

СОДЕРЖАНИЕ

Перечень сокращений и обозначений 4

Введение 5

1 Анализ и разработка требований 7

1.1 Назначение и область применения 7

1.2 Постановка задачи 7

1.3 Описание алгоритма функционирования системы 8

1.4 Выбор состава программных и технических средств 10

2 Разработка серверной части информационной системы 13

2.1 Проектирование базы данных 13

2.2 Создание объектов базы данных 16

3 Разработка клиентской части информационной системы 23

3.1 Разработка приложения для доступа к базе данных 23

3.2 Разграничение прав доступа пользователей 24

3.3 Разработка и экспорт отчетов 25

4 Руководство пользователя 27

4.1 Установка ИС 27

4.1.1 Установка серверной части 27

4.1.2 Установка клиентской части 27

4.2 Инструкция по работе 28

4.2.1 Первый запуск приложения 28

4.2.2 Использование вкладки «Сотрудники» 30

4.2.3 Использование вкладки «Система кадров» 32

4.2.4 Использование вкладки «Начисления» 33

4.2.5 Использование вкладки «Отпуска» 34

4.2.6 Использование вкладки «Больничные» 35

4.2.7 Использование вкладки «Настройки» 36

4.2.8 Использование вкладки «Карточка сотрудника» 38

5 Тестирование разработанной системы 40

5.1 Тестирование методом черного ящика 40

6 Определение затрат на разработку ИС для ООО «Кедр» 43

7 Охрана труда и техника безопасности при работе с ПК 54

7.1 Общие требования безопасности 54

7.2 Требования безопасности перед началом работы 54

7.3 Требования безопасности во время работы 55

7.4 Требования охраны труда в аварийных ситуациях 55

7.5 Требования охраны труда по окончанию работы 56

Заключение 57

Список использованных источников 58

**ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ**

В настоящем дипломном проекте применяются следующие сокращения и обозначения:

БД – база данных

ЖКД – жидкокристаллический дисплей

ИС – информационная система

ООО – общество с ограниченной ответственностью

ПК – персональный компьютер

СУБД – система управления базами данных

ЭВМ – электронно-вычислительная машина

ERD – диаграмма «сущность-связь»

LINQ – язык запросов к источнику данных

MS – Microsoft

SQL – язык структурированных запросов

ВВЕДЕНИЕ

На сегодняшний день всё большее количество предприятий и учреждений постепенно переходит к электронному документообороту. Компании хранят на компьютерах данные в виде текстовых документов, таблиц и файлах различных форматов. Вместе с таким переходом у предприятий появляется потребность хранить данные о своих сотрудниках в единой упорядоченной системе, которая бы предоставляла возможность просмотра и обработки данных и разграничивала бы права доступа к ним.

Обеспечить хранение и обработку больших объемов информации можно путем разработки и внедрения автоматизированных ИС [9].

В силу вышесказанного целью дипломного проектирования является разработка ИС для ООО «Кедр». Разрабатываемая система должна позволить хранить все данные о сотрудниках предприятия, начислениях, отпусках и больничных, а также обеспечить возможность добавления, изменения и удаления информации о должностях, отделах, ставках, графиках работы. Помимо этого ИС должна проверять корректность всех данных, вводимых пользователем, для того, чтобы обеспечивать целостность данных.

Для достижения поставленной цели требуется выполнить следующие задачи:

* провести анализ предметной области,
* проанализировать возможные подходы к поставленной задаче,
* проанализировать методы решения поставленной задачи с обоснованием выбранного метода,
* выбрать эффективные алгоритмы с учетом их устойчивости и точности,
* спроектировать модели, необходимые для разработки ИС,
* разработать БД,
* реализовать бизнес-логику на стороне сервера, разработав необходимые представления, функции, процедуры и триггеры в БД,
* спроектировать интерфейс клиентского приложения,
* разработать клиентское приложение,
* реализовать экспорт данных в формате .xls,
* реализовать экспорт данных в формате .docx,
* реализовать отправку сообщений по электронной почте,
* реализовать отправку смс-сообщений,
* провести отладку кода ИС [11],
* провести тестирование ИС,
* проанализировать полученные в ходе тестирования и отладки результаты работы ИС,
* составить руководство пользователя по установке и эксплуатации ИС.

Выполнение всех вышеперечисленных задач должно повлечь за собой создание ИС для ООО «Кедр».

1. **Анализ и разработка требований**
   1. **Назначение и область применения**

Областью применения информационной системы является ООО «Кедр».

Требуется создать систему, которая предоставит менеджерам возможность хранить, модифицировать и выводить на экран информацию о сотрудниках предприятия, отделах, должностях, ставках, графиках работ, начислениях, отпусках и больничных. Разрабатываемая ИС нужна для обеспечения быстрого доступа к информации о сотрудниках, кадровых записях, начислениях, больничных и отпусках, а также для уменьшения затрат времени для сбора необходимой информации о сотрудниках, что повлечет за собой увеличение скорости получения или модификации необходимой информации, и её одновременном обновлении во всей базе.

Сотрудникам предприятия ИС предоставит возможность просматривать информацию о своих начислениях, больничных и отпусках, а также предоставит возможность экспорта своих данных в электронный документ локально, либо в сообщении на электронную почту. Если сотрудник является начальником какого-либо отдела, система предоставит возможность отправки сообщения всем сотрудникам своего отдела в виде смс-сообщения либо сообщения на электронную почту.

* 1. **Постановка задачи**

Требуется спроектировать и разработать серверную и клиентскую части ИС, автоматизирующей хранение, обработку и представление информации о сотрудниках, отделах, должностях, ставках, рабочих графиках, начислениях, отпусках и больничных.

Создание ИС позволит организовать доступ к списку сотрудников, графиков работ, ставок, должностей, отделов, начислений, отпусков и больничных, а также предоставит возможность вести структурированный учет всей информации и выполнять экспорт данных в виде электронных таблиц или текстовых документов [6].

Для достижения этой цели необходимо создать приложение с удобным интерфейсом пользователя и БД.

ИС должна обеспечивать выполнение следующих задач:

* авторизацию по номеру телефона с смс-кодом или паре логин-пароль,
* просмотр информации о сотрудниках,
* предоставление менеджеру возможности модификации данных о сотрудниках, графиках работ, ставках, должностях, отделах, начислениях, отпусках, больничных,
* фильтрацию сотрудников по фамилии, имени, отчеству [3],
* добавление записей о новых сотрудниках, графиках работ, ставках, должностях, отделах, начислениях, отпусках, больничных,
* экспорт списка начислений, больничных или отпусков в формате .xls,
* изменение настроек приложения [10],
* экспорт карточки сотрудника в формате .docx локально и в письме на электронную почту,
* отправка начальником отдела сообщения сотрудникам своего отдела посредством смс-сообщения или отправки письма на электронную почту.

**1.3 Описание алгоритма функционирования системы**

После запуска ИС перед пользователем отображается окно авторизации, предоставляющее возможность авторизации под одной из категорий пользователей (администратор приложения, менеджер, сотрудник). Для различных категорий пользователей после успешной авторизации предлагаются различные интерфейсы пользователей.

Администратору приложения должна быть доступна следующая функциональность:

* настройка параметров системы (редактирование информации о системных пользователях),
* настройка подключения приложения к БД.

Менеджеру должна быть доступна следующая функциональность:

* добавление, удаление или изменение информации о сотрудниках, графиках работ, ставках, должностях, отделах, начислениях, отпусках, больничных,
* добавление информации о новых сотрудниках, графиках работ, ставках, должностях, отделах, начислениях, отпусках, больничных,
* экспорт записей о начислениях, больничных и отпусках локально.

Сотруднику должна быть доступна следующая функциональность:

* поиск информации о сотрудниках по фамилии, имени, отчеству,
* сортировка по фамилии или дате рождения,
* экспорт своей карточки сотрудника локально или посредством электронного письма.
* передача сообщений сотрудникам своего отдела (при условии, что авторизованный пользователь – начальник отдела).

Для наибольшей наглядности на рисунке 1 приведена диаграмма вариантов использования.

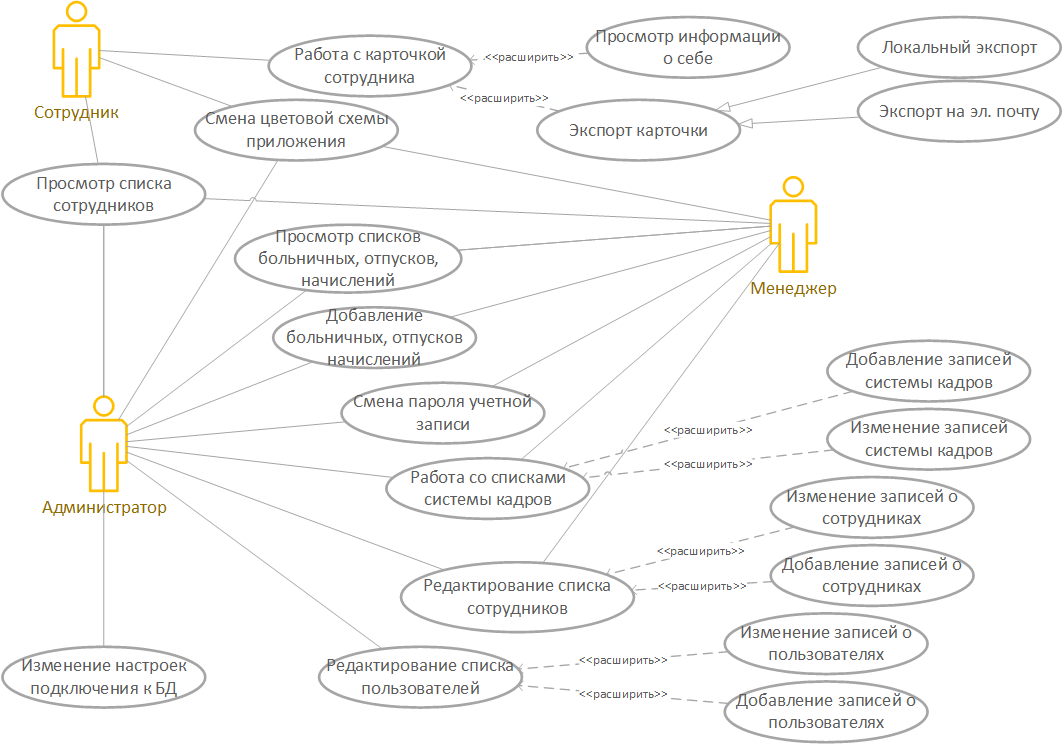


Рисунок 1 – Диаграмма вариантов использования

**1.4 Выбор состава программных и технических средств**

Согласно цели проектирования требуется создать многопользовательскую клиент-серверную информационную систему для ООО «Кедр».

Эксплуатироваться разрабатываемая ИС будет на персональных компьютерах с установленной ОС семейства Windows версии не ниже Windows 10.

В качестве системы управления базами данных выбрана СУБД Microsoft SQL Server 2019 Express, т.к. она является удобной в работе и имеет собственный язык запросов, который оптимален тем, что информацию из БД можно извлекать по любому критерию или совокупности критериев [4]. Также в этой СУБД поддерживается возможность создания различных подпрограмм на языке SQL.

Приложение будет написано на языке программирования C#, т.к. в нем присутствуют простые в применении компоненты для работы с базами данных [20], в частности – с базами данных Microsoft SQL Server [5]. Для разработки приложения будет использоваться интегрированная среда разработки программ Microsoft Visual Studio 2019, т.к. она позволяет достаточно быстро создавать приложения на языке программирования C# для Windows [1]. Для создания визуальных стилей приложения будет использоваться библиотека MaterialSkin, являющаяся свободной для коммерческого и некоммерческого использования. Для отправки смс-сообщений из приложения будет осуществляться через смс-шлюз, предоставляемый сервисом SmsAero.

Для функционирования системы на стороне сервера достаточны следующие программные и технические средства:

* ОС Windows 10 TH1 1507 или выше, либо Windows Server 2016 или выше,
* сервер БД: Microsoft SQL Server версии не ниже 2019 года,
* программное обеспечение для конфигурирования, управления и администрирования MSSQL: SQL Server Management Studio 18 или выше,
* процессор Intel или совместимый процессор с тактовой частотой 1,4 ГГц и выше (рекомендуется 2 ГГц и выше),
* оперативная память минимум 512 МБ (рекомендуется 2 ГБ и выше),
* объем свободного места на жестком диске не менее 6 ГБ,
* манипуляторы: клавиатура и мышь,
* компьютерный монитор: ЖКД с диагональю не менее 21".

