



ФЕДЕРАЛЬНОЕ АГЕНТСТВО МОРСКОГО И РЕЧНОГО ТРАНСПОРТА

**Федеральное государственное бюджетное образовательное учреждение высшего образования
«Государственный университет морского и речного флота имени адмирала С.О. Макарова»
(ФГБОУ ВО «ГУМРФ имени адмирала С.О. Макарова»)
Воронежский филиал**

Кафедра математики, информационных систем и технологий

Направление подготовки 09.03.02 Информационные системы и технологии
(код, наименование направления подготовки/специальности)

Форма обучения очная

«К ЗАЩИТЕ ДОПУЩЕН(А)»
Заведующий кафедрой

(подпись)

Черняева С. Н.

(ФИО)

20

Выпускная квалификационная работа

Обучающегося Давлетшина Булата Азатовича
(фамилия, имя, отчество)

Вид работы выпускная квалификационная работа бакалавра
(выпускная квалификационная работа бакалавра, специалиста, магистра)

Пояснительная записка

Тема Разработка приложения для учета рабочего времени сотрудников предприятия
(на примере ООО «АТП №1»)
(полное название темы квалификационной работы, в соответствии с приказом об утверждении тематики ВКР)

Руководитель работы доцент, к.э.н. Скрипников О. А.
(должность, подпись, фамилия, инициалы, дата)

Консультант _____
(при наличии) (должность, подпись, фамилия, инициалы, дата)

Консультант _____
(должность, подпись, фамилия, инициалы, дата)

Обучающийся Давлетшин Булат Азатович
(подпись, фамилия, инициалы, дата)

Воронеж
2024

ФЕДЕРАЛЬНОЕ АГЕНТСТВО МОРСКОГО И РЕЧНОГО ТРАНСПОРТА

**Федеральное государственное бюджетное образовательное учреждение высшего образования
«Государственный университет морского и речного флота имени адмирала С.О. Макарова»
(ФГБОУ ВО «ГУМРФ имени адмирала С.О. Макарова»)
Воронежский филиал**

Кафедра математики, информационных систем и технологий

Направление подготовки 09.03.02 Информационные системы и технологии

(код, наименование направления подготовки/специальности)

Форма обучения очная

УТВЕРЖДАЮ
Заведующий кафедрой

(подпись)

Черняева С. Н.

(ФИО)

2024

**Задание
на выпускную квалификационную работу**

Вид работы ВКР бакалавра

(ВКР бакалавра, ВКР специалиста, ВКР магистра)

Обучающемуся Давлетшину Булату Азатовичу

(фамилия, имя, отчество)

Тема Разработка приложения для учета рабочего времени сотрудников предприятия
(на примере ООО «АТП №1»)

Утверждена приказом ректора университета от _____ 20____, № _____

Срок сдачи законченной работы _____ 20____

Исходные данные (или цель ВКР):

Разработать приложения учета рабочего времени сотрудников предприятия.

Перечень подлежащих исследованию, разработке, проектированию вопросов (краткое содержание ВКР):

(актуальность темы, цели и задачи ВКР; аналитический обзор литературных источников; постановка задачи исследования, разработки, проектирования; содержание процедуры исследования, разработки, проектирования; обсуждение результатов; дополнительные вопросы, подлежащие разработке; заключение – выводы по работе в целом, оценка степени решения поставленных задач, практические рекомендации; и др.)

– Введение. Актуальность выбранной темы, цель и задачи ВКР

(наименование вопроса, раздела и его краткое содержание)

– Исследовательский раздел. Правовые аспекты рабочего времени, Существующие методы учета рабочего времени, Анализ и оценка, Анализ существующих приложений для учета рабочего времени, Оценка эффективности использования ресурсов, Анализ имеющегося программного обеспечения на предприятии, Организационная структура предприятия ООО «АТП №1», Анализ языков программирования, Языки программирования потерявшие популярность, Исследование языка программирования COBOL, Исследование языка программирования Fortran, Исследование языка программирования Visual Basic, Языки программирования пользующиеся спросом, Язык программирования С# (Си Шарп), Язык программирования Python, Язык программирования Java.

(наименование вопроса, раздела и его краткое содержание)

– Проектный раздел. Определение среды разработки приложения, Проектирование и разработка приложения.

(наименование вопроса, раздела и его краткое содержание)

– Заключение. Выводы по работе в целом. Оценка степени решения поставленных задач

(наименование вопроса, раздела и его краткое содержание)

Практические рекомендации

Перечень графического материала (или презентационного материала):

1. Титульный лист

2. Цель и задачи ВКР

3. Режим рабочего времени

4. Аналоги программ по учету рабочего времени

5. Используемая программа на предприятие

6. Организационная структура предприятия ООО «АТП №1»

7. Окно авторизации для разработанного приложения

8. Главное меню приложения

9. Результаты ВКР

Консультанты по разделам ВКР (при наличии):

1. _____

(наименование раздела, ученая степень, ученое звание и должность, ФИО консультанта)

2. _____

(наименование раздела, ученая степень, ученое звание и должность, ФИО консультанта)

3. _____

(наименование раздела, ученая степень, ученое звание и должность, ФИО консультанта)

Дата выдачи задания: _____ 20____

Задание согласовано и принято к исполнению: _____ 20____

Руководитель ВКР: доцент, к. э. н.

Скрипников О.А.

(должность, ученая степень, ученое звание, ФИО)

(подпись)

Обучающийся: ИТ-4-1 Давлетшин Булат Азатович
(учебная группа, ФИО)

(подпись)

ФЕДЕРАЛЬНОЕ АГЕНТСТВО МОРСКОГО И РЕЧНОГО ТРАНСПОРТА

**Федеральное государственное бюджетное образовательное учреждение высшего образования
«Государственный университет морского и речного флота имени адмирала С.О. Макарова»
(ФГБОУ ВО «ГУМРФ имени адмирала С.О. Макарова»)
Воронежский филиал**

**ОТЗЫВ
руководителя выпускной квалификационной работы**

Обучающегося Давлетшина Булата Азатовича
(фамилия, имя, отчество)

На тему Разработка приложения для учета рабочего времени сотрудников предприятия

Направление подготовки 09.03.02 Информационные системы и технологии
(код и наименование направления подготовки/специальности)

Кафедра математики, информационных систем и технологий

Руководитель выпускной квалификационной работы _____

(должность, ученая степень, ученое звание, фамилия, имя, отчество)

(в отзыве должна содержаться характеристика проделанной обучающимся работы по всем разделам выпускной квалификационной работы: уровень, полнота и качество поэтапной разработки обучающимся темы ВКР; отношение обучающегося к работе при ее написании, его аккуратность, добросовестность, трудоспособность; степень самостоятельности и инициативности обучающегося при написании работы; работа с литературой; умение обобщать и делать правильные выводы и предложения из полученных данных; достоинства и недостатки работы; оценка автора работы как будущего специалиста и возможностей заниматься тем или иным видом трудовой деятельности)

Заключение _____
(оценивается уровень достижения обучающимся запланированных результатов выполнения ВКР)

Руководитель ВКР: _____ (подпись) _____ (ФИО)

Дата _____ 20 ____

Содержание

ВВЕДЕНИЕ	7
1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИЗУЧЕНИЯ РАБОЧЕГО ВРЕМЕНИ	9
1.1 Правовые аспекты рабочего времени	9
1.2 Существующие методы учета рабочего времени	12
1.3 Анализ и оценка	13
1.3.1 Анализ существующих приложений для учета рабочего времени ..	13
1.3.2 Оценка эффективности использования ресурсов	14
1.3.3 Анализ имеющегося программного обеспечения на предприятии ..	15
1.3.4 Организационная структура предприятия ООО «АТП №1»	20
1.4 Анализ языков программирования	22
1.4.1 Языки программирования потерявшие популярность	22
1.4.2 Исследование языка программирования COBOL	22
1.4.3 Исследование языка программирования Fortran	23
1.4.4 Исследование языка программирования Visual Basic	24
1.4.5 Языки программирования пользующиеся спросом	25
1.4.6 Язык программирования С# (Си Шарп)	26
1.4.7 Язык программирования Python	26
1.4.8 Язык программирования Java	27
2 ПРОЕКТНЫЙ РАЗДЕЛ. ВЫБОР СРЕДЫ РАЗРАБОТКИ	28
2.1 Определение среды разработки приложения	28
2.2 Проектирование и разработка приложения	28
ЗАКЛЮЧЕНИЕ	47
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ	48
Приложение А	53

ВВЕДЕНИЕ

Тема разработки приложения учета рабочего времени сотрудников является актуальной и востребованной в современном бизнесе. С учетом изменяющихся требований к работе и управлению персоналом, такое приложение может значительно облегчить процесс учета рабочего времени, повысить эффективность работы сотрудников и улучшить контроль за выполнением задач.

