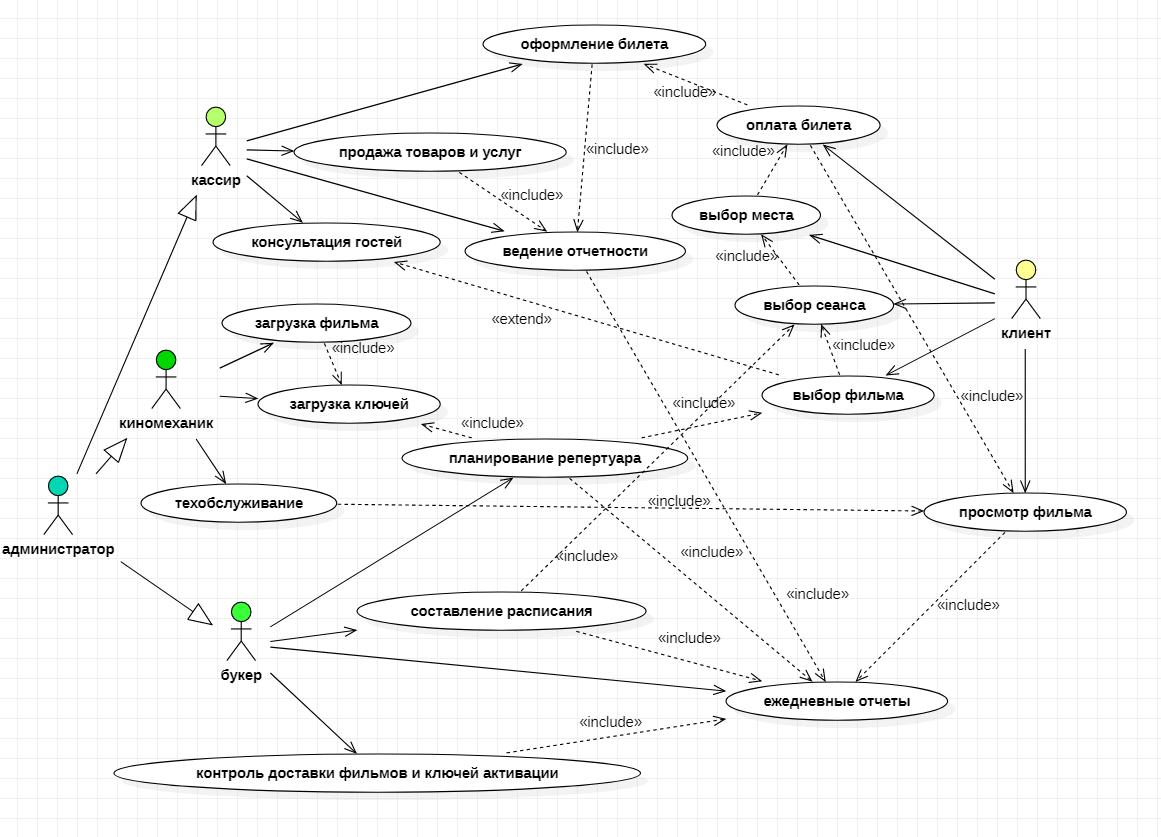


Рисунок 4. Общая диаграмма

Диаграмма прецедентов «Администратора» (рис. 5), показывает что администратор осуществляет контроль за работниками.

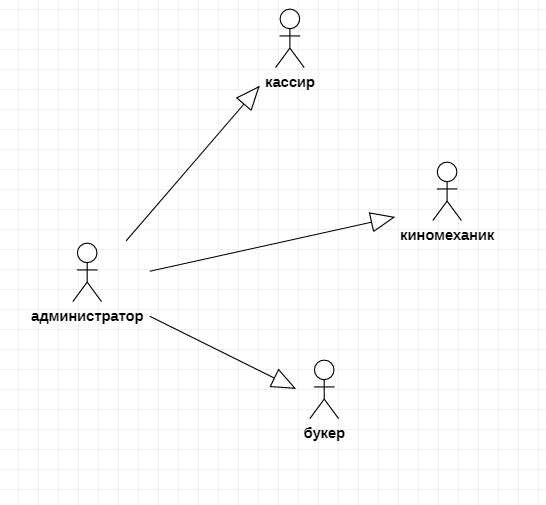


Рисунок 5. Диаграмма администратора

Диаграмма прецедентов «Кассира» (рис. 6). В обязанности входит: обслуживание клиентов, ведение отчетности проданных билетов и кассы

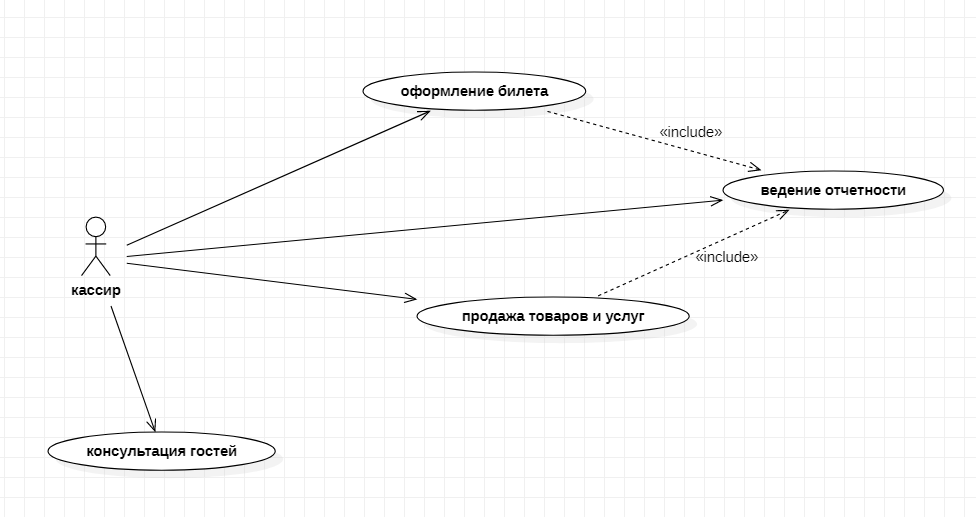


Рисунок 6. Диаграмма кассира

Диаграмма прецедентов «Киномеханик» (рис. 7) показывает, что киномеханик не ведет никакой отчетности, но обслуживает проекторы и залы

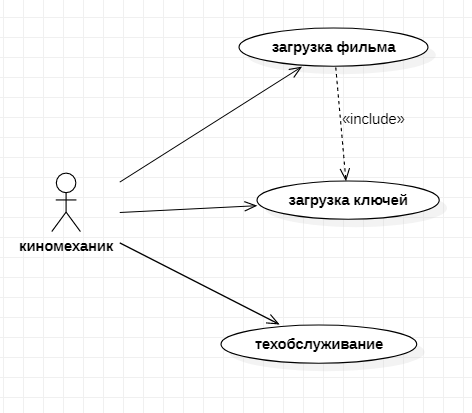


Рисунок 7. Диаграмма киномеханика

Один из самых важных работников – Букер. Как мы видим (рис. 8), кинопоказ осуществляется под его надзором. В его обязанности входит планирование репертуара и расписания кинотеатра, и, самое главное, на чем и будем делать акцент – контроль за KDM.



Рисунок 8. Диаграмма букера

Диаграмма «Клиент» (рис. 9) показывает один из сценариев поведения человека, пришедшего в кинотеатр.