Для функционирования системы на стороне клиента достаточны следующие программные и технические средства:

* операционная система Windows 7 пакетом обновления 1 (SP1) или выше,
* .Net Framework версии 4.8 и выше,
* процессор Intel или совместимый процессор с тактовой частотой 1 ГГц и выше (рекомендуется 2 ГГц и выше),
* оперативная память минимум 1024 МБ (рекомендуется 2 ГБ и выше),
* объем свободного места на жестком диске не менее 4 ГБ,
* компьютерный монитор: ЖКД с диагональю не менее 21",
* манипуляторы: клавиатура и мышь.

**2 Разработка серверной части информационной системы**

**2.1 Проектирование базы данных**

В БД требуется хранить информацию о сотрудниках (код сотрудника, имя, фамилия, отчество, дата рождения, путь к фотографии, номер телефона, e-mail адрес), карточках сотрудника (номер карточки, сотрудник, заработная плата, должность, отдел, ставка, график работы), должностях (номер должности, название должности), отделах (номер отдела, название отдела), ставках (номер ставки, наименование ставки), графиках работ (номер графика, наименование графика), начислениях (номер начисления, тип начисления, сотрудник, размер начисления, дата начисления), типы начислений (номер типа начисления, название типа начисления), больничных (номер больничного, сотрудник, дата начала больничного, дата конца больничного), отпусках (номер отпуска, сотрудник, дата начала отпуска, дата конца отпуска), приемах на работу (номер приема, сотрудник, дата приема), увольнениях (номер увольнения, сотрудник, дата увольнения, причина увольнения), пользователях системы (номер пользователя, логин, пароль, тип пользователя) и типах пользователей (номер типа пользователя, название типа пользователя).

В ходе анализа предметной области выявлены следующие её особенности:

* в таблице должности не могут находиться две должности с одинаковым названием,
* в таблице ставки не могут находиться две ставки с одинаковым названием,
* в таблице отделы не могут находиться два отдела с одинаковым названием,
* в таблице графики не могут находиться два графика с одинаковым названием,
* при внесении больничных в таблицу нужно учитывать то, что дата начала больничного не может быть больше текущей даты, а конец больничного не может быть раньше, чем его начало,
* при внесении больничных в таблицу отпусков учитывать то, что конец отпуска не может быть раньше, чем его начало,
* дата начисления должна устанавливаться автоматически по системному времени ПК,
* экспорт карточки сотрудника на электронную почту возможен только при наличии у такого сотрудника адреса в БД.

На рисунке 2 показана концептуальная модель предметной области в виде ERD.

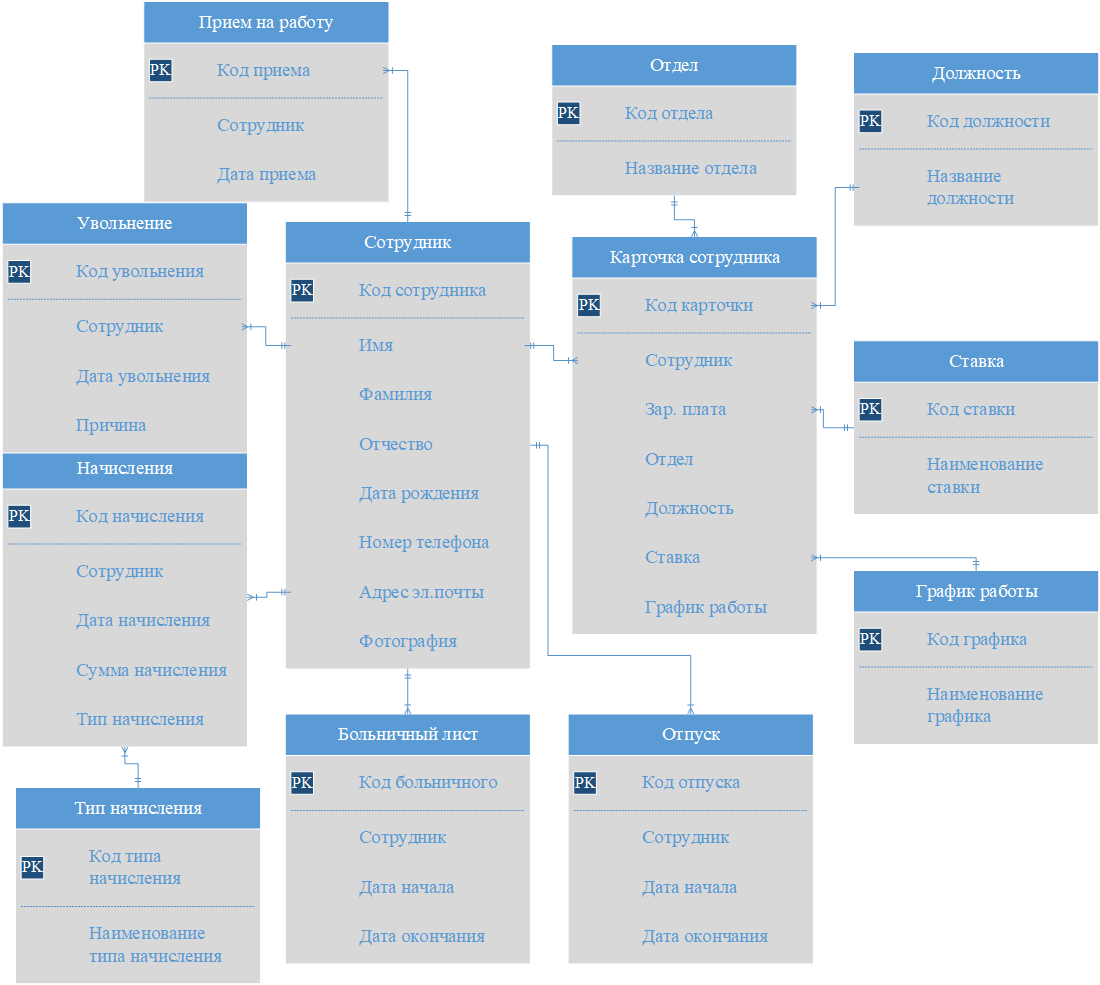


Рисунок 2 – Концептуальная модель

На рисунке 3 показана логическая модель предметной области, полученная путем преобразования концептуальной модели [7].

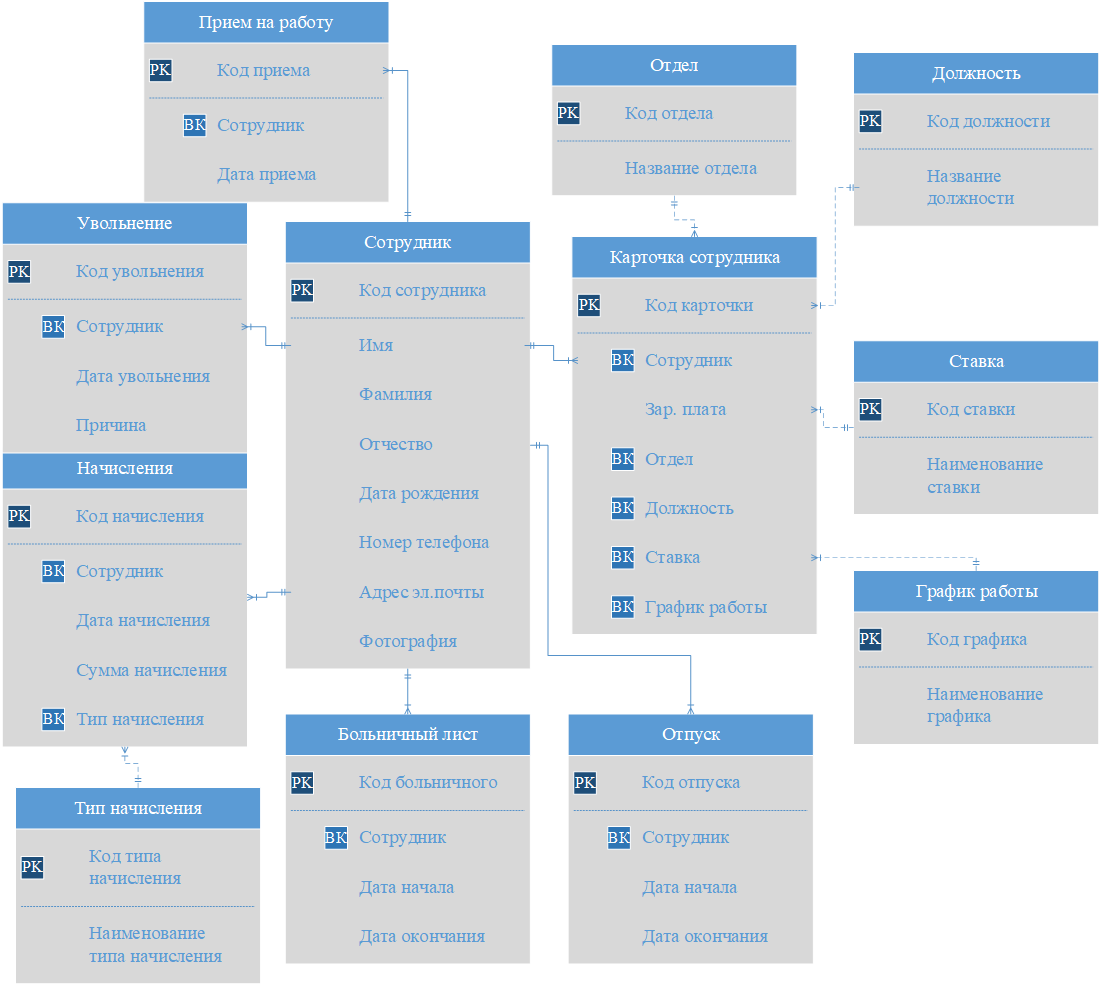


Рисунок 3 – Логическая модель

На рисунке 4 показана физическая модель предметной области, разработанная на основе логической модели для СУБД MS SQL Server 2019.

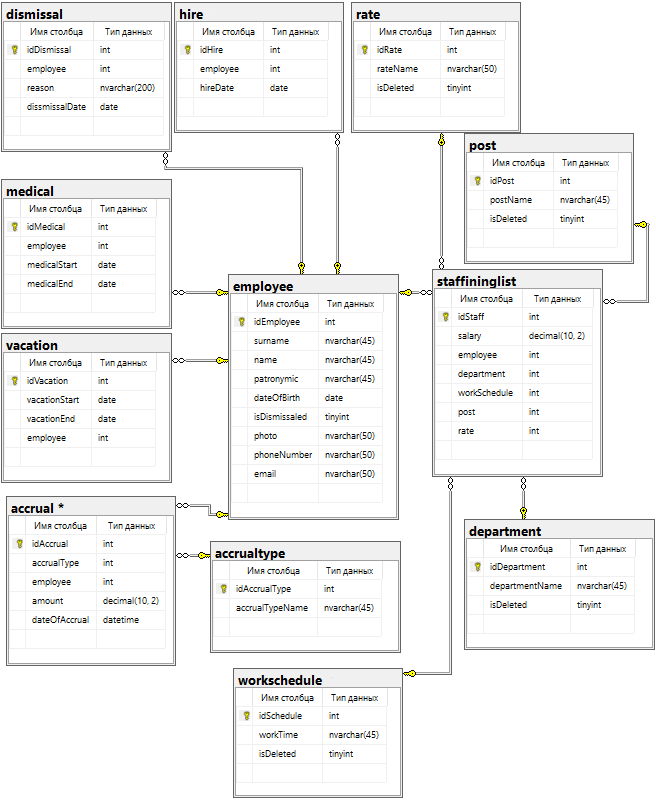


Рисунок 4 – Физическая модель

**2.2 Создание объектов базы данных**

В таблице 1 в виде словаря данных представлено описание созданных таблиц и ограничений целостности БД.