В условиях современного рынка, где мобильность и гибкость работы становятся все более важными, приложение учета рабочего времени может помочь отслеживать время работы сотрудников как в офисе, так и на удаленной работе. Это позволит компаниям более точно определить рабочие часы сотрудников, учитывая их отгулы, переработки и другие нюансы.

Также приложение учета рабочего времени может помочь улучшить коммуникацию между сотрудниками и руководством компании, обеспечивая более прозрачные отчеты и подробную статистику по рабочему времени.

Научная новизна данной работы заключается в разработке и применении специализированного приложения для учета рабочего времени сотрудников с использованием современных технологий и методов.

Объектом исследования в данной работе выступает ООО «АТП №1».

Предмет исследования – приложение для учета рабочего времени сотрудников предприятия.

Исходя из обоснованной актуальности темы, целью данной работы является: разработка приложения для учета рабочего времени сотрудников предприятия.

Для достижения поставленной цели необходимо выполнить следующие задачи:

- изучить теоретико-методологические основы рабочего времени в организации;
- провести анализ существующих систем и приложений для учета рабочего времени;

- провести анализ программного обеспечения используемым предприятием;
- исследовать существующие языки программирования;
- оценить эффективность разработанного проекта.

1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИЗУЧЕНИЯ РАБОЧЕГО ВРЕМЕНИ

1.1 Правовые аспекты рабочего времени

Прибыль и успех компании напрямую зависят от того, насколько эффективно она работает. Одной из предпосылок эффективной работы является автоматизация задач, при которой машины выполняют работу, с которой они в состоянии справиться правильно, в то время как люди настраивают машины и наблюдают за ними. Естественно, не в каждой задаче людей можно заменить машинами, но в тех случаях, когда это применимо, эффективность работы значительно возрастает, а количество ошибок и затрат снижается. В настоящее время большинство компаний внедрили тот или иной метод отслеживания время работы своих сотрудников. Чеки на зарплату выдаются в соответствии с полученными данными, что гарантирует выплату сотрудникам в соответствии с выполненной работой.

Некоторые работодатели по-прежнему используют ручку и бумагу, чтобы отмечать, когда сотрудники приходят на работу и когда уходят. В результате к концу месяца у менеджеров на столе оказываются кипы бумаги. К последствиям такого подхода относятся значительное количество часов, затрачиваемых на ручные вычисления, а также человеческие ошибки.

Что такое режим рабочего времени и зачем он нужен:

Особенности режима рабочего времени заключаются в том, что каждая организация, предприятие или фирма вправе самостоятельно устанавливать его исходя из специфики своей деятельности. Стоит помнить, что режим работы, выходные дни, количество смен и другие пункты должны быть прописаны в трудовом договоре. Если работнику предлагается изменение режима рабочего времени, этот нюанс следует не только оговорить, но и заключить в трудовой договор. Есть пять режимов работы:

– Гибкий график работы. Характеризуется тем, что продолжительность, начало или окончание работы работник определяет

самостоятельно, но по согласованию с работодателем и с внесением в трудовой договор сведений о согласии руководства на гибкий график.

– Неполный рабочий день. Устанавливается также по соглашению между руководством и работником. Оплата за данный вид работ будет производиться в зависимости от затраченного на работу времени или объема выполненной работы. За введением неполного рабочего времени обычно могут претендовать лишь несколько категорий граждан, указанных в ст. 93 Трудового кодекса Российской Федерации.

– Ненормированный рабочий день. Он заключается в том, что отдельные работники или весь трудовой коллектив согласно договору, выполняют свои обязанности во внеурочное время или в течение срока, меньшего установленного в организации рабочего дня. Подобные нюансы оговариваются отдельно между работниками и работодателем или прописываются в трудовом договоре, если специфика работы предполагает ненормирование всех рабочих дней.

– График работы на сменах. Это обычная практика в компаниях и организациях, где процесс производства требует больше времени, чем стандартный рабочий день. Сюда относятся фабрики и другие производственные предприятия. Каждая смена работает определенное количество времени, необходимое для эффективного производства и рационального использования оборудования. В зависимости от размера и специфики производства, в день может быть от двух до четырех смен.

График работы должен быть оговорен в трудовом договоре. В противном случае, в случае переработки на несколько часов будет трудно доказать свои права и получить компенсацию за свои трудовые усилия.

Таблица 1 – рабочее время и время отдыха в соответствии ТК РФ

Статья	91.	Определяет предел нормального рабочего времени в 40 часов в неделю.
Понятие рабочего времени.		В этих пределах нормальная продолжительность рабочего времени устанавливается трудовым договором и правилами внутреннего трудового распорядка.

Продолжение таблицы 1

Статья 93. Неполное рабочее время.	По соглашению сторон трудового договора работнику как при приеме на работу, так и впоследствии может устанавливаться неполное рабочее время (неполный рабочий день (смена) и (или) неполная рабочая неделя, в том числе с разделением рабочего дня на части).
Статья 100. Режим рабочего времени.	Режим рабочего времени должен предусматривать продолжительность рабочей недели (пятидневная с двумя выходными днями, шестидневная с одним выходным днем, рабочая неделя с предоставлением выходных дней по скользящему графику, неполная рабочая неделя), работу с ненормированным рабочим днем для отдельных категорий работников, продолжительность ежедневной работы (смены), в том числе неполного рабочего дня (смены), время начала и окончания работы, время перерывов в работе, число смен в сутки, чередование рабочих и нерабочих дней.
Статья 106. Понятие времени отдыха.	Время отдыха - время, в течение которого работник свободен от исполнения трудовых обязанностей и которое он может использовать по своему усмотрению.
Статья 107. Виды времени отдыха.	Видами времени отдыха являются: перерывы в течение рабочего дня (смены); ежедневный (междусменный) отдых; выходные дни (еженедельный непрерывный отдых); нерабочие праздничные дни; отпуска.
Статья 109. Специальные перерывы для обогрева и отдыха.	Работникам, работающим в холодное время года на открытом воздухе или в закрытых необогреваемых помещениях, а также грузчикам, занятым на погрузочно-разгрузочных работах, и другим работникам в необходимых случаях предоставляются специальные перерывы для обогрева и отдыха, которые включаются в рабочее время. Работодатель обязан обеспечить оборудование помещений для обогрева и отдыха работников.
Статья 111. Выходные дни.	Всем работникам предоставляются выходные дни (еженедельный непрерывный отдых). При пятидневной рабочей неделе работникам предоставляются два выходных дня в неделю, при шестидневной рабочей неделе - один выходной день.

Графики для работников формируют с учётом их должностей и обязанностей. Отметим, что не все успевают выполнить работу за 8 часов на пятидневке. А некоторым сотрудникам нужно задерживаться на смене дольше, в силу специфики предприятия.

1.2 Существующие методы учета рабочего времени

Учет времени может осуществляться различными способами, от ручного заполнения журналов рабочего времени до использования современных онлайн-приложений.

Ручной учет времени включает в себя заполнение журналов рабочего времени с указанием времени прихода и ухода с работы, а также перерывов. Для более детального контроля над рабочим временем сотрудников используют журналы учета, в которых сотрудники подписываются при каждом приходе и уходе с работы. Также для точного учета времени внедряются карты доступа, системы сканирования отпечатков пальцев или распознавания лиц.

Существуют специальные программы для учета рабочего времени с функцией автоматизации подсчета отработанных часов, перерывов и составления отчетов.

Кроме того, существуют системы учета времени по проектам или задачам, позволяющие работникам указывать время, затраченное на каждую конкретную задачу. С использованием современных мобильных устройств и онлайн-приложений сотрудники могут записывать свое рабочее время в любое время и из любой точки с доступом к интернету.

Для контроля качества работы на некоторых предприятиях устанавливают видеочамеры. Однако это часто вызывает у сотрудников нервозность и напряжение - не всем удается комфортно работать, чувствуя, что их постоянно наблюдают. Этот метод следует использовать только в

крайних случаях, если риск оправдан. Еще одним минусом видеонаблюдения является необходимость наличия специального сотрудника, который будет работать с камерами, наблюдать за обстановкой и фиксировать нарушения трудового порядка.

Другим способом четкого контроля за рабочим временем является использование тайм-трекеров. Принцип работы с ними прост - программы устанавливаются на рабочие компьютеры сотрудников, время работы отображается в специальном отчете. На основе этих данных ответственный за учет указывает часы в таблице.

1.3 Анализ и оценка

1.3.1 Анализ существующих приложений для учета рабочего времени

Существует множество программ для учета рабочего времени, проведем анализ и вкратце рассмотрим положительные и отрицательные стороны.