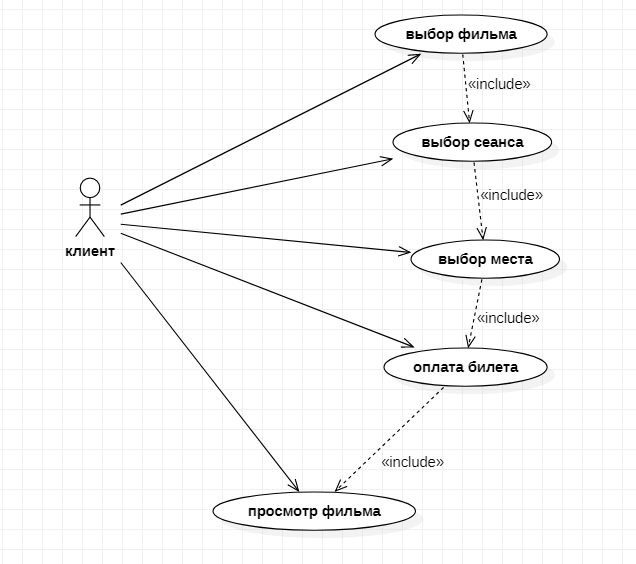


Рисунок 9. Диаграмма клиента

## Выбор инструментов

При выборе инструментов обращалось внимание на определенные критерии:

* Наличие необходимых библиотек
* Возможность подключений сторонних систем
* Наличие документации на русском языке
* Возможность работы на различных платформах
* Доступность

После некоторых поисков был выбран язык программирования Object Pascal, среда разработки Lazarus, GUI-оболочка IBExpert и система управления базами данных Firebird.

## Проектирование сценария

Данная программа ориентирована на такой сценарий (рис. 10).

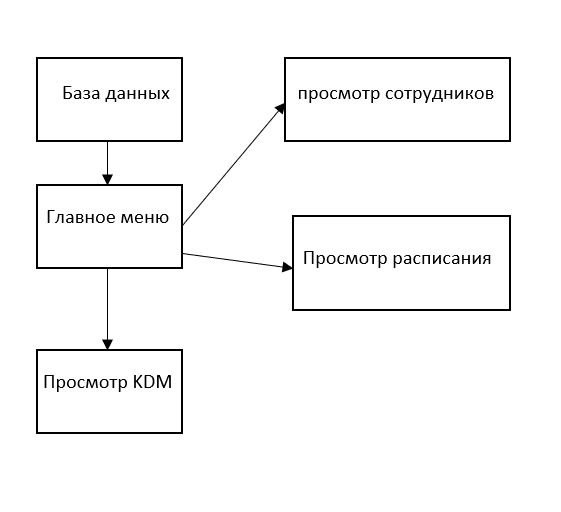


Рисунок 10. Сценарий

После запуска программы пользователь имеет несколько вариантов действий

1. Просмотр сотрудников
2. Просмотр ключей KDM с актуальной информацией
3. Просмотр расписание кинотеатра
4. Удаление, добавление и редактирование таблицы сотрудников
5. Работа с KDM
6. Работа с расписанием

При выборе просмотра любой таблицы пользователь будет получать готовую к просмотру соответствующую таблицу.

## Построение диаграммы классов

В данном параграфе представлены все классы, которые используются в программе, а также их взаимосвязи (рис. 11).

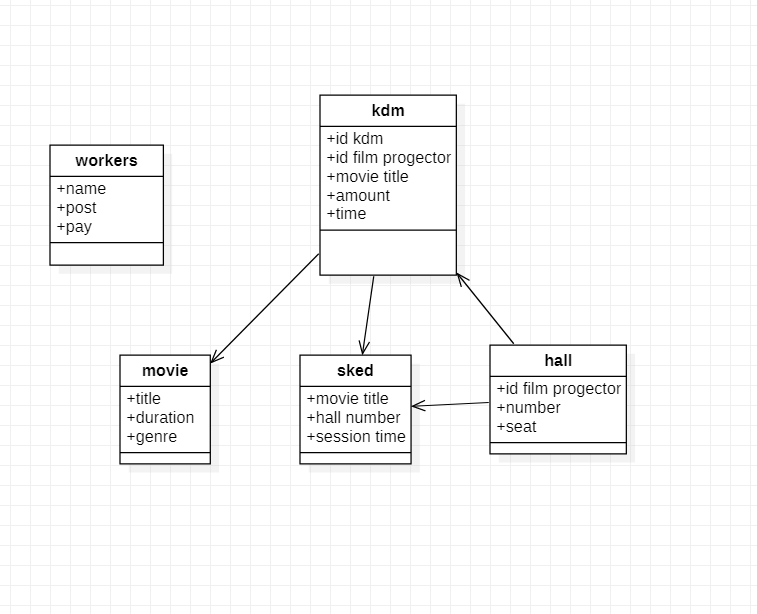


Рисунок 11. Диаграмма классов

## Описание главного модуля

В главный (центральный) модуль разработки входит класс Подсоединение БД, которые отвечает за взаимосвязь со всеми компонентами Базы данных, а также за их включение.

Данный модуль запускается автоматически при запуске программы что обеспечивает качественное соединение и отсутствие ошибок во время работы.

Данный модуль состоит из одного кода, который приведен в листинге 1.

**Листинг 1:**

unit umain;

{$mode objfpc}{$H+}

Interface

uses

Classes, SysUtils, db, Forms, Controls, Graphics, Dialogs, DBGrids, DBCtrls,

Calendar, EditBtn, StdCtrls, DBExtCtrls, ExtDlgs, ComCtrls, PopupNotifier,

ExtCtrls, Buttons, Menus, IBTable, VirtualTrees, DBDateTimePicker,

DateTimePicker, RTTIGrids, RTTICtrls, Types;

type

{ TFMain }

TFMain = class(TForm)

DBGrid1: TDBGrid;

DBGrid2: TDBGrid;

DBGrid3: TDBGrid;

DBGrid4: TDBGrid;

DBNavigator1: TDBNavigator;

DBNavigator2: TDBNavigator;

DBNavigator3: TDBNavigator;

DBNavigator4: TDBNavigator;

movie: TPageControl;

TabSheet1: TTabSheet;

workers: TTabSheet;

KDM\_key: TTabSheet;

timetable: TTabSheet;

procedure cmbkdmChange(Sender: TObject);

procedure DBComboBox1Change(Sender: TObject);

procedure DBNavigator1Click(Sender: TObject; Button: TDBNavButtonType);

procedure sdfsClick(Sender: TObject);

//procedure movieChange(Sender: TObject);

procedure KDM\_keyContextPopup(Sender: TObject; MousePos: TPoint;

var Handled: Boolean);

//procedure movieChange(Sender: TObject);

procedure TILabel1Click(Sender: TObject);

procedure timetableContextPopup(Sender: TObject; MousePos: TPoint;

var Handled: Boolean);

procedure TIProgressBar1DragDrop(Sender, Source: TObject; X, Y: Integer);

procedure workersContextPopup(Sender: TObject; MousePos: TPoint;

var Handled: Boolean);

private

public

end;

## Описание спецификаций к модулям

Центральный модуль представляет из себя набор обязательных элементов базы данных таких как таблицы, запросы и транзакции. И центральную функцию, которая представляет из себя запуск и настройку базы данных.