Таблица 1 – Словарь данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
| employee | | | | |
| PK | idEmployee | int | + | Автоинкремент |
|  | surname | nvarchar(50) | + | Не может содержать цифры |
|  | name | nvarchar(50) | + | Не может содержать цифры |
|  | patronymic | nvarchar(50) | - | Не может содержать цифры |
|  | dateOfBirth | date | + | не больше GetDate() |
|  | phone | nvarchar(11) | - | Содержит только один символ и цифры |
|  | email | nvarchar(50) | - |  |
| staffingList | | | | |
| PK | idStaffing | int | + | Автоинкремент |
| FK1 | employee | int | + |  |
| FK2 | department | int | + |  |
| FK3 | post | int | + |  |
| FK4 | rate | int | + |  |
| FK5 | workTime | int | + |  |
|  | salary | int | + | Не может быть отрицательным |
| department | | | | |
| PK | idDepart | int | + | Автоинкремент |
|  | nameDepart | varchar(60) | + | Уникальное |
| post | | | | |
| PK | idPost | int | + | Автоинкремент |
|  | namePost | varchar(60) | + | Уникальное |
| workSchedule | | | | |
| PK | idWorkTime | int | + | Автоинкремент |
|  | schName | varchar(60) | + | Уникальное |

Продолжение таблицы 1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ключ | Поле | | | Тип Данных | | Обязательное | Примечание | |
| accrual | | | | | | | | |
| PK | | idAccrual | int | | + | | | Автоинкремент |
| FK | | employee | int | | + | | |  |
|  | | amount | decimal (9,2) | | + | | |  |
|  | | date | datetime | | + | | | По ум. GetDate() |
| vacation | | | | | | | | |
| PK | | idVacation | int | | + | | | Автоинкремент |
| FK | | employee | int | | + | | |  |
|  | | startTime | date | | + | | |  |
|  | | endTime | date | | + | | |  |
| medical | | | | | | | | |
| PK | | idMedical | int | | + | | | Автоинкремент |
| FK | | employee | int | | + | | |  |
|  | | startTime | date | | + | | |  |
|  | | endTime | date | | + | | |  |
| hire | | | | | | | | |
| PK | | idHire | int | | + | | | Автоинкремент |
| FK | | employee | int | | + | | |  |
|  | | hireDate | date | | + | | | По ум. GetDate() |
| dismissal | | | | | | | | |
| PK | | idDismissal | int | | + | | | Автоинкремент |
| FK | | employee | int | | + | | |  |
|  | | dismissDate | date | | + | | | По ум. GetDate() |
|  | | reason | nvarchar(200) | |  | | |  |

Для решения поставленных задач в БД созданы следующие триггеры:

– триггер «onEmployeeHire», выполняющийся после записи в таблице «Employee», который добавляет запись в таблицу «Hire» с кодом сотрудника и текущей датой (код создания триггера представлен листингом 1),

– триггер «onDepartmentDelete», выполняющийся вместо удаления записи из таблицы «Department», устанавливает значение 1 в столбец «isDeleted» (код создания триггера представлен листингом 2).

Листинг 1 – Код создания триггера onEmployeeAdd

|  |
| --- |
| /\*Код создания триггера onEmployeeAdd \*/  CREATE TRIGGER [dbo].[onEmployeeAdd]  ON [dbo].[Employee] --для таблицы Employee  AFTER INSERT --После добавления записи  AS  BEGIN  INSERT Hire(Employee) SELECT IDENT\_CURRENT('employee')  --Вставить новую запись в таблицу Hire |

Листинг 2 – Код создания триггера onDepartmentDelete

|  |
| --- |
| /\* код создания триггера onDepartmentDelete \*/  CREATE TRIGGER [dbo].[ onDepartmentDelete]  ON [dbo].[Department]-- для таблицы Product  INSTEAD OF DELETE -- вместо удаления  AS  UPDATE Department --обновление таблицы Department  --установить значение поля isDeleted, равное единице  SET isDeleted = 1  --где код удаляемого отдела равен коду из таблицы  --«Department»  WHERE idDepart =(SELECT idDepart FROM deleted) |

Для решения поставленных задач в БД созданы следующие представления [16]:

– представление «FullEmployeeInfo», возвращающее информацию о сотрудниках, их отделах, должностях, ставках, графиках, зарплатах (код представлен листингом 3),

– представление «ViewEmployeesArccruals», возвращающее начисления всех сотрудников на основе данных из таблиц «Arrcual», «Employee» и «AccrualType»,

– представление «ViewEmployeesMedicals», возвращающее больничные всех сотрудников на основе данных из таблиц «Medical» и «Employee»,

– представление «ViewEmployeesVacations», возвращающее отпуска всех сотрудников на основе данных из таблиц «Vacation» и «Employee».

Листинг 3 – Код создания представления FullEmployeeInfo

|  |
| --- |
| -- код создания представления fullemployeeinfo  CREATE VIEW [dbo].[fullemployeeinfo] AS  --Выбрать номер сотрудника, имя, фамилию, отчество, дату  --рождения, фото, должность, отдел, график, ставку, номер  --телефона, адрес эл.почты, зарплату, пометку об увольнении  SELECT dbo.staffininglist.Employee AS Номер,  dbo.employee.Name AS Имя,  dbo.employee.Surname AS Фамилия,  dbo.employee.Patronymic AS Отчество,  dbo.employee.DateOfBirth AS [Дата рождения],  dbo.employee.photo AS Фото,  dbo.post.PostName AS Должность,  dbo.department.DepartmentName AS Отдел,  dbo.workschedule.WorkTime AS График,  dbo.rate.RateName AS Ставка,  dbo.employee.phoneNumber AS Телефон,  dbo.staffininglist.Salary AS Зарплата,  dbo.employee.isDismissaled AS Уволен,  dbo.employee.email AS [Эл.почта]  FROM dbo.staffininglist –- из таблицы staffinglist  INNER JOIN dbo.employee -– присоединение таблицы employee  -- где код сотрудника равен коду сотрудника из employee  ON dbo.staffininglist.Employee = dbo.employee.idEmployee  INNER JOIN dbo.post –-присоединение таблицы post  --где номер должности равен номеру из post  ON dbo.staffininglist.Post = dbo.post.idPost  INNER JOIN dbo.department-—присоединение таблицы department  --где код отдела равен коду из department  ON dbo.staffininglist.Department =  dbo.department.idDepartment  INNER JOIN dbo.workschedule—-присоединение workschedule  --где код графика равен коду из workschedule  ON dbo.staffininglist.WorkSchedule =  dbo.workschedule.idSchedule  INNER JOIN dbo.rate—-присоединение таблицы rate где код ставки равен коду из rate  ON dbo.staffininglist.Rate = dbo.rate.idRate |

Для решения поставленных задач в БД созданы следующие функции пользователя:

– скалярная функция «TotalEmployeeAccruals», возвращающая сумму всех начислений для сотрудника, код которого передан в параметрах (код создания функции представлен листингом 4),

– табличная функция «GetListOfEmployeesDepartment», возвращающая таблицу сотрудников по номеру отдела, переданному в параметрах,

– табличная функция «GetListOfEmployeesRate», возвращающая таблицу сотрудников по номеру ставки, переданному в параметрах,

– табличная функция «GetListOfEmployeestPost», возвращающая таблицу сотрудников по номеру должности, переданному в параметрах,

– табличная функция «GetListOfEmployeesWorkTime», возвращающая таблицу сотрудников по номеру графика, переданному в параметрах,

Листинг 4 – Код создания функции пользователя TotalEmployeeAccruals

|  |
| --- |
| -- код создания функции пользователя TotalEmployeeAccruals  CREATE FUNCTION TotalEmployeeAccruals  (  @idEmployee int --параметр код сотрудника  )  RETURNS decimal(18,2)-- возвращаемый тип  AS  BEGIN  RETURN SUM((SELECT [Accrual].amount)--выбрать сумму  --начисления  FROM [Accrual]--из таблицы начисления  -- где код сотрудника равен параметру  WHERE [Accrual].Employee = @ idEmployee)  END; |

Для решения поставленных задач в БД созданы следующие хранимые процедуры:

– хранимая процедура «addAccrual», добавляющая начисление на основе переданной в параметрах суммы, типа начисления и номера сотрудника (код создания хранимой процедуры представлен листингом 5),

– хранимая процедура «addMedical», добавляющая запись о больничном в таблицу «medical» на основе кода сотрудника, даты начала и конца больничного, переданных в параметрах,

– хранимая процедура «addVacation», добавляющая запись об отпуске в таблицу «vacation» на основе кода сотрудника, даты начала и конца отпуска, переданных в параметрах.

Листинг 5 – Код создания хранимой процедуры addAccrual

|  |
| --- |
| -- код создания хранимой процедуры addAccrual  CREATE PROCEDURE [dbo].[ addAccrual]  @amount DECIMAL(7,2),@idEmployee int, @accrualType int  AS  BEGIN  --вставить в таблицу Accrual  INSERT INTO [Accrual](employee, accrualType, amount)  Values (@idEmployee,@accrualType,@amount);  --значения номера сотрудника  --типа начисления и суммы  END; |

**3 Разработка клиентской части ИС**

**3.1 Разработка приложения для доступа к БД**

В ходе дипломного проектирования разработано оконное приложение WinForms с использованием технологии EntityFramework [2]. Строка настроек подключения к БД может быть изменена с помощью учетной записи администратора через соответствующую форму [8].

Получение данных реализовано через класс Core (листинг 6), который получает данные таблицы из БД и приводит их к соответствующему классу в приложении.

Листинг 6 – Код класса Core

|  |
| --- |
| // код создания класса Core  class Core  {  //объявление поля context  private static companyEntities context;  private Core() { }  //свойство для поля context  public static bakeryEntities Context  {  get  {  if (context == null)//если context равен null  {  //переопределение поля context  context = new companyEntities();  }  return context;//вернуть context  }  }  } |

Фильтрация реализована с использованием LINQ [15] в методе FilterEmployees. Код метода представлен в листинге 7.

Листинг 7 – Код метода FilterEmployees

|  |
| --- |
| /\*код метода фильтрации сотрудников на основе полученного в параметрах текста\*/  private void FilterEmployees(string searchText)  {  //приведение искомого текста к нижнему регистру  searchText = searchText.ToLower();  //объявление списка отфильтрованных сотрудников  List<fullemployeeinfo> filteredEmployees;  //если требуется отображать с уволенными сотрудниками  if (WithDissmissaled)  {  /\*отфильтрованные сотрудники приравниваются к сотрудникам, имя, фамилия или отчество которых содержит искомый текст\*/  filteredEmployees = employees  .Where(n => n.Фамилия.ToLower().Contains(searchText)  ||n.Имя.ToLower().Contains(searchText)  ||n.Отчество.ToLower().Contains(searchText))  .ToList();  }  //иначе  else  {  /\*отфильтрованные сотрудники приравниваются к сотрудникам, имя, фамилия или отчество которых содержит искомый текст и отсутствует отметка об увольнении равна 0\*/  filteredEmployees = employees  .Where(n => n.Уволен == 0 && (n.Фамилия.ToLower().Contains(searchText)  || n.Имя.ToLower().Contains(searchText)  || n.Отчество.ToLower().Contains(searchText)))  .ToList();  }  //вызов метода вывода отфильтрованного списка на форму  SetEmployeesToPanel(filteredEmployees);  } |

**3.2 Разграничение прав доступа пользователей**

Разграничение прав доступа выполнено посредством создания таблиц «Пользователи» и «Роли» [13]. Схема таблиц приведена на рисунке 5. Типы данных столбцов в таблицах «Пользователи» и «Роли» представлены в таблице 2.

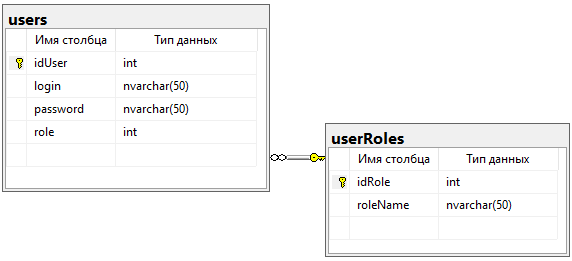


Рисунок 5 – Схема таблиц пользователей и их ролей

Таблица 2 – Словарь данных таблиц User и UserRole

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ключ | Поле | Тип данных | Обязательное | Примечание |
| users | | | | |
| PK | idUser | int | + | Автоинкремент |
| FK | role | int | + |  |
|  | login | nvarchar(50) | + | Уникальное |
|  | password | nvarchar(50) | + |  |
| userRoles | | | | |
| PK | idRole | int | + | Автоинкремент |
| FK | roleName | nvarchar(50) | + | Уникальное |

На уровне интерфейса реализовано окно авторизации, в котором требуется ввести логин и пароль от пользователя либо его номер телефона и код, высланный в смс. При успешной авторизации пользователь получает доступ к работе с приложением [14]. В зависимости от роли пользователя разграничивается доступ к окнам и кнопкам.

**3.3 Разработка и экспорт отчетов**

В клиентском приложении реализован экспорт списков начислений, отпусков и больничных в формате .xls. Экспорт выполняется локально через диалоговое окно сохранения файла. Также реализован экспорт карточки сотрудника в формате .docx. Экспорт выполняется локально либо отправляется посредством отправки письма на электронную почту сотрудника. Код метода отправки сообщения на электронную почту представлен листингом 9.