– Toggl — это простая и удобная программа для отслеживания времени, которая позволяет сотрудникам отслеживать время, затраченное на различные задачи и проекты.

К положительным качествам можно отнести такие пункты как:

- Удобный интерфейс;
- Удобная аналитика и отчетность.

К отрицательным же сторонам данной программы мы можем добавить:

- Ограничение функционала в бесплатной версии;
- Не совсем точное отслеживание рабочего времени.

– TimeCamp — это программа учета рабочего времени, которая предоставляет детальную информацию о том, как сотрудники тратят свое время, а также позволяет управлять проектами и задачами.

Положительными качествами данной программы являются:

- Обширный функционал;

- Поддержка мобильных устройств.

Отрицательными качествами будут:

- Отсутствие возможности интеграции с другими приложениями;
- Ограниченная поддержка обратной связи.

– Clockify — это бесплатная программа учета рабочего времени, которая позволяет сотрудникам отслеживать время, затраченное на различные задачи, проекты и клиентов.

Отметим такие положительные качества как:

- Возможность интеграции с другими инструментами для управления проектами.

Отрицательные качества:

- Не всегда стабильная работа и задержки при обновлении данных;
- Не все функции работают без сбоев;
- Ограниченные настройки и персонализация интерфейса.

Все эти программы предоставляют схожий функционал, но отличаются по уровню сложности, стоимости и возможностям интеграции с другими системами учета и управления проектами.

При выборе программы учета рабочего времени важно учитывать потребности и особенности вашей компании, а также предпочтения сотрудников.

1.3.2 Оценка эффективности использования ресурсов

В борьбе со временем, где каждая секунда имеет значение, организации, предприятия и отдельные лица постоянно ищут способы оптимальной производительности и эффективности труда. Хотя традиционные методы бухгалтерского учета, такие как финансовый учет и учет затрат, предоставляют ценные идеи, существует краеугольный камень, которого часто недооценивают: учет времени.

Учет времени, так же как учет затрат и финансовый учет, играет важную роль в финансовом благополучии и успехе организаций. Как денежный поток важен для бизнеса, так и управление временем не менее важно для формирования пути развития организации. По аналогии с потоком денег, представляющим движение средств внутри компании, поток времени означает распределение и использование рабочего времени в организации.

Эффективное управление временем напрямую влияет на производительность труда сотрудников, завершение проектов и, в итоге, на денежный поток организации. Как компании внимательно отслеживают и управляют денежными потоками для обеспечения финансовой стабильности, так и учет времени предоставляет инструменты для мониторинга, анализа и оптимизации временных ресурсов.

Понимая, как рабочее время распределяется и используется в различных сферах деятельности, работодатели могут выявлять узкие места, оптимизировать процессы и принимать обоснованные решения для повышения общей производительности труда и финансовых показателей.

Учет рабочего времени включает в себя отслеживание и анализ того, как мы распоряжаемся своим временем. Это выходит за рамки финансового учета и учета расходов, учитывая время как ценный ресурс, который обязательно нужно учитывать. Путем тщательного измерения и анализа времени организации могут оценить свою продуктивность, выявить неэффективность и принять обоснованные решения для оптимизации производительности.

1.3.3 Анализ имеющегося программного обеспечения на предприятии

Как пример разберем процесс учета рабочего времени сотрудников в ООО «АТП №1». Как правило, ежедневно его ведут бухгалтера, кроме выходных и праздничных дней. В штате сотрудников ООО имеется помощник бухгалтера, что обеспечивает постоянное внесение отработанных

часов в табель учета рабочего времени. Таким образом, если один из сотрудников бухгалтерии решит взять отпуск или больничный, второй служит гарантией, что труд остальных работников будет справедливо оценен.

На рассматриваемом предприятии в качестве основной используется: 1С: Бухгалтерия предприятия – это программа бухгалтерского учета, которая позволяет вести учет в соответствии с требованиями законодательства и спецификой деятельности этого предприятия. Программа обладает широким функционалом и может быть настроена под конкретные потребности предприятия.

Оценка эффективности программы на предприятии проводится с целью определения, насколько успешно программа справляется с поставленными задачами и сколько пользы она приносит предприятию.

Считаю, что непременно стоит отметить такие важные пункты как:

- Сокращение времени на ведение бухгалтерского учета. Программа может автоматизировать многие процессы, что позволит уменьшить количество ручной бумажной работы и сократить время на выполнение бухгалтерских операций.

- Улучшение качества учета. Программа позволяет избежать ошибок и искажений в учете благодаря автоматизации процессов и проверке данных.

- Увеличение производительности сотрудников. Поскольку программа упрощает и ускоряет выполнение бухгалтерских задач, сотрудники могут сосредоточиться на более важных деловых задачах.

Теперь рассмотрим интерфейс 1С:Бухгалтерия предприятия используемое в организации.

Так как данная программа уже является по сути готовой версией, работникам бухгалтерии только и остается, что внести данные организации, такие как: ИНН, КПП, ОГРН. Также вносятся и остальные сопутствующие и необходимые данные, для нормальной работы с «1С-Отчетность» и возможности подключить электронный документооборот(ЭДО). Далее, в

разделах мы переходим в «Зарплаты и кадры», затем в модулях находим «Кадры» и из списка выбираем «Сотрудники».(см. рис. 1)

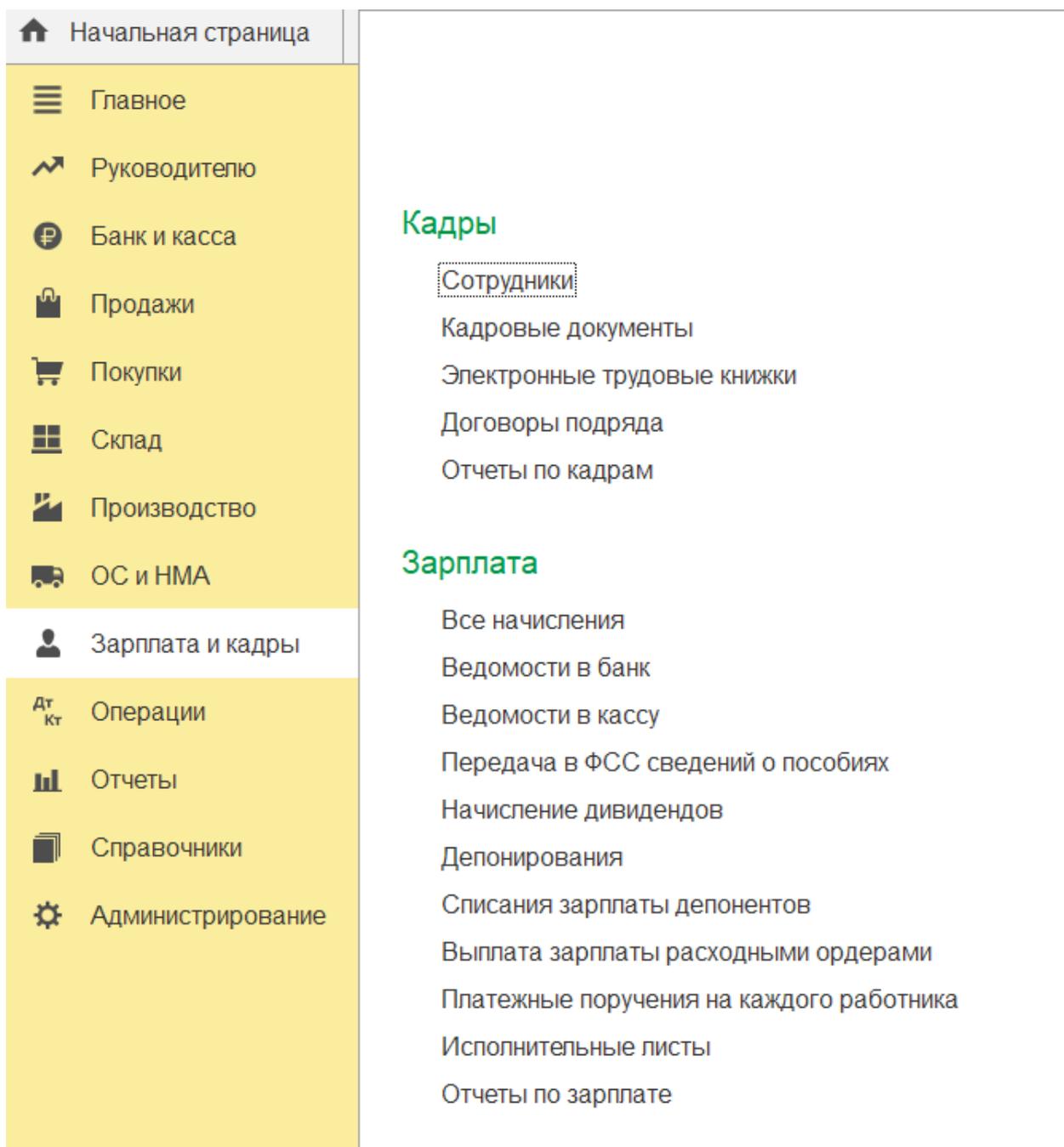


Рисунок 1 – Навигация по 1С:Бухгалтерия предприятия

Далее заполняются данные каждого сотрудника: серия и номер паспорта, орган и дату выдачи, код подразделения, дату рождения, ИНН, СНИЛС. Указывается должность, трудовая функция, оклад. В конечном результате, все выводится в список который можно просматривать по подразделениям.(см. рис. 2, 3)