Модуль просмотра таблиц представляет из себя несколько графических форм каждая из которых обращается в центральный модуль за своим запросом из базы данных для дальнейшей его вывода, так же подключает навигацию к полученному запросу. Так же этот модуль имеет два скрипта которые представляют из себя:

* 1. Подстановку в выпадающий список соответствующе значения;
  2. При выборе значения из списка производит изменение скрипта запроса и выводит новую отсортированную таблицу.

**Листинг 2. Модуль просмотра таблиц:**

unit udm;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, db, IBDatabase, IBTable, IBCustomDataSet, IBQuery, StdCtrls;

type

{ Tdm }

Tdm = class(TDataModule)

DataSource1: TDataSource;

dtsfilm: TDataSource;

dtsTimetable: TDataSource;

dtsWorkers: TDataSource;

ibdb: TIBDatabase;

ibfilmDURATION: TTimeField;

ibfilmGENRE: TIBStringField;

ibfilmNAME: TIBStringField;

ibkdm: TIBTable;

ibkdmAMOUNT: TIBIntegerField;

ibkdmID: TIBIntegerField;

ibkdmID\_FILM\_PROJECTOR: TIBIntegerField;

ibkdmNAME: TIBStringField;

ibkdmTIME: TTimeField;

ibqKDMID: TIBIntegerField;

ibqKDMID\_FILM\_PROJECTOR: TIBIntegerField;

ibqKDMNAMBER: TIBIntegerField;

ibqKDMNAME: TIBStringField;

ibqKDMSEAT: TIBIntegerField;

ibfilm: TIBTable;

ibTimetable: TIBTable;

IBTable1AMOUNT: TIBIntegerField;

IBTable1ID: TIBIntegerField;

IBTable1ID\_FILM\_PROJECTOR: TIBIntegerField;

IBTable1NAME: TIBStringField;

IBTable1TIME: TTimeField;

ibTimetableNAMBER: TIBIntegerField;

ibTimetableNAME: TIBStringField;

ibTimetableSESSIONTIME: TTimeField;

ibWorkers: TIBTable;

ibtr: TIBTransaction;

ibWorkersID: TIBIntegerField;

ibWorkersNAME: TIBStringField;

ibWorkersPAY: TIBBCDField;

ibWorkersPOST: TIBStringField;

procedure DataModuleCreate(Sender: TObject);

procedure DataSource1DataChange(Sender: TObject; Field: TField);

procedure dtsfilmDataChange(Sender: TObject; Field: TField);

procedure dtsKDMDataChange(Sender: TObject; Field: TField);

procedure dtsTimetableDataChange(Sender: TObject; Field: TField);

procedure dtsWorkersDataChange(Sender: TObject; Field: TField);

private

public

skladIndexes:TstringList;

procedure LoadTable(tableName:string;strl:TstringList;cmd:Tcombobox);

end;

var

dm: Tdm;

implementation

{$R \*.lfm}

{ Tdm }

procedure Tdm.DataModuleCreate(Sender: TObject);

begin //вручную открываем необходимые таблицы

ibtr.StartTransaction;

ibWorkers.Open;

ibTimetable.Open;

//ibqKDM.Open;

ibkdm.Open;

ibfilm.Open;

// KMDIndexes:=TstringList.Create;

// LoadTable('KDM\_key',nil,FMain.cmbkdm);

end;

## Описание модулей

На рисунке 12 приведен модуль данных, отвечающий за работу с БД. Он содержит в себе компонент транзакции TIBTransaction (ibtr), компонент соединения с базой данных TIBDataBase (ibdb), компоненты таблиц TIBTable (ibTimetable, ibWorkers, ibfilm, ibkdm) и компоненты TDataSource, отвечающие за управление потоками данных, за связи между набором данных и визуальными компонентами (dtsTimetable, dtsWorkers, dtsfilm, DataSource1).

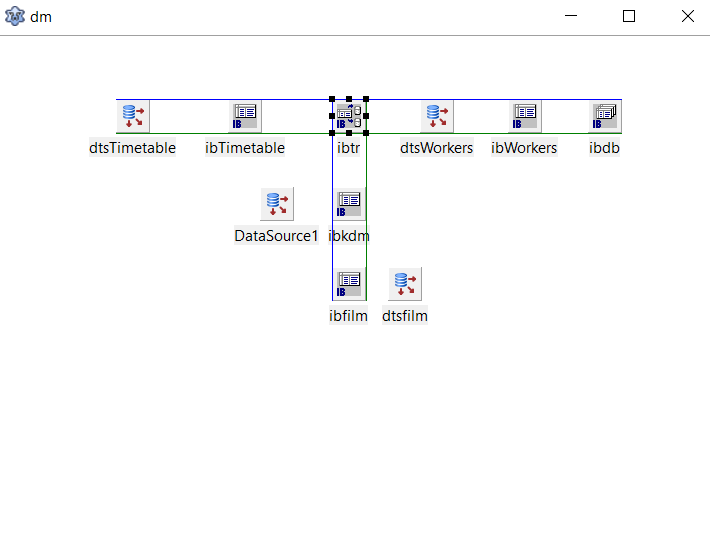


Рисунок 12. Модуль данных

## Описание тестовых наборов модулей

На рисунках ниже приведены результаты работы добавления записи в таблице. Рисунок 13 - до добавлении записи, рисунок 14 – после.