Листинг 9 – Код метода отправки сообщения

|  |
| --- |
| public static bool SendMail(string role,//отображаемое имя  string message,//сообщение  string theme, //тема письма  string addressee,//адрес получателя  //путь к прикрепляемому файлу  string filename = null)  {  try //блок защищенного кода  {  // установка адреса и отображаемого в письме имени  MailAddress from =  new MailAddress("kedrthecompany@gmail.com", role);  // объект получателя  MailAddress to = new MailAddress(addressee);  // создание объекта сообщения  MailMessage m = new MailMessage(from, to);  m.Subject = theme; // тема письма  m.Body = message; // текст письма  if (filename != null)//если файл прикреплен  {  //Добавление файла во вложения письма  m.Attachments.Add(new Attachment(filename));  }    /\* адрес smtp-сервера и порт, с которого отправляется письмо\*/  SmtpClient smtp = new SmtpClient("smtp.gmail.com", 587);  // логин и пароль  smtp.Credentials = new NetworkCredential("kedrthecompany@gmail.com", "\*\*\*\*\*\*\*");  //установка защищенного протокола  smtp.EnableSsl = true;  smtp.Send(m); //отправка письма  return true; //вернуть результат отправки true  }  catch (Exception)//Блок перехвата исключений  {  return false;//возвращает false в случае ошибки  }  } |

1. **Руководство пользователя**
   1. **Установка ИС**
      1. **Установка серверной части**

Для установки серверной части необходимо установить сервер и выполнить SQL-скрипт [19]. Для этого необходимо:

* скачать MS SQL Server 2019 Express Editions with Tools,
* выполнить установку сервера БД,
* запустить SQL Server Management Studio,
* войти под пользователем с ролью «dbmanager»
* запустить «script.sql» из папки с файлами программы.
  + 1. **Установка клиентской части**

Для установки клиентской части приложения необходимо в папке с файлами программы (рисунок 6) запустить программу установки приложения, согласиться с условиями использования приложения, выбрать директорию для установки приложения и дождаться конца установки.

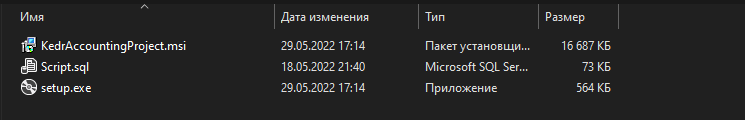


Рисунок 6 – Вид папки с инсталляционным пакетом «Setup»

* 1. **Инструкция по работе**
     1. **Первый запуск приложения**

При открытии формы авторизации может возникнуть проблема с входом в приложение в том случае, если не удается подключиться к БД. Если приложение вывело сообщение об ошибке (рисунок 7), требуется задать параметры строки подключения к БД. Для настройки подключения к БД следует нажать на кнопку настройки в левом нижнем углу формы авторизации и ввести новые данные для подключения к базе данных (рисунок 9). Подтвердить внесенные изменения, нажав на кнопку «Сохранить». Для того, чтобы проверить правильность введенных данных на форме настроек, следует нажать на кнопку «Проверить подключение». В случае успешной проверки приложение выведет соответствующее сообщение (рисунок 8).

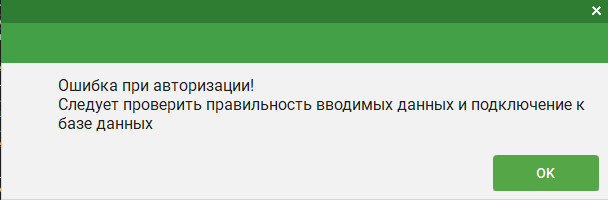


Рисунок 7 – ИС ООО Кедр. Вид окна с сообщением ошибки подключения

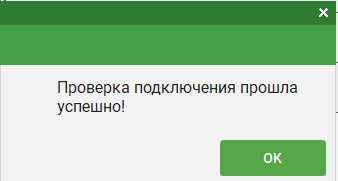


Рисунок 8 – ИС ООО Кедр. Вид окна с сообщением об успешной проверке подключения

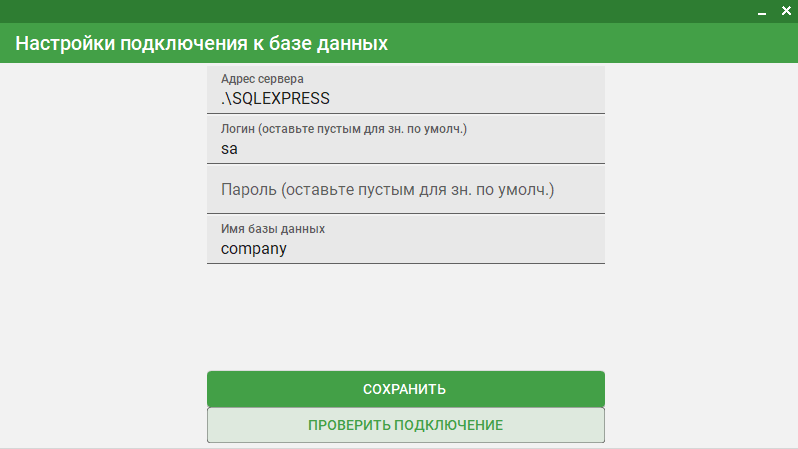


Рисунок 9 – ИС ООО Кедр. Вид окна изменения настроек подключения

Если все настройки выполнены правильно и данные для входа введены верно, то при нажатии кнопки «Войти» на форме авторизации приложение выведет сообщение с приветствием, после этого откроется главная форма приложения. Главное окно для роли администратора представлено на рисунке 10.

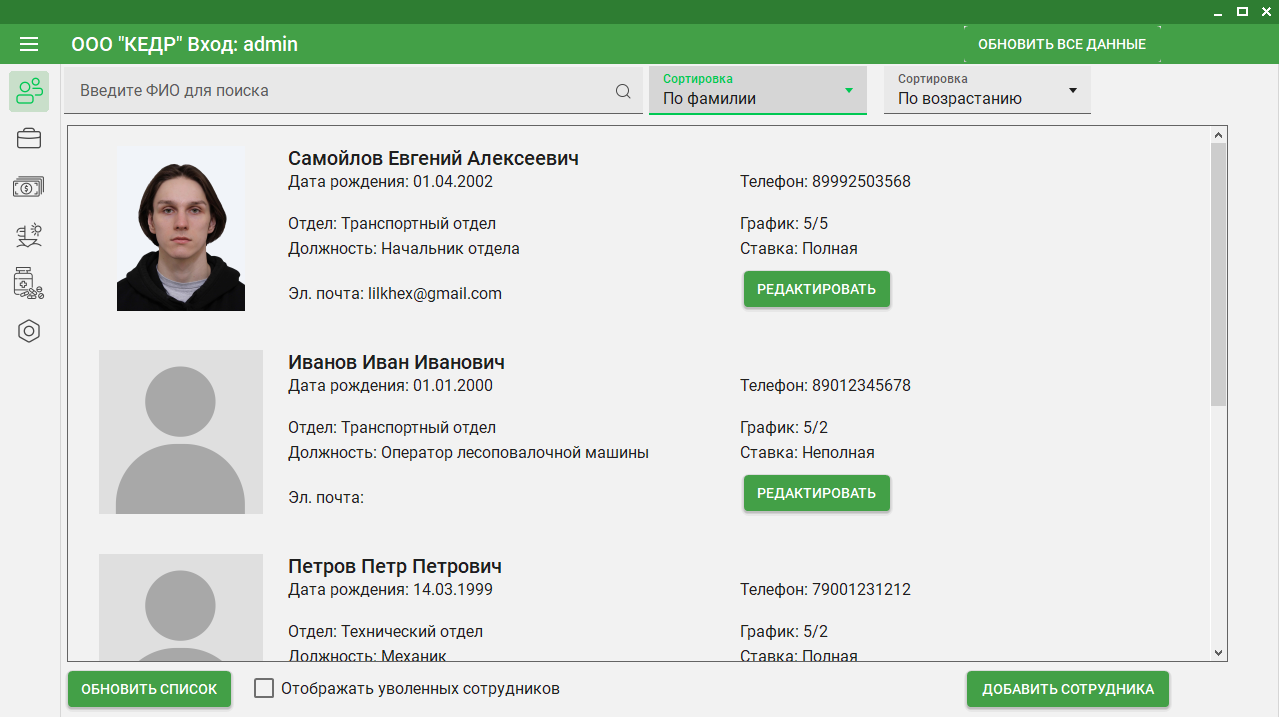


Рисунок 10 – ИС ООО Кедр. Вид главного окна приложения для роли администратора

* + 1. **Использование вкладки «Сотрудники»**

Для работы с сотрудниками следует в левом выпадающем меню выбрать вкладку «Сотрудники». Для поиска записи о сотруднике в базе по фамилии, имени или отчеству следует начать вводить искомый текст в текстовое поле в верхней части формы.

Для сортировки записей о сотрудниках по фамилии в алфавитном порядке или дате рождения следует выбрать соответствующий пункт в выпадающем меню «Сортировка». Для смены порядка сортировки следует выбрать соответствующий пункт в выпадающем меню «Порядок сортировки».

Для того, чтобы в списке отображались записи об уволенных сотрудниках, следует поставить пункт «Отображать уволенных сотрудников» внизу вкладки.

Для добавления новой записи о сотруднике следует нажать на кнопку «Добавить сотрудника», в правом нижнем углу окна. После этого откроется форма добавления записи о сотруднике, после заполнения которой нужно нажать на кнопку «Сохранить» для того, чтобы запись сохранилась в базе. Вид окна добавления записи о сотруднике представлен на рисунке 11.

Для редактирования записи о сотруднике следует нажать на кнопку «Редактировать», которая находится в строке с информацией о сотруднике. Вид окна редактирования сотрудника представлен на рисунке 12. После открытия окна редактирования можно выполнять модификацию данных о сотруднике. При вводе некорректных данных приложение установит значок ошибки напротив соответствующего поля для ввода и не даст сохранить информацию, пока все ошибки ввода не будут исправлены.

Для того, чтобы присвоить записи пометку об увольнении, следует нажать на кнопку «Уволить», которая находится в правом нижнем углу окна.

Для восстановления записи о сотруднике следует нажать на кнопку «Восстановить» в правом нижнем углу при условии, что запись о нем содержит пометку об увольнении.

Для выбора новой фотографии для карточки сотрудника следует нажать на кнопку «Выбрать фото», которая находится под текущей фотографией сотрудника. Если у записи сотрудника не установлена фотография карточки, то будет использовано фото по умолчанию.

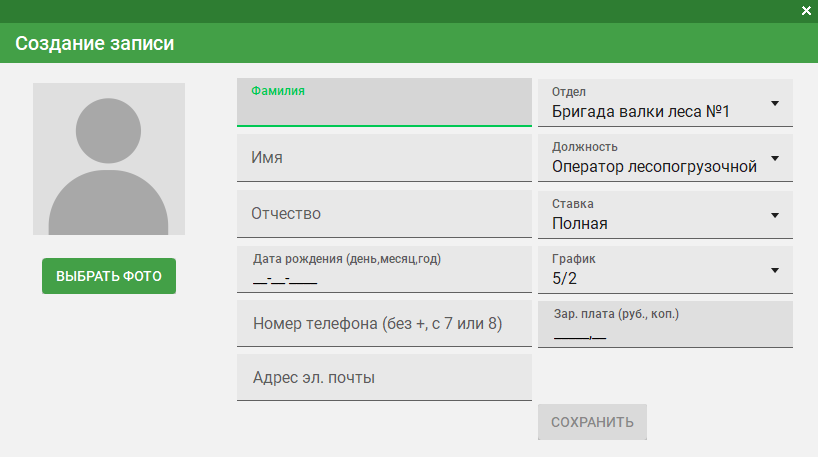


Рисунок 11 – ИС ООО Кедр. Вид окна «Создание записи»

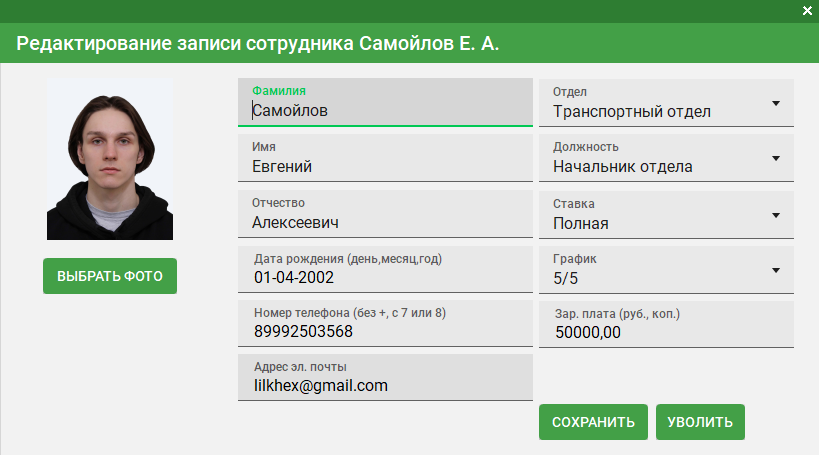


Рисунок 12 – ИС ООО Кедр. Вид окна редактирования записи о сотруднике

* + 1. **Использование вкладки «Система кадров»**

Для работы с записями об отделах, должностях, ставках, графиках работ предусмотрена вкладка «Система кадров». Для того, чтобы добавить запись об отделе, должности, ставке или графику, нужно нажать на кнопку «Добавить» под соответствующим списком. При этом откроется окно добавления записи системы кадров, вид окна представлен на рисунке 13.