Сотрудники

Подразделение: Бухгалтерия

ФИО	Таб. номер	Организация	Подразделение	Должность
Пучкина Екатерина Григорьевна	0000-00001	Автопредприятие №1 ОАО	Бухгалтерия	Бухгалтер
Созданов Александр Геннадьевич	0000-00002	Автопредприятие №1 ОАО	Информационный отдел	Системный администратор/оператор АИС
Соловьев Андрей Сергеевич	0000-00003	Автопредприятие №1 ОАО	Административно-хозяйственный отдел (А.Х.О.)	Слесарь
Горькова Галина Александровна	0000-00004	Автопредприятие №1 ОАО	Бухгалтерия	Бухгалтер
Лыбидов Алексей Сергеевич	0000-00005	Автопредприятие №1 ОАО	Административно-хозяйственный отдел (А.Х.О.)	Начальник технического подразделения
Рыбачев Дмитрий Владимирович	0000-00006	Автопредприятие №1 ОАО	Административно-хозяйственный отдел (А.Х.О.)	Техник-электрик
Арнутаев Антон Дмитриевич	0000-00007	Автопредприятие №1 ОАО	Кладовский отдел	Кристал
Козлов Иван Вячеславович	0000-00008	Автопредприятие №1 ОАО	Административно-хозяйственный отдел (А.Х.О.)	Специалист по сварочным работам

Рисунок 2 – Полный список сотрудников

Сотрудники

Подразделение: Административно-хозяйственный отдел (А.Х.О.)

ФИО	Таб. номер	Организация	Подразделение	Должность
Соловьев Андрей Сергеевич	0000-00003	Автопредприятие №1 ОАО	Административно-хозяйственный отдел (А.Х.О.)	Слесарь
Лыбидов Алексей Сергеевич	0000-00005	Автопредприятие №1 ОАО	Административно-хозяйственный отдел (А.Х.О.)	Начальник технического подразделения
Рыбачев Дмитрий Владимирович	0000-00006	Автопредприятие №1 ОАО	Административно-хозяйственный отдел (А.Х.О.)	Техник-электрик
Козлов Иван Вячеславович	0000-00008	Автопредприятие №1 ОАО	Административно-хозяйственный отдел (А.Х.О.)	Специалист по сварочным работам

Рисунок 3 – Список сотрудников с выбором подразделения

Теперь остается лишь заполнить график работы в готовой конфигурации. Так как до этого бухгалтеры уже внесли большую часть данных, остается всего лишь добавить людей в таблицу учета рабочего времени.(см. рис. 4)

Таким образом, воочию видим, что рабочая программа 1С имеет ряд преимуществ, но также и ряд нескольких недочетов.

1.3.4 Организационная структура предприятия ООО «АТП №1»

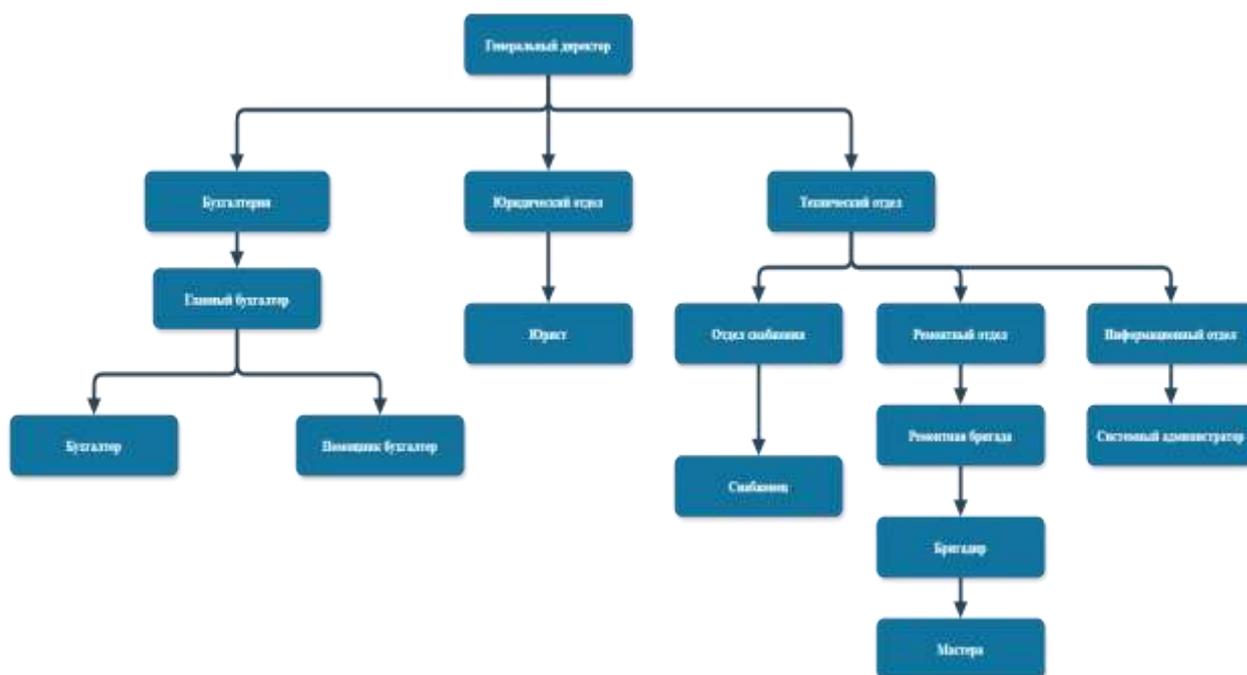


Рисунок 6 - Организационная структура предприятия ООО «АТП №1»

Задачи сотрудников предприятия:

Генеральный директор:

- руководит всей деятельностью организации;
- обеспечивает законность деятельности компании;
- организует учет и делопроизводство;
- отвечает за увеличение показателей эффективности предприятия:

прибыли, рентабельности, оборота.

Главный бухгалтер и его подчиненные:

- организацию работы по постановке и ведению бухгалтерского учёта организации;
- формирование учётной политики в соответствии с законодательством о бухгалтерском учёте;

- ведение регистров бухгалтерского учёта на основе применения современных информационных технологий;

- обеспечение сохранности бухгалтерских документов и сдачи их в установленном порядке в архив.

Юрист:

- выполняет свои должностные обязанности в рамках существующего законодательства;

- не допускает правовых нарушений, которые могут влечь за собой убытки;

- взаимодействовать с разными сферами общества и государственными органами в интересах предприятия;

- применять правовые нормы, полезные для интересов предприятия.

Снабженец:

- обеспечение поставок необходимых товаров;

- определение метода закупок товаров;

- мониторинг рыночной ситуации.

Бригадира и мастеров:

- организация работ;

- распределение задач и контроль за их выполнением;

- подготовка и проведение учений для повышения эффективности бригады;

- мастера осуществляет разборку, ремонт, сборку и испытания сложных узлов и механизмов. Выполняет ремонт, монтаж нового и демонтаж старого оборудования. Производит испытание, регулирование, наладку сложного оборудования, агрегатов и машин, сдачу после ремонта. Занимается изготовлением сложных приспособлений для ремонта и монтажа. Составляет дефектные ведомости на ремонт.

Системный администратор:

- установка, настройка технического оборудования, ПО и других компонентов инфраструктуры;
- резервное копирование и восстановление данных, защита от потери информации.

1.4 Анализ языков программирования

1.4.1 Языки программирования потерявшие популярность

На данный момент существует приличное количество языков программирования, при этом множество из них уже давно не актуально. Время летит вперед и технологии требуют постоянного развития.

Предлагаю немного разобрать языки программирования, которые стали менее актуальными.

1.4.2 Исследование языка программирования COBOL

Начнем, пожалуй, с такого языка как COBOL.

COBOL — это аббревиатура от Common Business Oriented Language, язык программирования, который был создан для использования в бизнесе. Изначально COBOL был процедурным, но с 2002 года COBOL стал объектно-ориентированным. Был разработан Конференцией языков информационных систем (CODASYL).