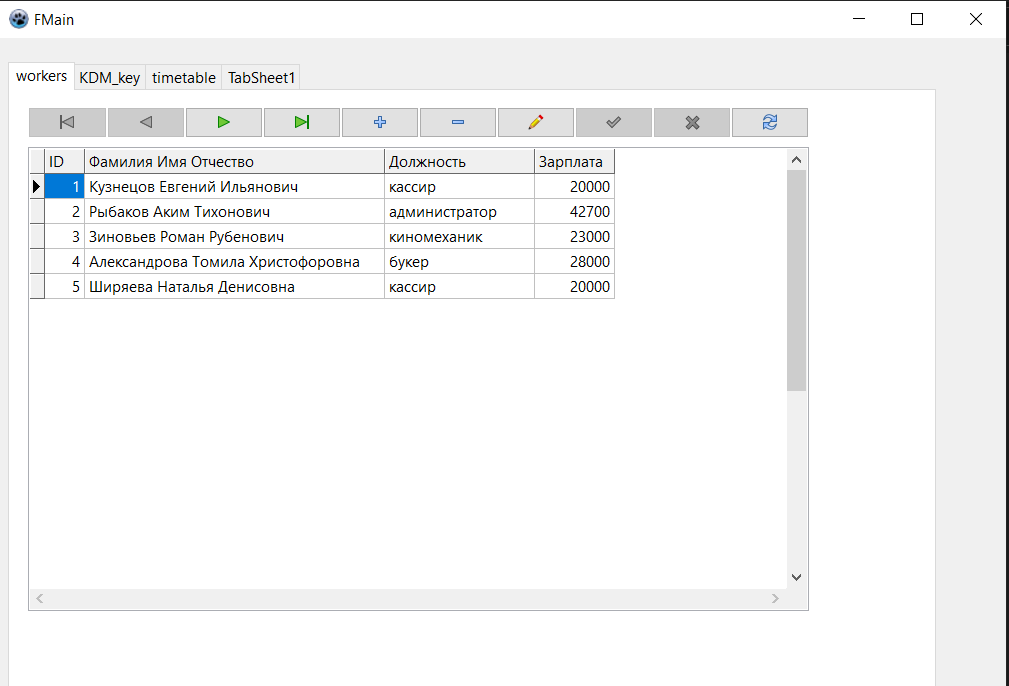


Рисунок 13. До добавления записи о сотруднике

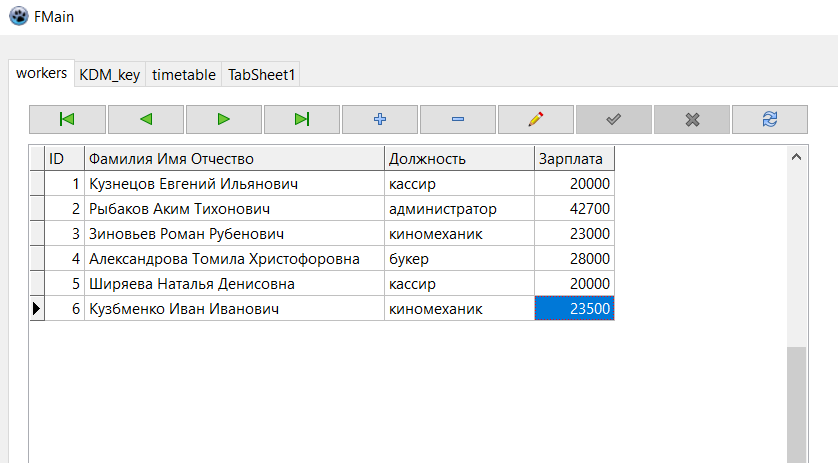


Рисунок 14. С новой записью.

Проверяем можем ли мы удалить записи (рис. 15, рис. 16).

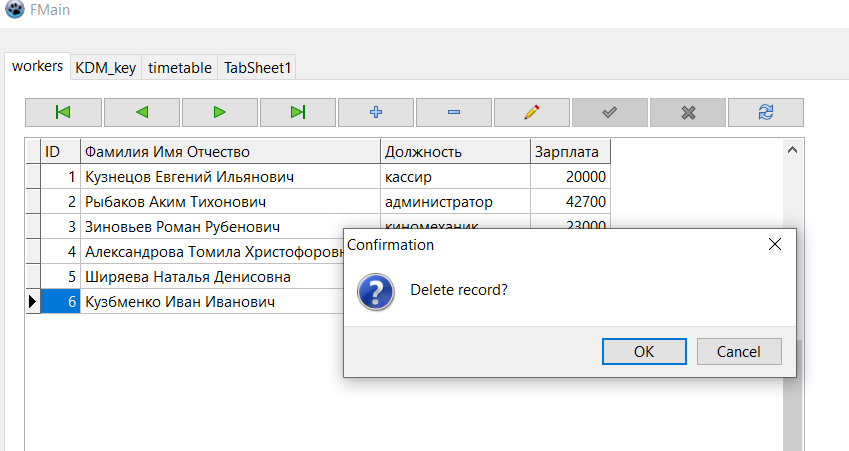


Рисунок 15. Удаление записи о сотруднике

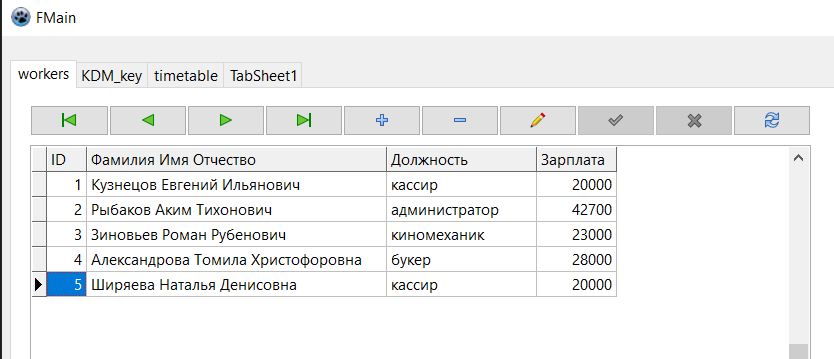


Рисунок 16. Таблица с удаленной записью

Остальные таблицы на вкладках так же успешно проходят тест.

## Описание применения средств отладки

В ходе практики при попытке запустить программу были получены следующие ошибки (рис 17)

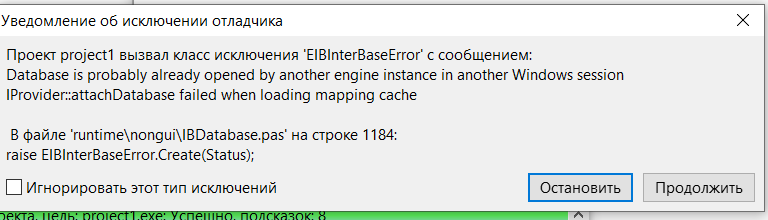


Рисунок 17. Уведомление об исключении

После определенных манипуляций, исключения были успешно ликвидированы.

## Анализ оптимальности использования памяти и быстродействия

Удаление мертвого кода является одним из методов оптимизации. Для быстродействия программы было решено воспользоваться данным методом. Были удалены следующие части кода:

**Листинг 2.10.1. Мертвый код в модуле umain**

procedure TFMain.movieChange(Sender: TObject);

begin

end;

**Листинг 2.10.1. Мертвый код в модуле udm**

ibqKDM.Open;

KMDIndexes:=TstringList.Create;

LoadTable('KDM\_key',nil,FMain.cmbkdm);

ibq.Close;

if not(ibq.Transaction.Active) then ibq.Transaction.StartTransaction;

ibq.sql.Text:='select \* from '+tableName;

ibq.Open;

while not(ibq.EOF) do

begin

//cmb.Items.Add(ibq.FieldByName('NAME').AsString);

ibq.Next;

end;

Размер файла project1.exe достаточно большой. Было решено оптимизировать компиляцию, тем самым уменьшить размер файла. Размер файла до оптимизации показан ниже



Рисунок 17. Размер файла до оптимизации

Для этого было необходимо отключить отладочную информацию и включить «вырезать символы из исполняемого файла».

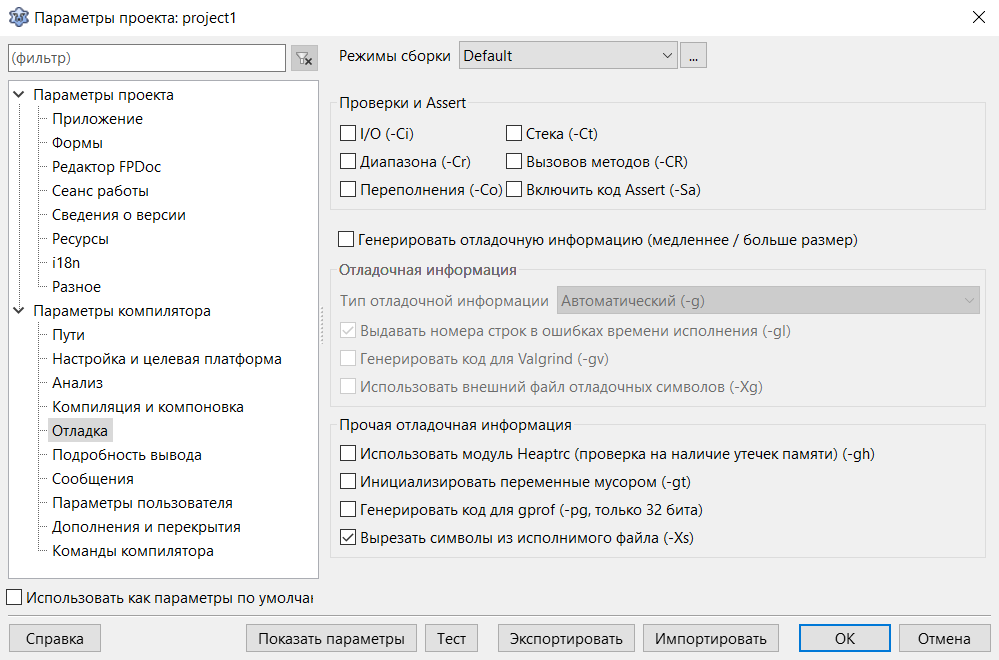


Рисунок 18. Изменение параметров отладки

Также нужно включить умную компоновку в «Стиль модуля» и в «Компоновка»