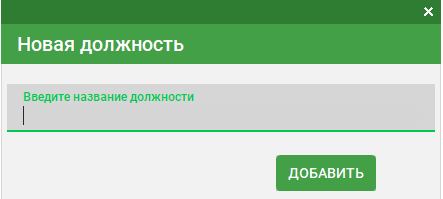


Рисунок 13 – ИС ООО Кедр. Вид окна добавления новой записи

Вид окна с открытой вкладкой «Система кадров» представлен на рисунке 14.

Для того, чтобы редактировать или удалить соответствующую запись требуется выбрать нужный элемент из списка и вызвать контекстное меню правой кнопкой мыши. Из контекстного меню можно выбрать пункт «Редактировать» либо «Удалить».

При удалении под элементом списка будет подпись «Больше не используется».

Для восстановления записи следует выбрать пункт «Редактировать» из контекстного меню и снять флажок «Не используется». При этом нужно сохранить отредактированную запись, нажав на кнопку «Сохранить» в нижнем правом углу окна редактирования.

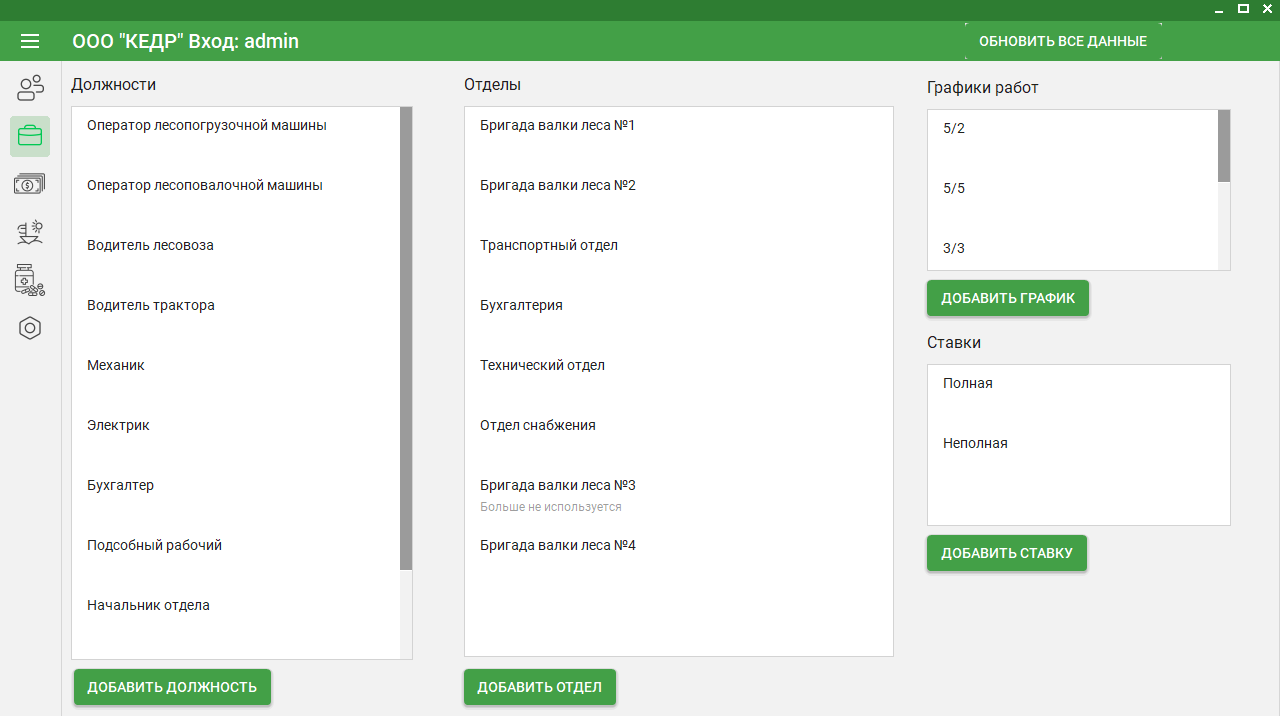


Рисунок 14 – ИС ООО Кедр. Вид окна с открытой вкладкой

«Система кадров»

* + 1. **Использование вкладки «Начисления»**

Для просмотра записей о начислениях и для добавления новых используется вкладка «Начисления». Вид окна с открытой вкладкой «Начисления» показан на рисунке 15.

Для того, чтобы добавить запись о новом начислении, нужно выбрать соответствующую запись о сотруднике из выпадающего списка справа на вкладке. В поле «Сумма начисления» следует указать необходимую сумму начисления, далее выбрать из выпадающего списка «Тип начисления» соответствующий тип начисления и нажать на кнопку «Добавить запись о начислении». При успешном добавлении записи приложение выведет соответствующее сообщение. Для того, чтобы экспортировать список начислений в формат .xls, нужно нажать на кнопку «Экспорт таблицы». В открывшемся окне нужно выбрать место для сохранения таблицы и указать название файла. При успешном экспорте приложение выдаст соответствующее сообщение.

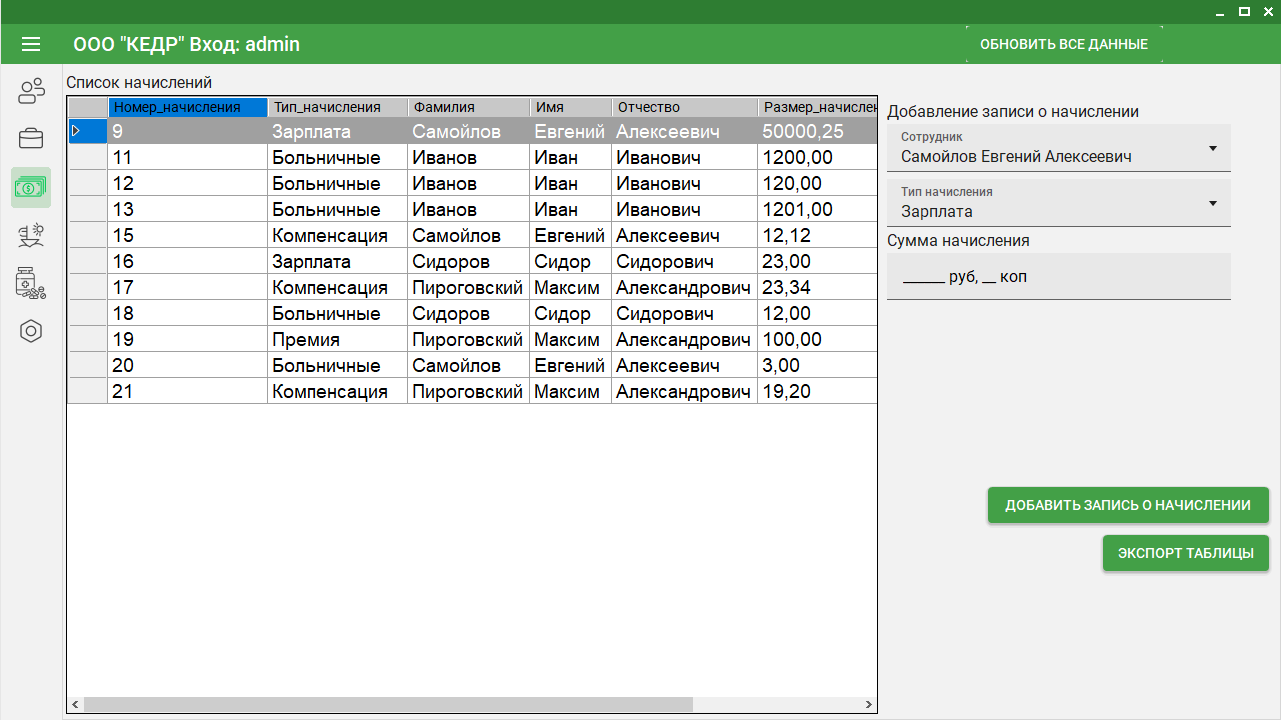


Рисунок 15 – ИС ООО Кедр. Вид окна с открытой вкладкой «Начисления»

**4.2.5 Использование вкладки «Отпуска»**

Для работы с записями об отпусках предусмотрена вкладка «Отпуска». Вид окна с открытой вкладкой представлен на рисунке 16.

Для того, чтобы добавить запись об отпуске, нужно из выпадающего списка «Сотрудники» выбрать соответствующую запись о сотруднике, далее выбрать дату начала отпуска в поле «Начало отпуска», дату окончания отпуска в поле «Конец отпуска» и нажать на кнопку «Добавить запись об отпуске». В случае успешного добавления записи приложение выведет соответствующее сообщение.

Для того, чтобы экспортировать список отпусков в формат .xls, нужно нажать на кнопку «Экспорт таблицы». В открывшемся окне выбрать место для сохранения таблицы и указать название файла. При успешном экспорте приложение выдаст соответствующее сообщение.

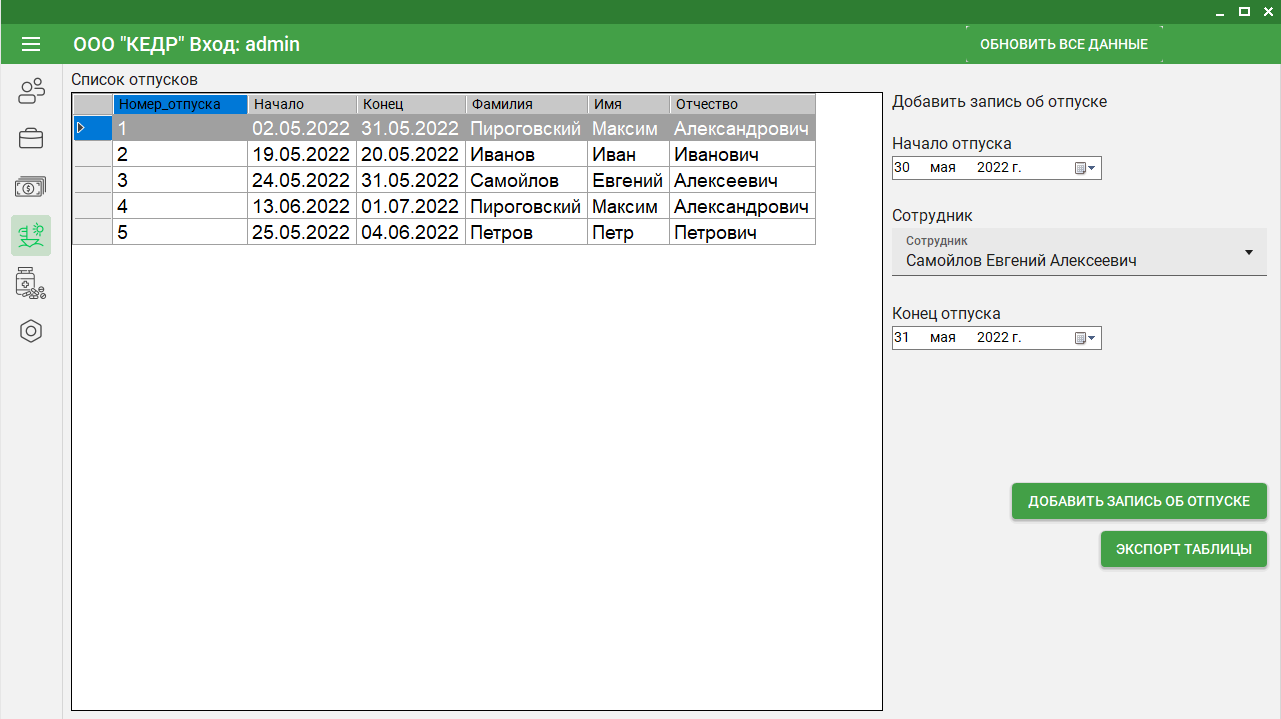


Рисунок 16 – ИС ООО Кедр. Вид окна с открытой вкладкой «Отпуска»

**4.2.6 Использование вкладки «Больничные»**

Для работы с больничными предусмотрена вкладка «Отпуска». Вид окна с открытой вкладкой представлен на рисунке 17.