В конце 1950-х годов производители компьютеров были обеспокоены огромными затратами на программирование. В 1959 году исследование показало, что стоимость программирования любой установки для обработки данных составляет около 800 тысяч долларов США, а когда дело доходит до перевода программы для запуска на новой машине, это обойдется примерно в 6 миллионов долларов.

Когда количество новых языков программирования росло огромными темпами, исследование также показало, что если бы использовался обычный

бизнес-ориентированный язык, то перевод был бы быстрее и дешевле. Итак, 8 апреля 1959 года Мэри К. Хоуз, специалист по компьютерным наукам в корпорации «Burroughs», созвала совещание пользователей компьютеров и производителей в Пенсильванском университете(США), чтобы организовать официальную встречу по общим деловым языкам.

Ну а что там насчет достоинств? Можно выделить следующее:

- COBOL является самодокументируемым языком.
- Он похож на английский, поэтому его легко понимать и читать;
- Он может легко обрабатывать огромные объемы данных;
- Он широко используется в бизнес-приложениях, администрациях и прост в обслуживании.

Теперь рассмотрим недостатки:

- Чтобы стать программистом на COBOL, необходимо хорошо разбираться в машинных языках.
- Поскольку язык похож на английский, приходится много печатать, что делает код громоздким.
- Время компиляции программы на COBOL больше, чем у большинства машинных языков программирования.

Стоит отметить что один и тот же пункт присутствует и в достоинствах и недостатках, это касается написания кода. Казалось бы, делалось для удобства, но стало очевидным минусом.

1.4.3 Исследование языка программирования Fortran

Далее упомяну такой язык как Fortran.

Fortran до сих пор широко используется в областях высокопроизводительных вычислений, которые требуют использования компьютерных технологий. Среди известных примеров программ на Fortran - NEMO (Nucleus for European Modeling of the Ocean), OFF (Fluid Dynamics Code) и модели института космических исследований Годдарда(США).

Fortran также остается популярным среди различных программ для моделирования климата, поскольку математические формулы легко преобразовываются в код, программа работает быстро и большинство специалистов в данной области понимают этот язык. Несмотря на то, что в 21 веке популярность других языков программирования возросла, постоянное присутствие Fortran в области высокопроизводительных вычислений говорит о том, что он никуда не исчезнет.

Достоинством языка программирования Fortran является то, что до сих пор написано значительное количество программного обеспечения, для которого нет аналога.

Недостатком Fortran является возраст который дает о себе знать, и изучение такого языка может быть затруднительно, что и ограничивает функционал данного языка.

1.4.4 Исследование языка программирования Visual Basic

Дальше рассмотрим такой язык программирования как Basic.

Basic (beginner's all-purpose symbolic instruction code) — это универсальный символьный обучающий код для начинающих в области компьютерного программирования, разработанный в 1964 году Джоном Г. Кемени и Томасом Э. Курцем из университета Дартмут(США). Basic был первым языком, доступным для использования на персональных компьютерах (Microsoft выпустила свою версию), и в последние годы он вновь приобрел популярность под именем Visual Basic, хотя последний почти не похож на ранние версии Basic.

Программы, которые реализованы на платформе языка Basic имели репутацию очень медленных в начале развития персональных компьютеров. Эта медлительность была вызвана тем, что в те дни Basic был "интерпретируемым" языком, то есть каждый раз, когда вы запускали программу на основе Basic, вы на самом деле запускали интерпретатор,

который пошагово выполнял ваш код, преобразуя его на ходу в форму, понятную вашему компьютеру. Этот процесс требовал времени. В настоящее время существует много хороших компиляторов для Basic. Компилятор выполняет преобразование заранее и только один раз, превращая код в программу, которую ваша система может запустить напрямую и с максимальной скоростью. Таким образом, современный скомпилированный Basic проще в использовании и почти так же быстр, как язык программирования C.

Одним из преимуществ этого языка является то, что команды напоминают английский язык, что делает их более понятными и запоминаемыми, чем команды других языков. Несмотря на это, современные версии Basic способны выполнять практически все функции, доступные в таких языках программирования, как C или Pascal.

А теперь минусы:

– Visual Basic — мощный язык, но он не подходит для программирования. Он намного медленнее других языков. Это собственный язык программирования, написанный Microsoft, поэтому программы, написанные на Visual Basic, нелегко перенести в другие операционные системы.

– Приложения, разработанные на Visual Basic, работают только в операционных системах Windows и в основном поддерживаются только для приложений баз данных и графики.

1.4.5 Языки программирования пользующиеся спросом

На сегодняшний день языки программирования используются не только для создания и разработки программ и приложений разного функционала, но также и для создания веб-сайтов, управления робототехникой, разработки искусственного интеллекта, анализа данных, игр, мобильных приложений, а также для автоматизации бизнес-процессов и

многих других целей. Языки программирования становятся все более востребованными и широко применяются в различных сферах человеческой деятельности.

1.4.6 Язык программирования C# (Си Шарп)

Язык программирования C# - современный язык программирования общего назначения, разработанный Microsoft. Впервые он был представлен в начале 2000-х годов как часть платформы Microsoft .NET. C# задуман как универсальный язык, подходящий для разработки различных типов программного обеспечения, включая настольные приложения, веб-приложения, мобильные приложения, облачные сервисы и многое другое.

Язык C# позволяет разработчикам создавать эффективные и надежные приложения. От веб-приложений и настольных приложений до разработки игр и многого другого - C# поражает своей универсальностью.

Стоит отметить такие положительные качества языка как:

- Простое освоение данного языка;
- Наличие огромной библиотеки которую в случае чего можно подгружать в любое время;
- Удобный интерфейс.

Но даже у таких языков не обошлось без минусов:

- C# привязан к платформе «Windows», что ограничивает его пригодность для других платформ;
- Не смотря на возможность разработки приложений под платформу «Android», не хватает должного функционала.

1.4.7 Язык программирования Python

Python - это интерпретируемый язык. Это означает, что он преобразуется в машиночитаемый код не перед запуском программы, а во

время выполнения. В прошлом этот тип языка назывался языком сценариев, подразумевая, что его можно использовать для решения стандартных задач.

Python - это язык программирования общего назначения, который можно использовать в любой современной компьютерной операционной системе. Его можно использовать для обработки текста, чисел, изображений, научных данных и всего остального, что вы можете сохранить на компьютере.

Зачастую язык программирования Python сравнивают с такими языками как PHP или Ruby. Но в отличие от них, Python имеет ряд преимуществ:

- Читабельность кода;
- Универсальность;
- Интерпретируемость.

1.4.8 Язык программирования Java

Java — очень популярный язык программирования, который можно использовать для создания различных программных приложений. Это объектно-ориентированный язык, созданный таким образом, чтобы его было легко читать, писать и изучать. Миллионы разработчиков используют Java для создания всего: от настольных программ до сложных веб-приложений. Это делает этот навык полезным для всех, кто хочет работать в сфере технологий.

Одна из основных причин, по которой вам следует изучать Java, заключается в том, что это невероятно универсальный язык. Вы можете использовать Java для создания приложений практически для любой платформы, включая настольные компьютеры, мобильные устройства и Интернет.

Сильными сторонами можно выделить:

- Независимость платформы;

- Надежность;
- Масштабируемость.

2 ПРОЕКТНЫЙ РАЗДЕЛ. ВЫБОР СРЕДЫ РАЗРАБОТКИ

2.1 Определение среды разработки приложения

Начиная разработку приложения, я твердо был настроен писать код именно в Visual Studio на языке программирования C#. Но почему именно на нем, а не на Java, или любом другом популярном языке программирования? Все просто, C# мы изучали и активно пользовались им на многих предметах, и даже курсах обучения в университете. Сразу же и определился со средой разработки, им стал Windows Forms (.NET Framework). Так же на выбор был предоставлен и Windows Forms (Майкрософт), но он не обладал тем же функционалом.

2.2 Проектирование и разработка приложения

Перед непосредственным написанием кода для элементов: Button(кнопка), Label(этикетка), TextBox(текстовое поле) и прочих, я сразу задался вопросом, можно ли сделать фоновое изображение с возможностью менять его, так как стандартные были невзрачными и однотонными. Для этого я создал отдельный проект с классификацией «Библиотека классов»(см. рис. 7), что в последующем, позволило добавить собственный элемент в общий список.

```

7  using System.Threading.Tasks;
8  using System.Windows.Forms;
9
10 namespace ClassLibrary1
11 {
12     Ссылка 1
13     public class ArtanPanel : Panel
14     {
15         private int borderRadius = 30;
16         private float gradientAngle = 90F;
17         private Color gradientTopColor = Color.DodgerBlue;
18         private Color gradientBottomColor = Color.DodgerBlue;
19
20         Ссылка 0
21         public ArtanPanel()
22         {
23             this.BackColor = Color.White;
24             this.ForeColor = Color.Black;
25             this.Size = new Size(350, 200);
26         }
27
28         Ссылка 0
29         public int BorderRadius...
30         Ссылка 0
31         public float GradientAngle...
32         Ссылка 1
33         public Color GradientTopColor...
34         Ссылка 1
35         public Color GradientBottomColor...
36
37         Ссылка 1
38         private GraphicsPath GetArtanPath(RectangleF rectangle, float radius)...
39
40         Ссылка 0
41         protected override void OnPaint(PaintEventArgs e)...
42     }
43 }

```

Рисунок 7 – Составляющие кода для элемента «Label»

С помощью данных элементов можно внести что-то новое в основной код. Вот так на рисунке 8 выглядит применяемый элемент.