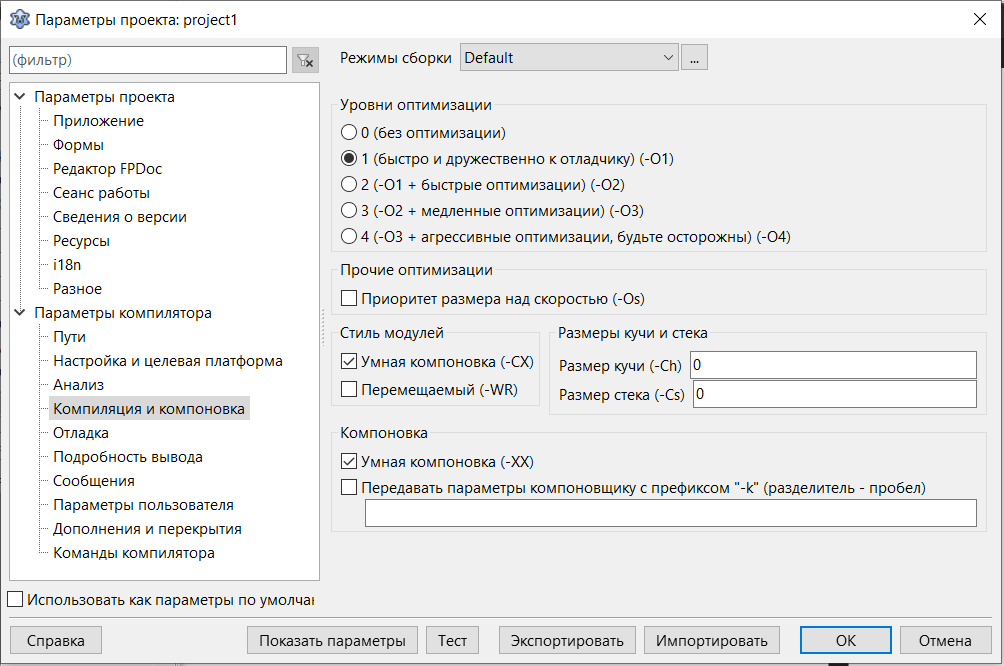


Рисунок 19. Параметры компиляции и компоновки

Для уменьшения памяти можно убрать стандартный значок.

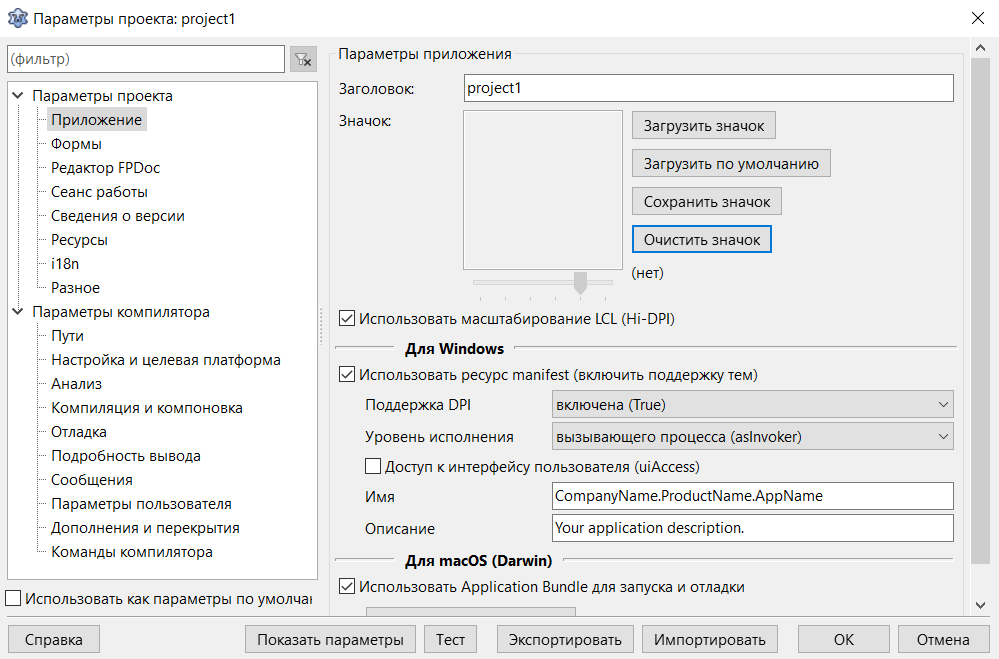


Рисунок 20. Очищение стандартного значка

После проведенной оптимизации компиляции можно заметить существенную разницу в размере файла. До оптимизации размер exe-файла составлял 46027 КБ, а после 5761 КБ



Рисунок 21. Размер файла после оптимизации

# **Глава 3. Эксплуатационная часть**

## 3.1. Руководство оператора

### 3.1.2. Назначение программы

Разработанное приложение позволяет осуществлять различный учет данных на предприятии быстрого питания. При этом пользователю необязательно обладать умением работы с базами данных. Программа работает с таблицами Firebird – записывает все изменения, внесенные пользователем в таблицу.

### 3.1.3. Условия выполнения программы

Минимальные требования:

Операционная система: Windows: 7 x 32bit;

### 3.1.4. Выполнение программы

Перед началом запуска оператор должен записать в файл DataBase.ini путь БД для работы. Если путь указан правильно, то запускается приложение project1

После запуска приложения открывается окно, содержащее четыре вкладки «workers» - работники, «KDM\_key» - KDM ключи, «timetable» - Расписание и «TabSheet1» - список фильмов

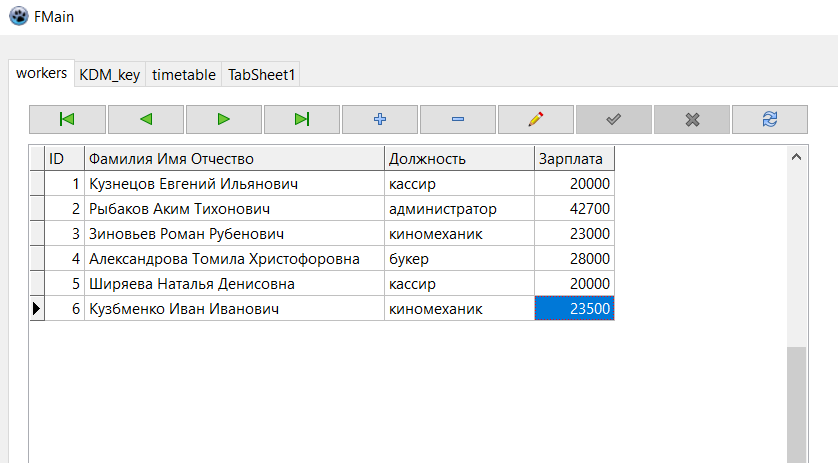


Рисунок 22. Вид приложения при открытии. Вкладка «работники»

При нажатии на одну из предложенных вкладок, открывается соответствующее окно.