Для того, чтобы добавить запись об отпуске, нужно из выпадающего списка «Сотрудники» выбрать соответствующую запись о сотруднике, далее выбрать дату начала больничного в поле «Начало больничного», дату окончания больничного в поле «Конец больничного» и нажать на кнопку «Добавить запись о больничном». В случае успешного добавления записи приложение выведет окно с соответствующим сообщением.

Для того, чтобы экспортировать список больничных в формат .xls, нужно нажать на кнопку «Экспорт таблицы». В открывшемся окне выбрать место для сохранения таблицы и указать название файла. При успешном экспорте приложение выдаст соответствующее сообщение.

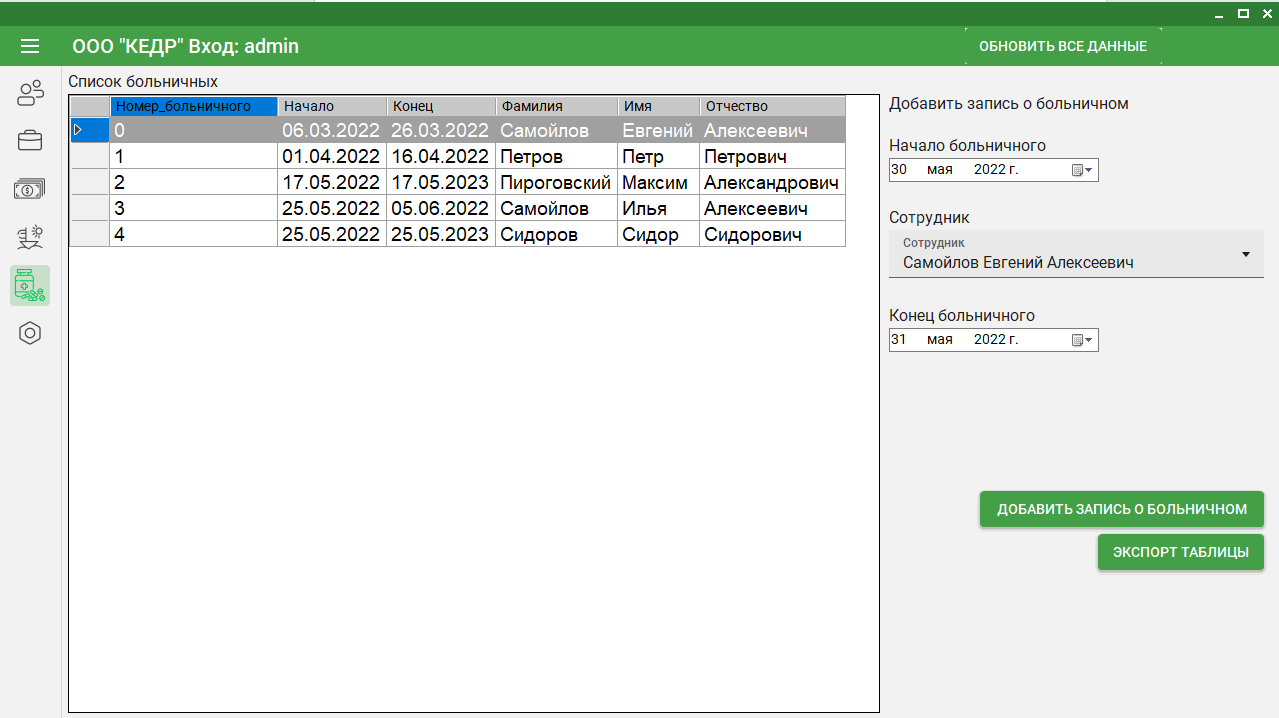


Рисунок 17 – ИС ООО Кедр. Вид окна с открытой вкладкой «Больничные»

**4.2.7 Использование вкладки «Настройки»**

Для того, чтобы перейти во вкладку «Настройки», следует выбрать соответствующий пункт из выпадающего меню слева. При переходе в раздел настроек их количество различается в зависимости от того, под каким типом учетной записи выполнена авторизация. Для сотрудника доступны следующие настройки:

* «Темная тема», используется для изменения внешнего вида приложения,
* «Сменить пользователя», используется для смены пользователя.

Для администратора доступны следующие настройки:

* «Темная тема», используется для изменения внешнего вида приложения,
* «Управление пользователями», используется для редактирования данных пользователей ИС,
* «Настройка базы данных», используется для смены настроек подключения к базе данных,
* «Сменить пароль», используется для смены пароля текущего пользователя.
* «Сменить пользователя», используется для смены пользователя.

Для менеджера доступны следующие настройки:

* «Сменить пользователя», используется для смены пользователя,
* «Темная тема», используется для изменения внешнего вида приложения,
* «Сменить пароль», используется для смены пароля текущего пользователя.

Вид окна с разделом «Настройки» для учетной записи администратора представлен на рисунке 18.

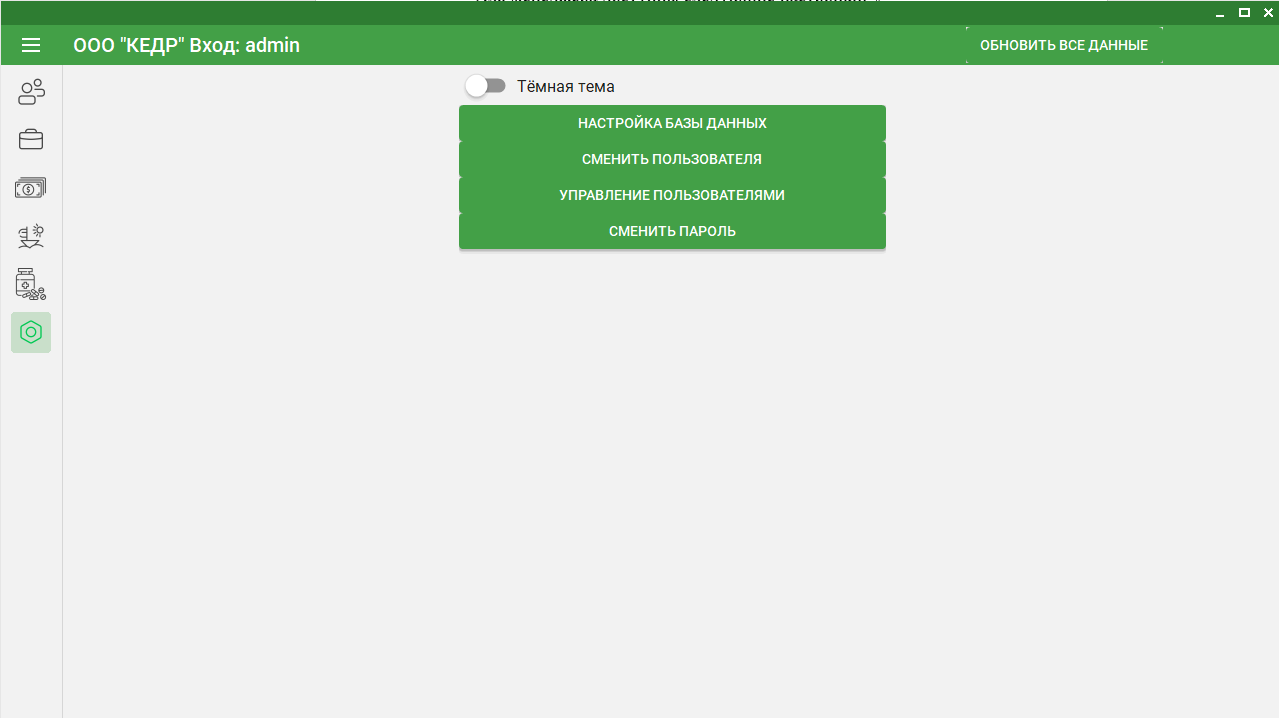


Рисунок 18 – ИС ООО Кедр. Вид окна с открытой вкладкой «Настройки»

**4.2.8 Использование вкладки «Карточка сотрудника»**

Если в приложении выполнена авторизация по номеру телефона, то в приложении становится доступной вкладка «Карточка сотрудника». На этой вкладке сотрудник может посмотреть информацию о себе и своих начислениях, отпусках, больничных. Для того, чтобы отобразить список следует выбрать соответствующий переключатель внизу вкладки. Вид окна с вкладкой представлен на рисунке 19.

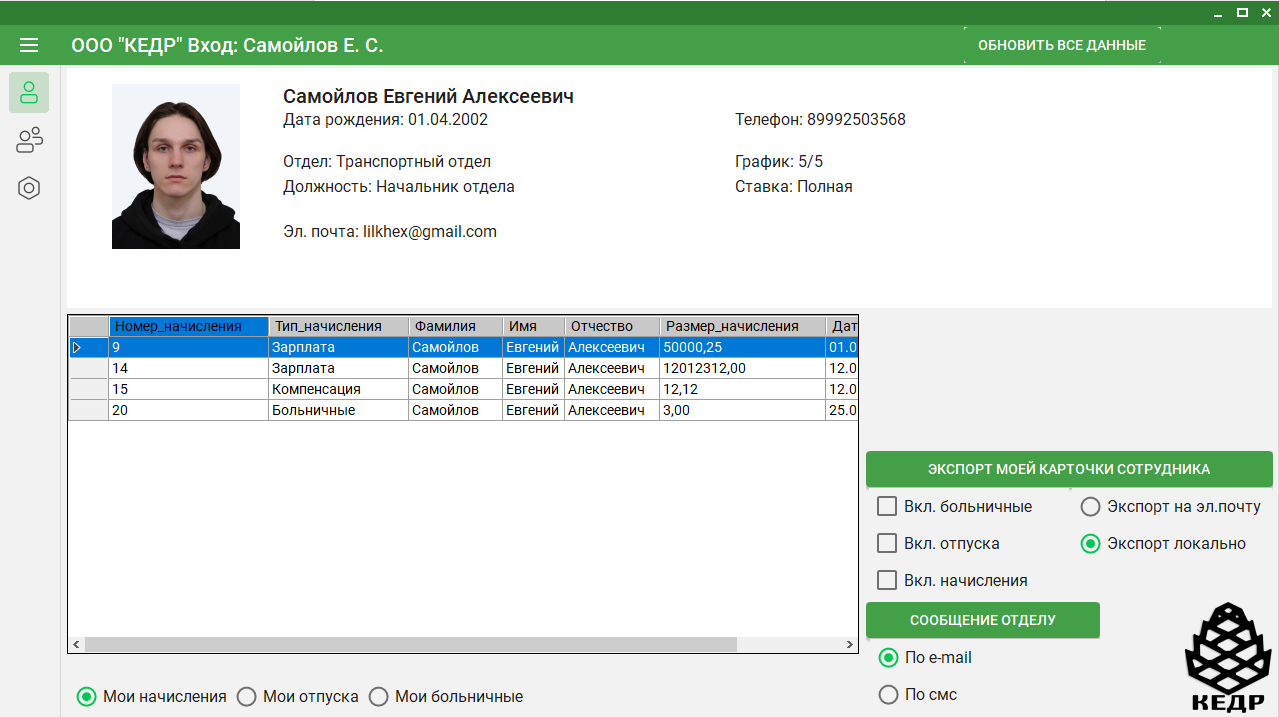


Рисунок 19 – ИС ООО Кедр. Вид окна с вкладкой «Карточка сотрудника»

Также сотруднику доступен экспорт своей карточки в формат .docx. При желании сотрудник может включить туда начисления, больничные или отпуска. Экспорт карточки доступен локально, в этом случае откроется диалоговое окно сохранения файла на диске. Также доступен экспорт на электронную почту, указанную в карточке сотрудника.

Если авторизованный сотрудник является начальником какого-либо отдела, то ему доступна функция рассылки сообщения сотрудникам своего отдела посредством отправки сообщений на электронную почту или смс-сообщений.

Для того, чтобы отправить сообщение, следует выбрать соответствующий пункт на панели в правом нижнем углу и нажать на кнопку «Отправить сообщение». После этого откроется окно, где следует ввести текст отправляемого сообщения и нажать на кнопку «Отправить».