Рисунок 8 – Элементы «ArtanComponent»

Далее я приступил к выполнению основной идеи по созданию приложения для учета рабочего времени. Во-первых, необходимо было разметить интерфейс для удобства, чем я и занялся(см. рис. 9), а далее уже добавлял необходимый функционал.

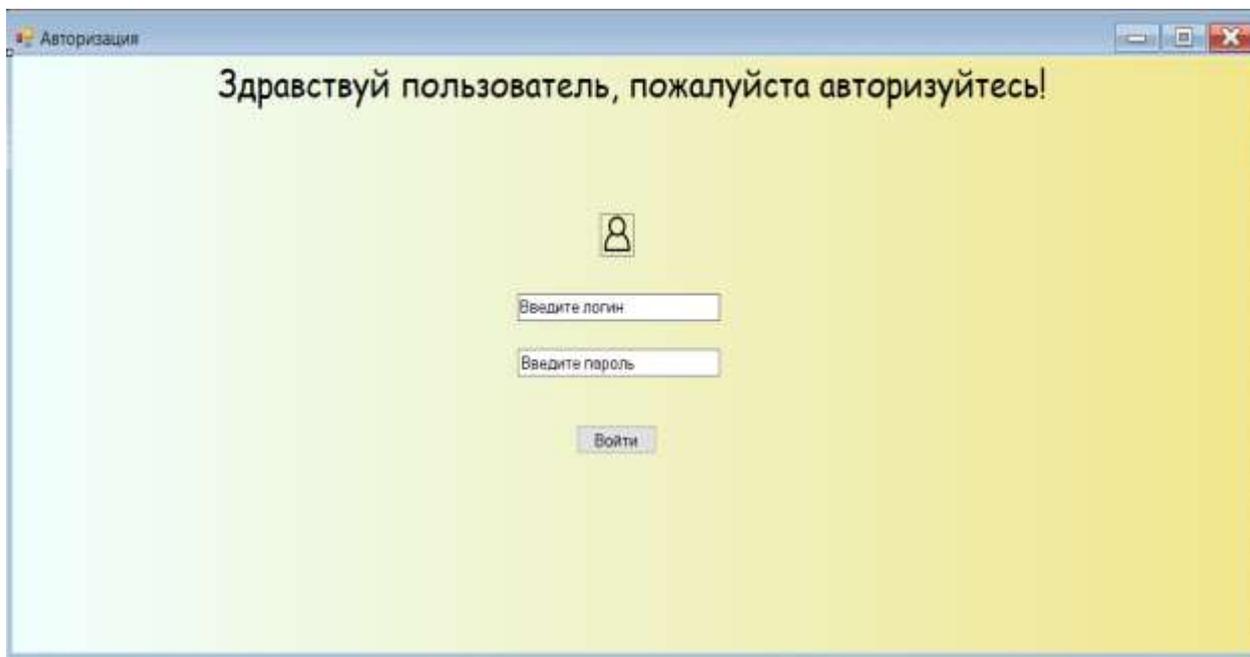


Рисунок 9 – Окно авторизации

Стоило учесть тот факт, что за компьютером не всегда мог находиться необходимый пользователь, поэтому была сделана простая, но нужная защита.(см. рис. 10)

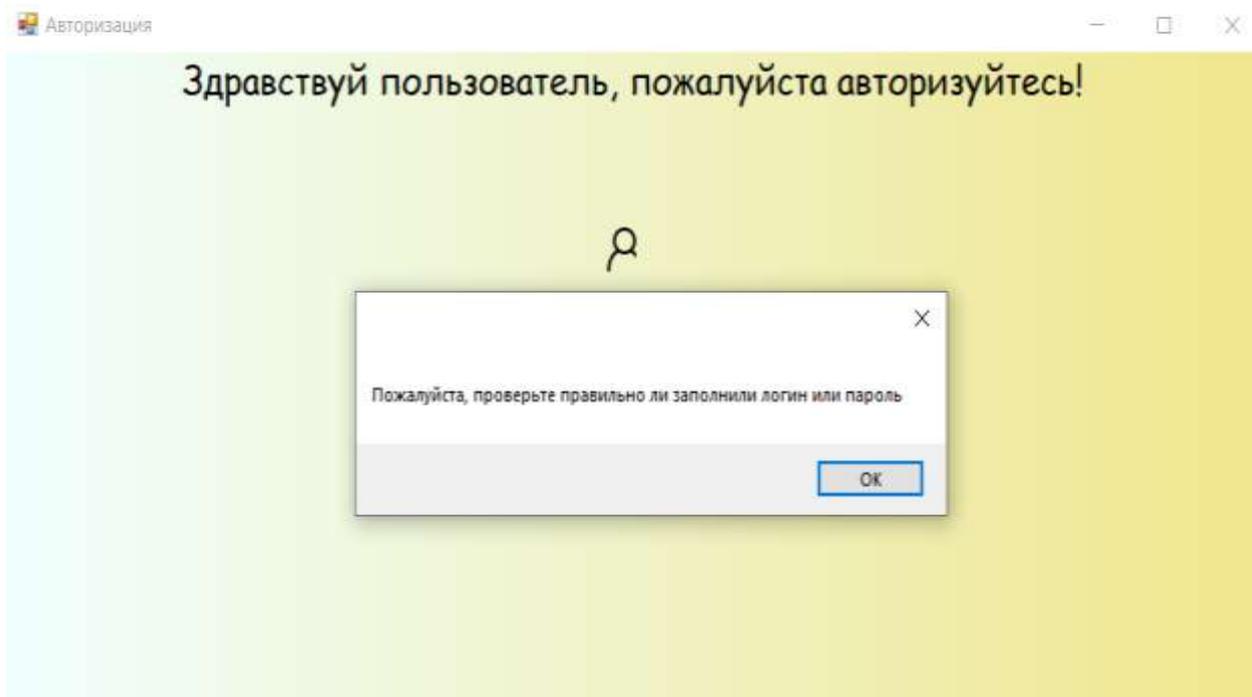


Рисунок 10 – Предупреждение о неправильности введения логина или пароля

После введения правильного логина и пароля, пользователь попадает в главное меню, откуда уже может совершать ряд нужных ему действий.(см. рис. 11) Такие как внесение нового сотрудника в базу данных, заполняя все необходимое. Паспортные данные, адрес проживания, индекс, номер мобильного телефона, должность.