Вкладка «KDM\_key» откроет таблицу «KDM ключи». (Рис. 23)

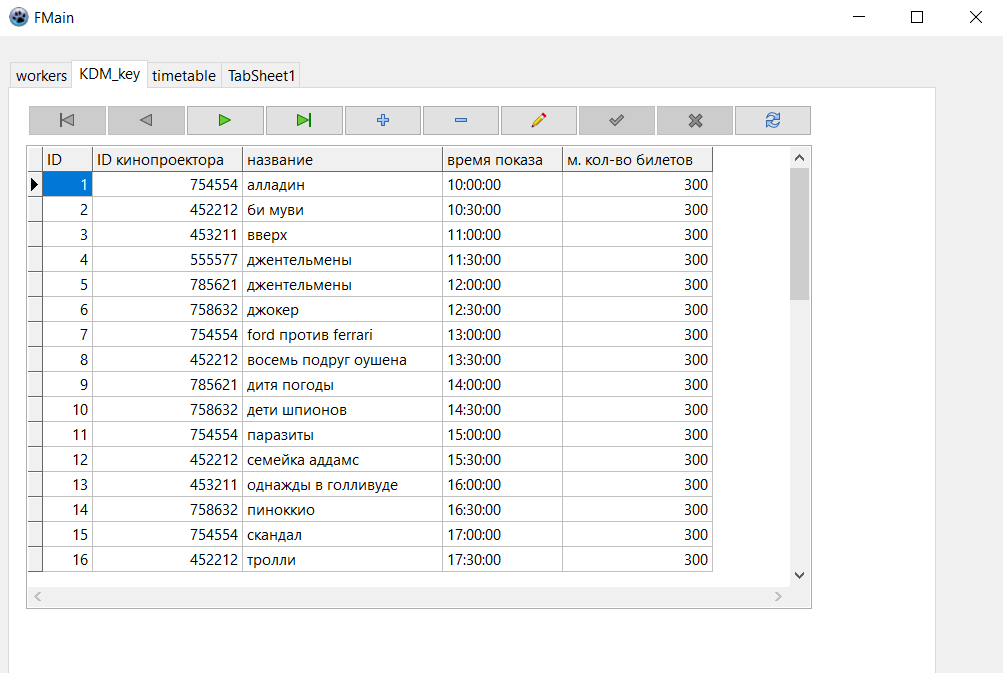


Рисунок 23. Таблица «KDM ключи»

Вкладка «timetable» откроет таблицу «расписание». (Рис. 24)

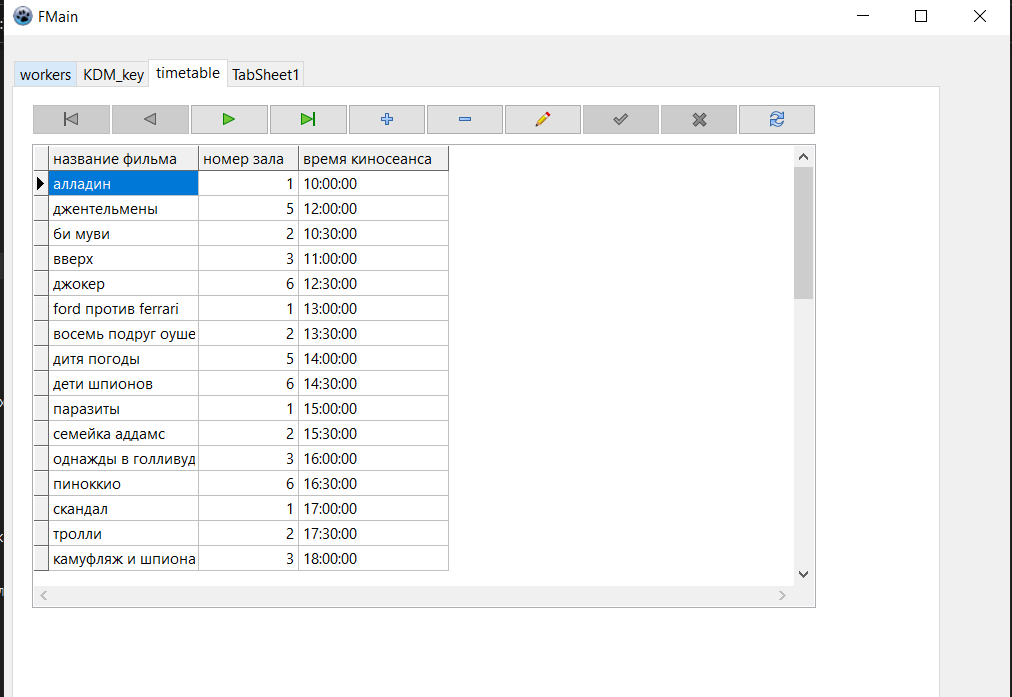


Рисунок 24. Таблица «расписание»

Вкладка «TabSheet1» откроет таблицу «фильмы». (Рис. 25)

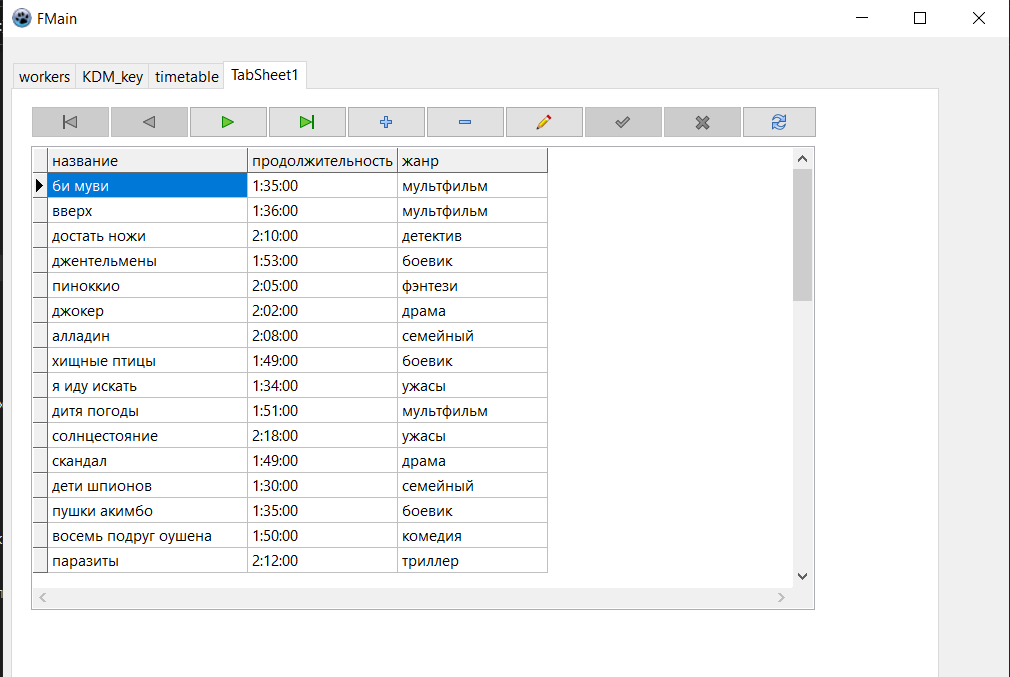


Рисунок 25. Таблица «фильмы»

# **Заключение**

В результате прохождения практики было разработано приложение «Кинотеатр».