Кнопка «Отправить» будет недоступна пока пользователь не введет текст сообщения. При отсутствии у сотрудников отдела номеров телефона либо адресов электронной почты приложение будет пропускать таких сотрудников и отправит сообщение только тем, у кого присутствует номер телефона или адрес электронной почты.

После успешной отправки сообщения приложение выведет соответствующее сообщение. Вид окна отправки сообщений представлен на рисунке 20.

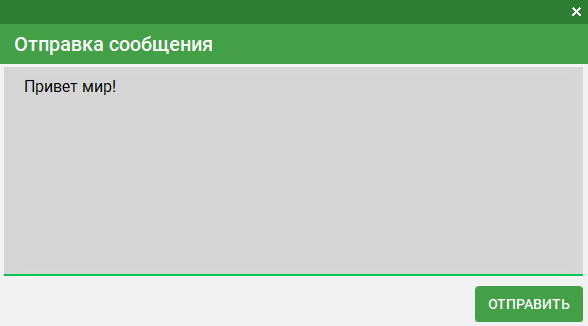


Рисунок 20 – ИС ООО Кедр. Вид окна «Отправка сообщения»

1. **Тестирование разработанной системы**
   1. Тестирование методом черного ящика

Разработанная ИС протестирована методом черного ящика с целью поиска ошибок и сбоев. В процессе тестирования проверены все формы приложения. Результаты тестирования представлены в таблице 3.

На этапе тестирования выявлена проблема с экспортом таблиц с списками начислений, больничных и отпусков. Проблема заключалась в том, что приложение не могло продолжать работу с файлом, так как он занят другим процессом.

После выявления проблемы во время тестирования приняты меры по ее устранению. В итоге использовано принудительное закрытие потока и открытие документа для работы уже в новом потоке.

Разработанная ИС соответствует всем требованиям, перечисленным в первом разделе, является полностью работоспособной и готова к использованию [17].

Таблица 3 – Набор тестов и результаты тестирования

|  |  |  |
| --- | --- | --- |
| Действие | Ожидаемый результат | Полученный результат |
| Выбор вкладки «Сотрудники» на главной форме | Открытие вкладки «Сотрудники» | Совпадает с ожидаемым результатом |
| Выбор вкладки «Начисления» на главной форме | Открытие вкладки «Начисления» | Совпадает с ожидаемым результатом |
| Нажатие на кнопку входа при незаполненных полях ввода логина и пароля | Вывод сообщения о том, что данные не заполнены | Совпадает с ожидаемым результатом |
| Нажатие на переключатель входа по номеру телефона | Вместо полей с логином и паролем отображаются поля с вводом телефона и кодом подтверждения | Совпадает с ожидаемым результатом |

Продолжение таблицы 3

|  |  |  |
| --- | --- | --- |
| Действие | Ожидаемый результат | Полученный результат |
| Нажатие на кнопку «Обновить все данные» | Обновление всех списков на главной форме | Совпадает с ожидаемым результатом |
| Ввод текста «С» в поле поиска сотрудника | Вывод списка сотрудников, фамилия которых содержит «С» | Совпадает с ожидаемым результатом |
| Нажатие на кнопку «Добавить сотрудника» | Открытие окна заполнения данных для нового сотрудника | Совпадает с ожидаемым результатом |
| Нажатие на кнопку «Редактировать» в строке сотрудника | Открытие окна редактирования данных сотрудника | Совпадает с ожидаемым результатом |
| Нажатие на кнопку «Уволить» при редактировании сотрудника | Пометка об увольнении в записи о сотруднике, вывод сообщения об успешной операции | Совпадает с ожидаемым результатом |
| Нажатие на переключатель «Темная тема» во вкладке настроек | Переключение цветовой темы приложения | Совпадает с ожидаемым результатом |
| Нажатие на кнопку «Экспорт карточки» с установленным флажком «Локально» во вкладке «Карточка сотрудника» | Открытие диалогового окна сохранения файла | Совпадает с ожидаемым результатом |
| Нажатие на кнопку «Экспорт карточки» с установленным флажком «На эл. почту» | Отправка документа с информацией о сотруднике на электронный почтовый ящик сотрудника | Совпадает с ожидаемым результатом |
| Установка флажка «Отображать уволенных сотрудников» | Отображение списка всех сотрудников, включая уволенных | Совпадает с ожидаемым результатом |
| Установка значения поля сортировка «По фамилии», порядок сортировки «По возрастанию» | Вывод списка сотрудников, отсортированных по фамилии в алфавитном порядке | Совпадает с ожидаемым результатом |
| Нажатие на кнопку «Сохранить» в окне добавления нового сотрудника | Вывод сообщения о результате добавления записи и обновление данных о сотрудниках на форме | Совпадает с ожидаемым результатом |
| Нажатие кнопки «Добавить запись о начислении» на вкладке «Начисления» с заполненными полями ввода | Вывод сообщения о результате добавления записи и обновление данных о начислениях на форме | Совпадает с ожидаемым результатом |
| Нажатие кнопки «Удалить» в контекстном меню системы кадров | Пометка об удалении записи в соответствии с выбором | Совпадает с ожидаемым результатом |

Продолжение таблицы 3

|  |  |  |
| --- | --- | --- |
| Действие | Ожидаемый результат | Полученный результат |
| Нажатие кнопки «Сохранить» в окне добавления нового сотрудника с незаполненным полем «Отчество» | Вывод сообщения об удачном сохранении записи, обновление списка | Совпадает с ожидаемым результатом |
| Нажатие кнопки «Сменить» на форме смены подключения к БД с заполненными полями ввода | Вывод сообщения с результатом смены настроек подключения | Совпадает с ожидаемым результатом |

На основе результатов тестирования можно сделать вывод что разработанное приложение функционирует корректно, явных ошибок в его работе не выявлено, следовательно, его можно рекомендовать для внедрения и использования.

**6 Определение затрат на разработку ИС**

Рассматриваемый дипломный проект нацелен на разработку информационной системы для ООО «Кедр».

Затраты на создание программного продукта , руб, определяются по формуле

, (1)

где – затраты на оплату машинного времени, руб;

– общие затраты, руб.

Трудоёмкость разработки программного продукта , чел.ч, определяется по формуле

(2)

где – затраты труда на подготовку описания задачи, чел.ч;

– затраты труда на исследование алгоритма решения задачи, чел.ч;

– затраты труда на разработку алгоритма, чел.ч;

– затраты труда на разработку диаграмм алгоритма, чел.ч;

– затраты труда на программирование по готовой диаграмме, чел.ч;

– затраты труда на отладку программы ЭВМ, чел.ч;

– затраты труда на подготовку документации, чел.ч.

Составление затрат вычисляется при помощи условного числа операторов. Условное число операторов , ед, в программе определяется по формуле

, (3)

где q – число операторов (исходных команд), ед;

с – коэффициент, учитывающий новизну и сложность программы;

р – коэффициент коррекции программы в ходе разработки, зависит от точности и корректности поставленной задачи (0.05-0.1).

Коэффициент, учитывающий новизну и сложность программы определяется исходя из таблицы 4 на пересечении групп сложности и степени новизны.

Программные продукты по степени новизны могут быть отнесены к одной из четырех групп:

* группа А – разработка принципиально новых задач,
* группа Б – разработка оригинальных программ,
* группа В – разработка программ с использованием типовых решений,
* группа Г – разовая типовая задача.

По степени сложности программные продукты могут быть отнесены к одной из трех групп:

* 1 – алгоритмы оптимизации и моделирования систем,
* 2 – задачи учета, отчетности и статистики,
* 3 – стандартные алгоритмы.

Таблица 4 – Значение коэффициента новизны и сложности программы

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Язык программирования | Группа сложности | Степень новизны | | | | Коэффициент В |
| А | Б | В | Г |
| Высокого уровня | 1 | 1,38 | 1,26 | 1,15 | 1,20 | 1,20 |
| 2 | 1,30 | 1,19 | 1,08 | 0,65 | 1,35 |
| 3 | 1,20 | 1,10 | 1,00 | 0,60 | 1,50 |
| Низкого уровня | 1 | 1,58 | 1,45 | 1,32 | 0,79 | 1,20 |
| 2 | 1,49 | 1,37 | 1,24 | 0,74 | 1,35 |
| 3 | 1,38 | 1,26 | 1,15 | 0,69 | 1,50 |

В разработанной программе число операторов составляет около 1060 (q=1060).

Созданный программный продукт по степени новизны относится к разработке программ с использованием типовых решений (группа В), а по степени сложности алгоритма к задаче учета, отчетности и статистики (группа 2). Тогда, по таблице 1 коэффициент c = 1,08 и коэффициент B = 1,35.

С учётом того, что задача поставлена достаточно чётко, коэффициент p примем равным 0,06 (p=0,06). Условное число операторов согласно формуле (3) составляет

Q=1060∙1,08∙(1,00+0,06)=1213,49 ед.

Затраты труда на подготовку описания задачи to, чел.ч, точно определить невозможно, т.к. это связано с творческим характером работы. Примем данное значение равным 15 чел.ч (to=15).

Затраты труда на изучение описания задачи с учётом уточнения описания и квалификации программиста tu , чел.ч, определяются по формуле

=(Q∙B)/(75..85∙K) , (4)

где Q – условное число команд;

B – коэффициент увеличения затрат труда вследствие недостаточного описания задачи, уточнений и некоторой недоработки;

K – коэффициент квалификации разработчика.

Коэффициент увеличения затрат труда вследствие недостаточного описания задачи, уточнений квалификации разработчика (К) можно по данным таблицы 5.

Таблица 5 – Коэффициент квалификации разработчика.

|  |  |
| --- | --- |
| Опыт работы | Коэффициент квалификации |
| До двух лет | 0,80 |
| 2-3 года | 1,00 |
| 3-5 лет | 1,10 – 1.20 |
| 5-7 лет | 1,30 – 1.40 |
| Более 7 лет | 1,50 – 1,60 |

По таблице 2 для работающих до двух лет К = 0,8.

=1213,49 ∙1,35/(85∙0,8)=24,09 чел.ч.

Теперь необходимо вычислить затраты труда на различных стадиях разработки программного продукта.

Затраты труда на разработку алгоритма решения задачи tа, чел.ч, определяются по формуле

, (5)

1213,49 /(75∙0,8)=20,22 чел.ч.

Затраты труда на разработку диаграмм алгоритма решения задачи tб , чел.ч, определяются по формуле

, (6)

1213,49 /(75∙0,8)=20,22 чел.ч.

Затраты труда на составление программы по готовой диаграмме tп, чел.ч, определяются по формуле

, (7)

=1213,49 /(75∙0,8)=20,22 чел.ч.

Затраты труда на отладку программы на ЭВМ при комплексной отладке tотл, чел.ч, определяются по формуле

, (8)

где – затраты труда на отладку программы на ЭВМ при автономной отладке одной задачи, чел.ч.

Затраты труда на отладку программы на ЭВМ при автономной отладке одной задачи tотл A, чел.ч, определяются по формуле

, (9)

=1213,49/(75∙0,8)= 20,22 чел.ч.

Произведен расчет затрат на отладку программы на ЭВМ при комплексной отладке по формуле (8)

1,5∙20,22=30,33 чел.ч.

Затраты труда на подготовку документации по задаче tд , чел.ч, определяются по формуле

, (10)

где – затраты труда на подготовку материалов рукописи, чел.ч;

– затраты на редактирование, печать и оформление документации, чел.ч.

Затраты труда на подготовку материалов рукописи tдр, чел.ч, определяются по формуле

=Q/(150…200∙K), (11)

=1213,49/(200∙0,8)=7,58 чел.ч.

Затраты на редактирование, печать и оформление документации tдо, чел.ч, определяются по формуле

=0,75∙tдр, (12)

.

Произведен расчет затрат труда на подготовку документации по задаче по формуле (10)

.

Трудоёмкость разработки программного продукта согласно формуле (2) составляет

.

Затраты на оплату машинного времени при отладке программы , руб, определяются по формуле

, (13)

где – цена машино–часа арендного времени, руб/ч;

– фактическое время отладки программы на ЭВМ, чел.ч.

Фактическое время отладки , чел.ч, определяется по формуле:

, (14)

.

Цена машино-часа , руб/ч, определяется по формуле

, (15)

где – затраты на эксплуатацию ЭВМ, руб;

– действительный месячный фонд времени ЭВМ, ч.