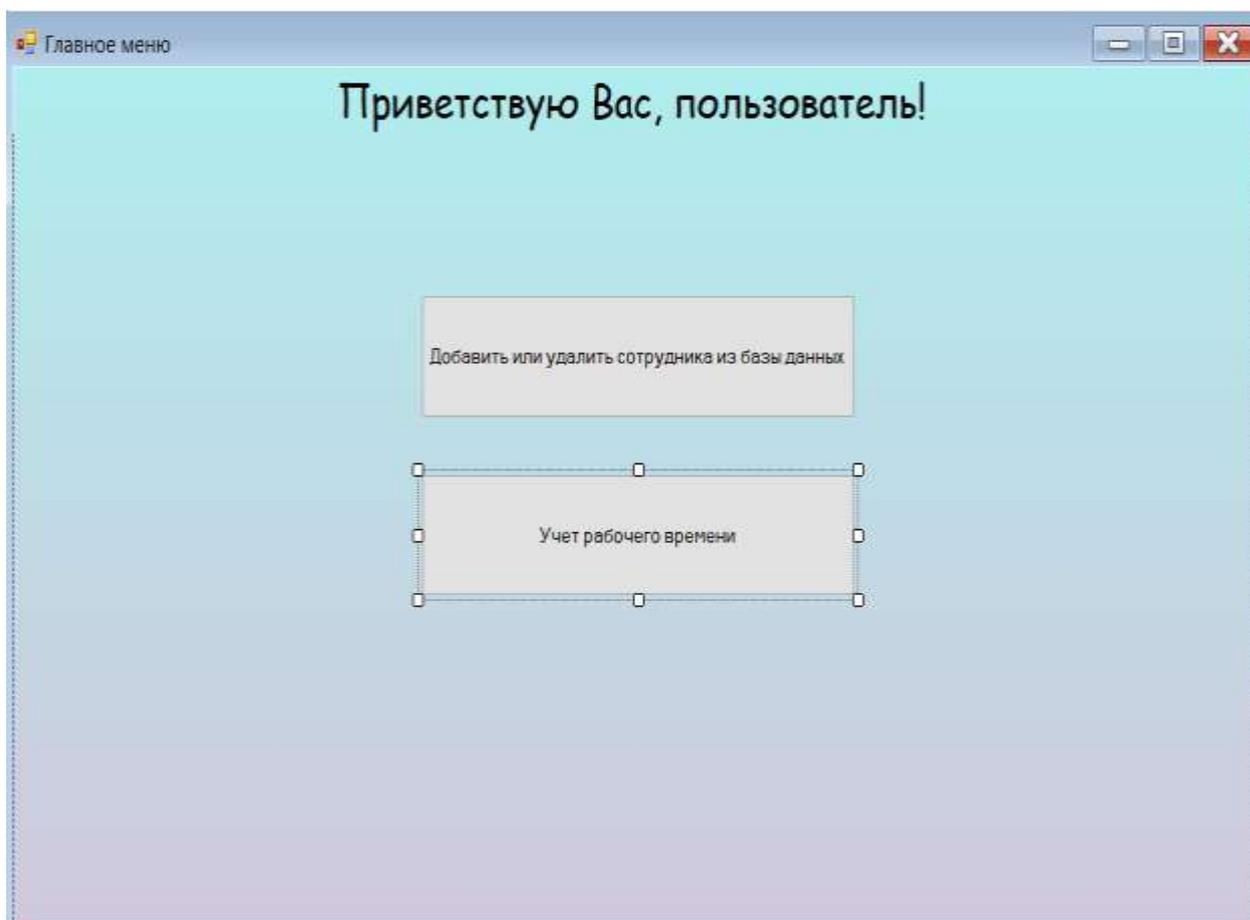


Рисунок 11 – Главное меню с возможностью перехода

Из главного меню можно перейти в новое окно, добавления и удаления сотрудника из БД. Там вы увидите несколько окон для внесения информации о сотруднике, после внесения сотрудника в БД, вы так же можете взаимодействовать с двумя списками и соответствующим им кнопкам. И вы всегда можете вернуться в главное меню, через кнопку назад в нижнем правом углу окна программы.

 Добавить или удалить сотрудника

Иванов Иван Иванович

г. Липецк, ул. 9 Мая, д. 10, кв 56

+79804975614

398000

Слесарь

Добавить в базу данных

Рисунок 12 – Добавление нового сотрудника

Иванов Иван Иванович

г. Липецк, ул. 9 Мая, д. 10, кв 56

+79804975614

398000

Слесарь

Добавить в базу данных

Операция выполнена



Сотрудник успешно добавлен

ОК

Рисунок 13 – Диалоговое окно о выполненной операции по добавлению сотрудника

После добавление в базу данных сотрудника, программа сообщит на об этом. Так же, если вы случайно нажмете повторно на кнопку добавления, программа сообщит что такой сотрудник уже добавлен.(см. рис 14)

Добавить или удалить сотрудника

Иванов Иван Иванович

г. Липецк, ул. 9 Мая, д. 10, кв 56

+79804975614

398000

Слесарь

Добавить в базу данных

Внимание! ×

Сотрудник уже добавлен

OK

Рисунок 14 – Предупреждение о уже имеющемся сотруднике

Теперь у нас есть на выбор просмотреть всю информацию о добавленных сотрудниках, либо же удалить их.

Добавить или удалить сотрудника

Иванов Иван Иванович

Пожалуйста, выберите пользователя

Пожалуйста, выберите сотрудника

г. Пятик., ул. 9 Мая, д. 10, кв 56

+79804975614

398000

Слесарь

Показать информацию

Удалить сотрудника

Добавить в базу данных

Назад

Рисунок 15 – Окно «Добавить или удалить сотрудника»

При желании посмотреть информацию, нажимая на кнопку «Показать информацию», откроется новое окно с желаемыми данными.(см .рис. 16, 17)