В ходе работы были проанализированы существующие разработки, посвященные данному направлению и изучены следующие темы: «Работа с базами данных FireBird» и «Разработка приложения с базами данных».

В дальнейшем планируется разработать сортировку по жанрам, возрастному цензу, добавить выпадающие списки с вариантом выбора интересующего зала и просмотра его расписание на день, добавить окно учета продажи билетов.

# **Список литературы и интернет-источников**

1. Работа в программах, программирование и многое другое...

https://www.youtube.com/channel/UCkLD\_cf5JC58UivY5Ph4Kqg

1. Национальный Открытый Университет "ИНТУИТ". Вячеслав Ачкасов. Лекция Введение в Lazarus

<http://msk.edu.ua/ivk/Informatika/Uch_posobiya/Programmirovanie/Lazarus/1_vvedenie_v_Lazarus.htm>

1. Официальный сайт ООО «Киноплан»

<https://kinoplan.ru/>

1. Официальный сайт Lazarus

https://www.lazarus-ide.org/

1. Официальный сайт FireBird:

<https://www.firebirdsql.org/en/server-packages/>

# **Приложения**

**Модуль umain**

unit umain;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, db, Forms, Controls, Graphics, Dialogs, DBGrids, DBCtrls,

Calendar, EditBtn, StdCtrls, DBExtCtrls, ExtDlgs, ComCtrls, PopupNotifier,

ExtCtrls, Buttons, Menus, IBTable, VirtualTrees, DBDateTimePicker,

DateTimePicker, RTTIGrids, RTTICtrls, Types;

type

{ TFMain }

TFMain = class(TForm)

DBGrid1: TDBGrid;

DBGrid2: TDBGrid;

DBGrid3: TDBGrid;

DBGrid4: TDBGrid;

DBNavigator1: TDBNavigator;

DBNavigator2: TDBNavigator;

DBNavigator3: TDBNavigator;

DBNavigator4: TDBNavigator;

movie: TPageControl;

TabSheet1: TTabSheet;

workers: TTabSheet;

KDM\_key: TTabSheet;

timetable: TTabSheet;

procedure cmbkdmChange(Sender: TObject);

procedure DBComboBox1Change(Sender: TObject);

procedure DBNavigator1Click(Sender: TObject; Button: TDBNavButtonType);

procedure sdfsClick(Sender: TObject);

// procedure movieChange(Sender: TObject);

procedure KDM\_keyContextPopup(Sender: TObject; MousePos: TPoint;

var Handled: Boolean);

//procedure movieChange(Sender: TObject);

procedure TILabel1Click(Sender: TObject);

procedure timetableContextPopup(Sender: TObject; MousePos: TPoint;

var Handled: Boolean);

procedure TIProgressBar1DragDrop(Sender, Source: TObject; X, Y: Integer);

procedure workersContextPopup(Sender: TObject; MousePos: TPoint;

var Handled: Boolean);

private

public

end;

var

FMain: TFMain;

implementation

{$R \*.lfm}

{ TFMain }

**Модуль udm**

unit udm;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, db, IBDatabase, IBTable, IBCustomDataSet, IBQuery, StdCtrls;

type

{ Tdm }

Tdm = class(TDataModule)

DataSource1: TDataSource;

dtsfilm: TDataSource;

dtsTimetable: TDataSource;

dtsWorkers: TDataSource;

ibdb: TIBDatabase;

ibfilmDURATION: TTimeField;

ibfilmGENRE: TIBStringField;

ibfilmNAME: TIBStringField;

ibkdm: TIBTable;

ibkdmAMOUNT: TIBIntegerField;

ibkdmID: TIBIntegerField;

ibkdmID\_FILM\_PROJECTOR: TIBIntegerField;

ibkdmNAME: TIBStringField;

ibkdmTIME: TTimeField;

ibqKDMID: TIBIntegerField;

ibqKDMID\_FILM\_PROJECTOR: TIBIntegerField;

ibqKDMNAMBER: TIBIntegerField;

ibqKDMNAME: TIBStringField;

ibqKDMSEAT: TIBIntegerField;

ibfilm: TIBTable;

ibTimetable: TIBTable;

IBTable1AMOUNT: TIBIntegerField;

IBTable1ID: TIBIntegerField;

IBTable1ID\_FILM\_PROJECTOR: TIBIntegerField;

IBTable1NAME: TIBStringField;

IBTable1TIME: TTimeField;

ibTimetableNAMBER: TIBIntegerField;

ibTimetableNAME: TIBStringField;

ibTimetableSESSIONTIME: TTimeField;

ibWorkers: TIBTable;

ibtr: TIBTransaction;

ibWorkersID: TIBIntegerField;

ibWorkersNAME: TIBStringField;

ibWorkersPAY: TIBBCDField;

ibWorkersPOST: TIBStringField;

procedure DataModuleCreate(Sender: TObject);

procedure DataSource1DataChange(Sender: TObject; Field: TField);

procedure dtsfilmDataChange(Sender: TObject; Field: TField);

procedure dtsKDMDataChange(Sender: TObject; Field: TField);

procedure dtsTimetableDataChange(Sender: TObject; Field: TField);

procedure dtsWorkersDataChange(Sender: TObject; Field: TField);

private

public

skladIndexes:TstringList;

procedure LoadTable(tableName:string;strl:TstringList;cmd:Tcombobox);

end;

var

dm: Tdm;

implementation

{$R \*.lfm}

{ Tdm }

procedure Tdm.DataModuleCreate(Sender: TObject);

begin

ibtr.StartTransaction;

ibWorkers.Open;

ibTimetable.Open;

//ibqKDM.Open;

ibkdm.Open;

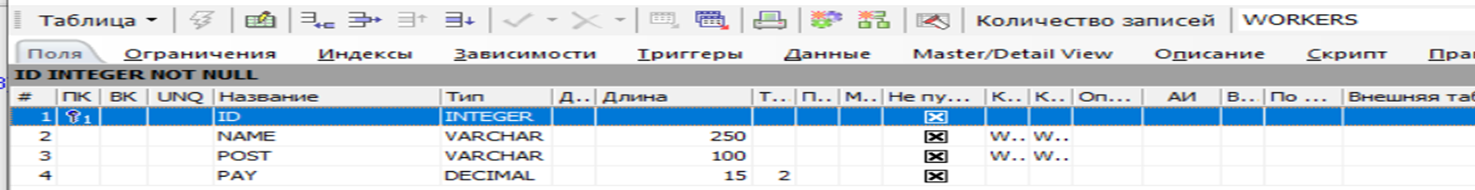
ibfilm.Open;

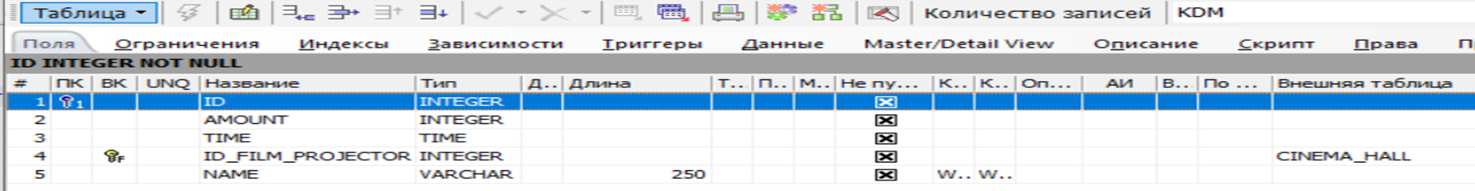
// KMDIndexes:=TstringList.Create;