Действительный месячный фонд времени ЭВМ , ч, определяется по формуле

, (16)

где  – коэффициент, учитывающий северную надбавку для работающих в местах приравненных к Крайнему Северу;

– коэффициент, учитывающий районную надбавку на территориях, приравненных к Крайнему Северу.

Общее количество дней в месяце = 30, число праздничных и выходных дней = 10.

Время простоя в профилактических работах определяется как еженедельная профилактика по 4 часа.

Произведен расчет действительного месячного фонда времени ЭВМ по формуле (16)

.

Затраты на эксплуатацию ЭВМ , руб, определяются по формуле

, (17)

где – издержки на амортизацию, руб;

– издержки на электроэнергию, потребляемую ЭВМ, руб.

В статье 256, пункте 1 налогового кодекса Российской Федерации указано, что амортизируемым признается имущество со сроком полезного использования более 12 месяцев и первоначальной стоимостью более 100000 рублей. Так как рыночная стоимость компьютера, на котором велась работа по разработке программы приобретён по рублей, что не превышает 100000 рублей, то издержки на амортизацию можно считать

Стоимость электроэнергии, потребляемой за месяц, , руб., определяется по формуле

, (18)

где – суммарная мощность ЭВМ, кВт;

– стоимость 1 кВт∙ч электроэнергии, руб.

Согласно техническому паспорту ЭВМ, потребление электроэнергии составляет 0,065 кВт.

Произведен расчет стоимости электроэнергии, потребляемой за месяц по формуле (18)

Затраты на эксплуатацию ЭВМ согласно формуле (17) составляют

Цена машино-часа согласно формуле (15) составляет

Затраты на оплату машинного времени при отладке программы согласно формуле (13) составляют

Общие затраты , руб., определяются по формуле

, (19)

где – издержки на заработную плату, руб;

– издержки на отчисления во внебюджетные фонды;

– издержки на амортизацию, руб;

– издержки на электроэнергию, потребляемую ЭВМ, руб;

– издержки на прочие и накладные расходы, руб.

Заработная плата работников , руб, определяется по формуле

, (20)

где  – коэффициент, учитывающий северную надбавку для работающих в местах приравненных к Крайнему Северу;

– коэффициент, учитывающий районную надбавку на территориях, приравненных к Крайнему Северу.

Районный коэффициент составляет 20% от основной заработной платы, а выплаты за выслугу лет, проработанных на территории, приравненной к территории Крайнего Севера – 50% от основной заработной платы.

Оклад программистов, работающих в г. Архангельске составляет руб.

Произведен расчет заработной платы работников по формуле (20)

Отчисления во внебюджетные фонды , руб, определяются по формуле

, (21)

где – фонд оплаты труда, руб;

– размер отчислений во внебюджетные фонды, %.

Т.к. число работников соответствует одному, то равен (=), а размер отчислений составляют 30% от суммы заработной платы.

Произведен расчет отчислений во внебюджетные фонды по формуле (21)

Прочие затраты , руб, принимаются в размере 10% в общей сумме затрат и определяются по формуле

, (22)

Общие затраты согласно формуле (19) составляют

Затраты на создание программного продукта согласно формуле (1) составляют

В конечном итоге трудоёмкость создания программного продукта составила 142,64 человеко-часа. В результате произведённых расчётов затраты на создание программного продукта составляют рубль [12]. Стоимость аналогичных информационных на рынке составляет в среднем 100 тысяч рублей. Разница в цене разрабатываемого продукта объясняется тем, что зарплата программиста взята в минимальном размере оплаты труда.

**7 Охрана труда и техника безопасности при работе на ПК**

**7.1 Общие требования безопасности**

Освещенность на поверхности стола в зоне размещения рабочего документа должна быть 300-500 лк. Освещение не должно составлять бликов на поверхности экрана и превышать 300 лк. В помещении необходимо наличие как искусственных источников освещения, так и естественных.

Рабочее место для работы с ПК должно быть оборудовано следующим:

- клавиатурой, располагающейся на поверхности стола на расстоянии 100-300 мм от края, обращенного к пользователю,

- рабочим столом, имеющим ширину от 800 мм до 1400 мм, глубину от 800 мм до 1000 мм, имеющий пространство для ног с высотой не менее 600 мм, высотой не менее 500 мм, глубиной на уровне колен не менее 450 мм и на уровне вытянутых ног не менее 650 мм,

- рабочим стулом, регулируемым по высоте и углам наклона сиденья и спинки,

- расстояние от глаз до экрана должно быть 600-700 мм, а также угол наклона экрана монитора должен быть 10-15 градусов по отношению к вертикали.

На рабочем месте и в помещении необходимо поддерживать порядок и чистоту, а также проводить систематическое проветривание. В случае аварии нужно прекратить работу до устранения аварийных причин.

**7.2 Требования безопасности перед началом работы**

Перед началом работы с ПК необходимо выполнить следующее:

* подготовить рабочее место,
* отрегулировать освещение на рабочем месте,
* убедиться в отсутствии бликов на экране,
* проверить провода питания и отсутствие оголенных участков проводов,
* проверить правильность установки стола, стула, подставки для ног, угла наклона экрана, положения клавиатуры, положения «мыши», при необходимости произвести регулировку рабочего стола и кресла, а также расположение элементов компьютера в соответствии с требованиями эргономики и для исключения неудобных поз и длительных напряжений тела.

**7.3 Требования безопасности во время работы**

Во время работы с ПК запрещается:

* прикасаться к задней панели системного блока при наличии питания,
* переключать разъёмы интерфейсных кабелей периферийных устройств при включенном питании,
* допускать попадание влаги на поверхность системного блока, монитора, рабочую поверхность клавиатуры, принтеров и других устройств,
* производить самостоятельное вскрытие и ремонт оборудования,
* работать на компьютере при снятых кожухах,
* отключать оборудование от электросети и вынимать электровилку, держась за шнур.
  1. **Требования охраны труда в аварийных ситуациях**

В случаях обрыва проводов питания, неисправности заземления и других повреждений, появления гари, нужно немедленно отключить питание и сообщить об аварийной ситуации руководителю.

При возникновении пожара, задымлении:

* открыть запасные выходы из здания, обесточить электропитание, закрыть окна и прикрыть двери,
* немедленно сообщить по телефону «112» в пожарную охрану, оповестить работающих, поставить в известность руководителя подразделения, сообщить о возгорании на пост охраны,
* приступить к тушению пожара первичными средствами пожаротушения, если это не сопряжено с риском для жизни,
* организовать встречу пожарной команды,
* покинуть здание и находиться в зоне эвакуации.

При несчастном случае требуется:

* немедленно организовать первую помощь пострадавшему и при необходимости доставить его в медицинскую организацию,
* принять неотложные меры по предотвращению развития аварийной или иной чрезвычайной ситуации и воздействия травмирующих факторов на других лиц,
* сохранить до начала расследования несчастного случая обстановку, какой она была на момент происшествия, если это не угрожает жизни и здоровью других лиц и не ведёт к катастрофе, аварии или возникновению иных чрезвычайных обстоятельств, а в случае невозможности ее сохранения – зафиксировать сложившуюся обстановку (составить схемы, провести другие мероприятия).

Нельзя приступать к работе до устранения неисправностей.

**7.5 Требования охраны труда по окончанию работы**

По окончанию работы с ПК требуется отключить ПК от электросети, отключив тумблеры, а также вытащить вилку из розетки, протереть внешнюю поверхность ПК и прибрать рабочее место.

**ЗАКЛЮЧЕНИЕ**

В результате проделанной работы по написанию дипломного проекта достигнута поставленная цель в виде разработанной информационной системы для ООО «Кедр».

В ходе проделанной работы разработаны:

* база данных,
* клиентская часть ИС.

Также выполнены следующие поставленные задачи:

* проведен анализ предметной области,
* выбраны эффективные алгоритмы,
* спроектированы модели, необходимые для разработки,
* реализован экспорт данных в форматах .xls и .docx,
* реализована отправка сообщений по смс и электронной почте,
* проведена отладка и тестирование ИС,
* проанализированы полученные в ходе тестирования и отладки результаты работы ИС,
* составлено руководство пользователя по установке и эксплуатации ИС.

В результате достигнута основная цель дипломного проекта, т.е. разработана многопользовательская информационная система, обеспечивающая графическое и текстовое отображение информации и ее хранение.

В процессе разработки задействованы различные информационные ресурсы для расширения теоретических знаний о среде разработки, языке программирования С#, а также изучена справочная информация по работе с СУБД Microsoft SQL Server 2019.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Албахари, Д. C# 8.0. Справочник. Полное описание языка / Д. Албахари. – Москва : Вильямс, 2020 – 240 с.

2. Рочев, К. В. Информационные технологии. Анализ и проектирование информационных систем : учебное пособие для вузов / К. В. Рочев. — 3-е изд., стер. — Санкт-Петербург : Лань, 2022. — 128 с. — ISBN 978-5-507-44339-0. // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/223442 (дата обращения: 03.06.2022) – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.

3. Вагнер, Б. Наиболее эффективное программирование на C# / Б. Вагнер. – Москва : Альфа-книга, 2018. – 240 с.

4. Васильев, А.Н. Программирование для начинающих на C# / А.Н. Васильев – Москва : Эксмо, 2022. – 592 с.

5. Гагарина, Л.Г. Технология разработки программного обеспечения: учебное пособие / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Виснадул. – Москва : ИД «ФОРУМ» : ИНФРА-М, 2022. – 400 с.

6. Гвоздева, В.А. Информатика, автоматизированные информационные технологии и системы: Учебник / В.А. Гвоздева. – Москва : ИД «ФОРУМ» : ИНФРА-М, 2021. – 542 с.

7. Голицына, О.Л. Основы проектирования баз данных: учебное пособие / О.Л. Голицына, Т.Л. Партыка, И.И. Попов. – 2-е изд., перераб. – Москва: ФОРУМ : ИНФРА-М, 2020. – 416 с.

8. Бедердинова, О. И. Создание приложений баз данных в среде Visual Studio : учебное пособие / О.И. Бедердинова, Т.А. Минеева, Ю.А. Водовозова. — Москва : ИНФРА-М, 2021. — 94 с. - ISBN 978-5-16-109411-2. - URL: <https://znanium.com/catalog/product/1243816> (дата обращения 13.05.2022). – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.

9. Вейцман, В. М. Проектирование информационных систем : учебное пособие для спо / В. М. Вейцман. — 2-е изд., стер. — Санкт-Петербург : Лань, 2021. — 316 с. — ISBN 978-5-8114-8572-7. // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/177833 (дата обращения: 02.06.2022). – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.

10. Мартин, Р. Чистая архитектура. Искусство разработки программного обеспечения/ Р. Мартин – Санкт-Петербург : Питер, 2019. – 410 с.

11. Мартин, Р. Чистый код: создание, анализ и рефакторинг / Р. Мартин. – Санкт-Петербург : Питер, 2018. – 463 с.

12. Попова, Т.В. Технико-экономическое обоснование дипломных проектов. Учебно-методическое пособие по выполнению экономической части дипломных проектов для студентов / Т.В. Попова. – Архангельск : АКТ (ф) СПбГУТ, 2020. – 43 с.

13. [Скит, Д.](https://www.combook.ru/authors/%D0%A1%D0%BA%D0%B8%D1%82%20%D0%94%D0%B6./) C# для профессионалов: тонкости программирования / Д. Скит. – Москва : Вильямс, 2019. – 608 с.

14. Стиллмен, Э. Изучаем C# / Э. Стиллмен, Дж. Грин. – 4-е изд. – Санкт-Петербург : Питер, 2022. – 816 с.

15. Тарасов, С.В. СУБД для программиста. Базы данных изнутри / С.В. Тарасов. – Москва : СОЛОНПРЕСС, 2020. – 320 с.

16. Туманов, В.Е. Основы проектирования реляционных баз данных / В.Е. Туманов. – Москва : Национальный Открытый Университет «ИНТУИТ», 2017. – 514 с.

17. Хант, Э. Программист-прогматик / Э. Хант, Д. Томас. – 2-е изд. –Киев : Диалектика, 2020. – 212 с.

18. Хорев, П.Б. Объектно-ориентированное программирование с примерами на C#: Учебное пособие / П.Б. Хорев. – Москва : Форум : ИНФРА-М, 2017. – 200 с.

19. Храпченко М.В. Базы данных. Практическое применение СУБД SQL- и NoSOL-типа для применения проектирования информационных систем / М.В. Храпченко. – Москва : Форум 2020. – 368 с.

20. Шарп, Д. Microsoft Visual C# / Д. Шарп. – 8-е изд – Санкт-Петербург : Питер, 2017. – 818 с.