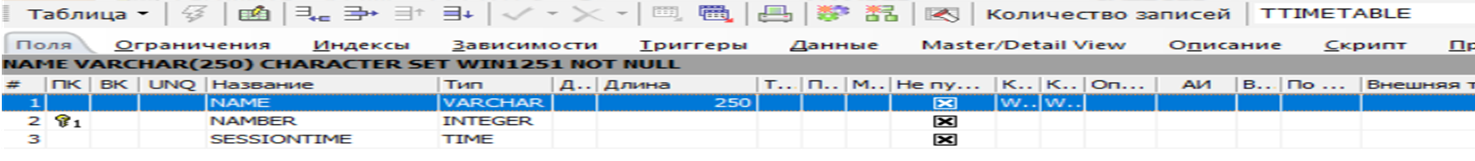
// LoadTable('KDM\_key',nil,FMain.cmbkdm);

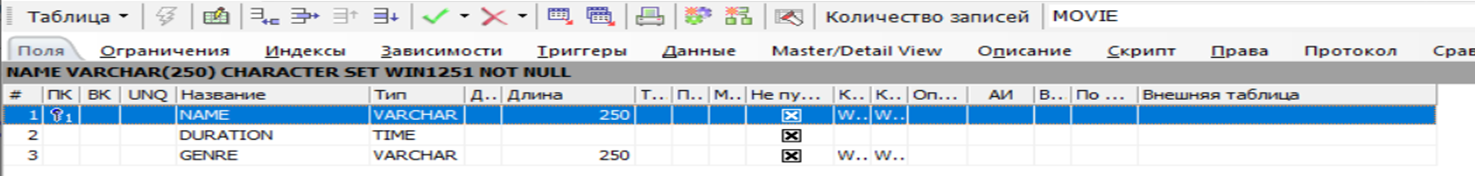
end;

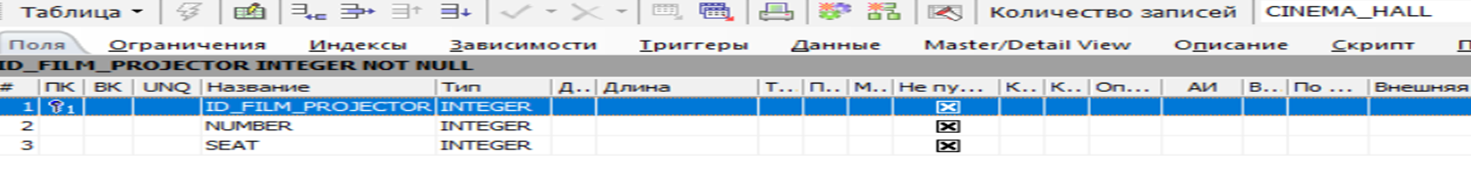
**Структура БД**











**Приложение 1**

**Дневник прохождения производственной практики**

**Записи выполненных работах.**

|  |  |  |  |
| --- | --- | --- | --- |
| Дата | Краткое содержание работ | Отметка о выполнении работы | Подпись руководителя практики |
| 10.03.21 | Изучение предметной области |  |  |
| 12.03.21 | Разработка плана программы |  |  |
| 14.03.21 | Разработка базы данных |  |  |
| 16.03.21 | Разработка интерфейса |  |  |
| 18.03.21 | Проверка правильности работы программы |  |  |
| 20.03.21 | Оформление отчета |  |  |

**Указание о заполнении дневника практики**

* В колонке «Дата» указывается период выполнения работы, изучения материала.
* В колонке «Содержание работ» записываются виды выполняемых студентом работ, наименование тем изучаемого материала.
* Отметку о выполнении работ ставит руководитель практики от предприятия.

**Приложение 2**

**Отзыв-характеристика**

**на студента, обучающегося на базовом уровне по специальности 09.02.03 «Программирование в компьютерных системах»**

(заполняется на рабочем месте непосредственным руководителем)

Ф.И.О. Лихторенко Олеся Сергеевна

Студент обучается на 4 курсе в группе П1-17

Для заполнения отзыва ответьте, пожалуйста, на следующие вопросы:

(Выбранные ответы отметьте в таблице любым доступным способом. Ответов может быть несколько, но дополняющих друг друга)

1. Понимает ли студент-практикант сущность и социальную значимость своей будущей профессии?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Проявляет ли студент-практикант к своей профессии устойчивый интерес?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Способен ли студент-практикант организовать собственную деятельность?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Выбирает ли студент-практикант типовые методы и способы выполнения профессиональных задач?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Оценивает ли студент-практикант эффективность и качество решения различных задач?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Принимает ли студент-практикант решения в стандартных и нестандартных ситуациях?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Может ли студент-практикант нести ответственность за принятые решения?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Осуществляет ли студент-практикант поиск необходимой информации, необходимой для эффективного выполнения профессиональных задач?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Может ли студент-практикант применить необходимую информацию, для эффективного выполнения профессиональных задач?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Повышает ли студент практикант своё личностное и профессиональное развитие?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Владеет ли студент-практикант информационной культурой?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Может ли анализировать студент-практикант информацию с использованием информационно-коммуникационных технологий?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Может ли оценивать студент-практикант информацию с использованием информационно-коммуникационных технологий?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Работал ли успешно студент-практикант в коллективе и в команде?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Как эффективно студент-практикант общался с коллегами, руководством, потребителями?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Берет ли студент-практикант на себя ответственность за работу членов команды (подчиненных) и за результат выполнения заданий?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Может ли студент-практикант самостоятельно определять задачи профессионального и личностного развития?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Может ли студент-практикант заниматься самообразованием?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Может ли студент-практикант осознанно планировать повышение квалификации?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

1. Ориентируется ли студент-практикант в условиях частой смены технологий в профессиональной деятельности?

Ответ: да/нет/интереса не проявляет/проявляет интерес настойчиво/инициативу не проявляет/проявляет инициативу настойчиво

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| №№  вопросов | Ответ:  да | Ответ:  нет | Ответ:  интереса не проявляет | Ответ: проявляет интерес настойчиво | Ответ: инициативу не проявляет | Ответ: проявляет инициативу настойчиво |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |
| 11 |  |  |  |  |  |  |
| 12 |  |  |  |  |  |  |
| 13 |  |  |  |  |  |  |
| 14 |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |
| 16 |  |  |  |  |  |  |
| 17 |  |  |  |  |  |  |
| 18 |  |  |  |  |  |  |
| 19 |  |  |  |  |  |  |
| 20 |  |  |  |  |  |  |

**Приложение 3**

**АТТЕСТАЦИОННЫЙ ЛИСТ ПО ПРАКТИКЕ**

Студент\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, обучающийся на 4 курсе по специальности 09.02.03 «Программирование в компьютерных системах» успешно прошел производственную практику в период с \_\_\_\_\_\_\_\_\_по\_\_\_\_\_\_\_\_\_\_

в организации \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Виды и качество выполнения работ**

|  |  |
| --- | --- |
| Виды и объемы работ, выполненных обучающимся во время практики | Качество выполнения работ в соответствии с технологией и/или требованиями организации, в которой проходила практика |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Характеристика учебной и профессиональной деятельности обучающегося во время производственной практики**

По итогам прохождения практики студент заслуживает оценки-

Руководитель практики \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата инициалы, фамилия

Студент группы \_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

код группы подпись, дата инициалы, фамилия

Королёв, 2021