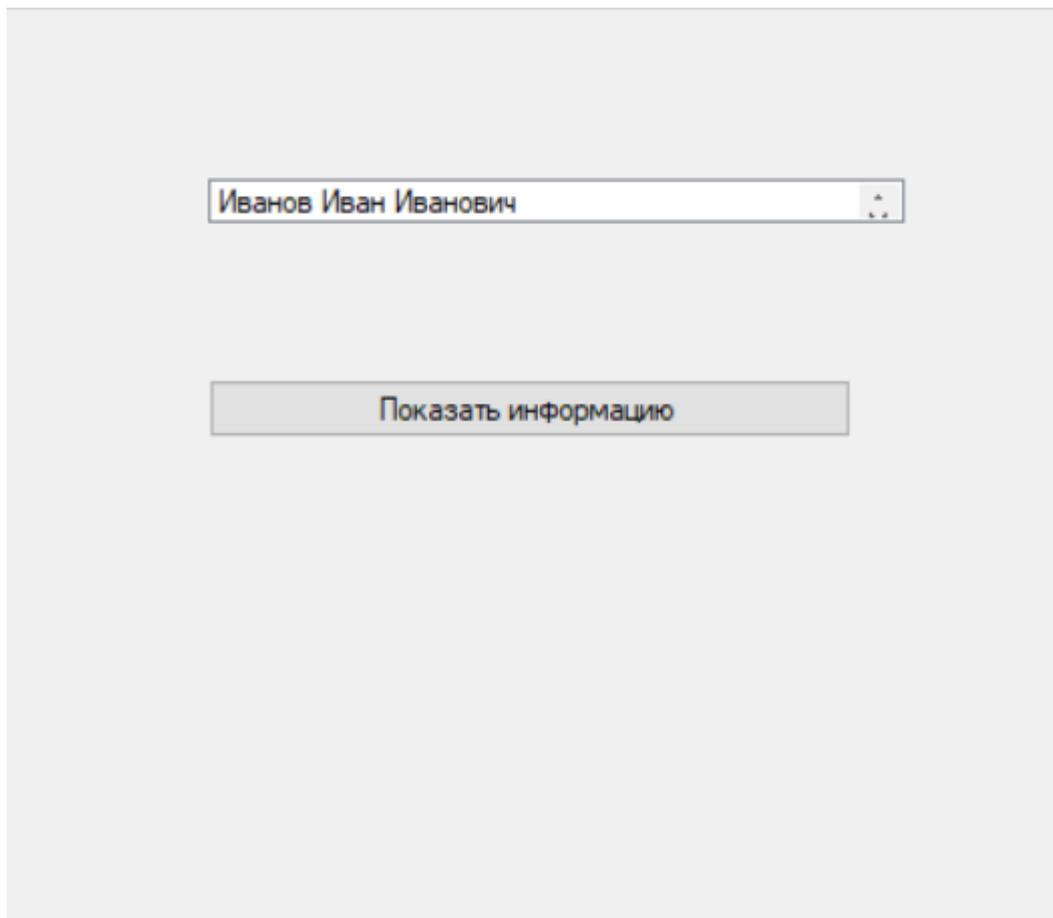


Рисунок 16 – Выбор сотрудника

 Добавить или удалить сотрудника

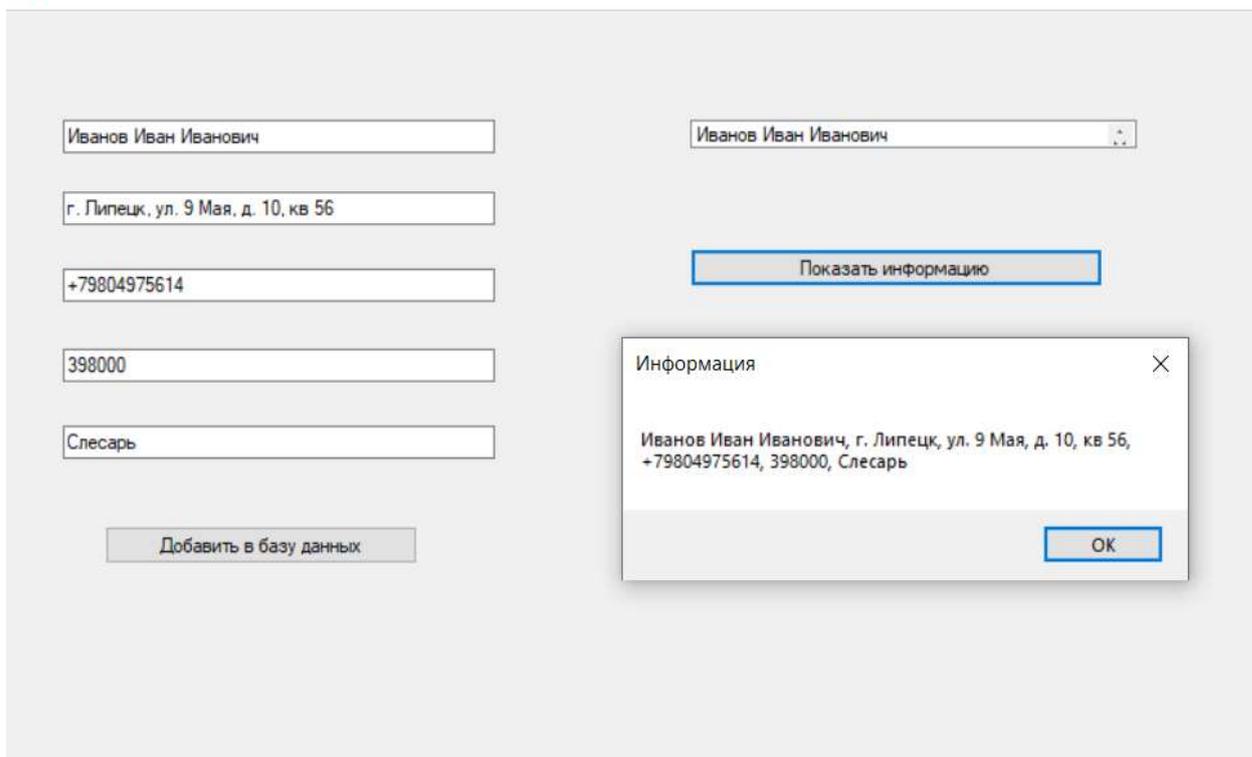


Рисунок 17 – Информация о сотруднике добавленном ранее

Теперь у нас осталась лишь кнопка «Удалить сотрудника». При её нажатии программа уточнит ваши намерения.(см. рис. 18) Так же могу сказать что это базовая защита от мгновенно удаления чего либо безвозвратно.

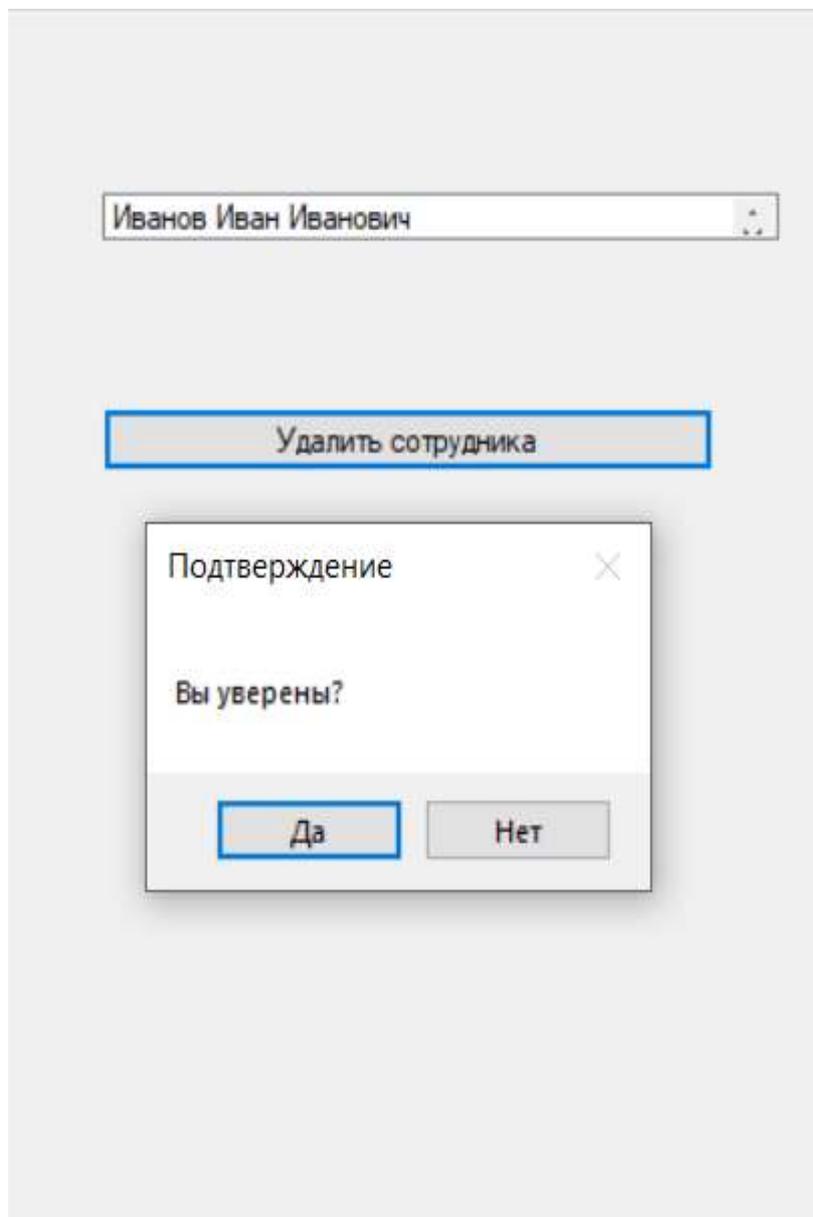


Рисунок 18 – Информационное окно с предложением выбора

В случае если вы все таки случайно нажали, вы можете отменить операцию, выбрав в новом окне кнопку «Нет». Далее последует еще одно окно.(см. рис. 19)

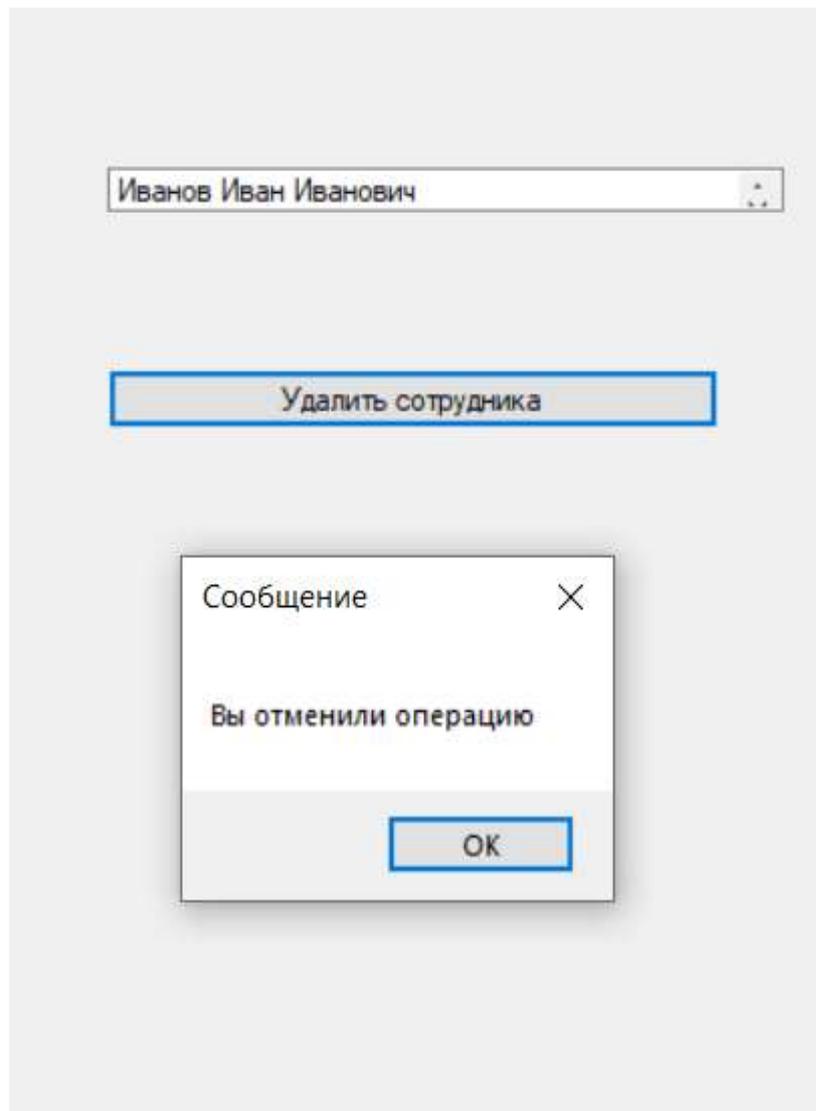


Рисунок 19 – Отмена операции по удалению сотрудника из БД

Но если вы все таки решили удалить сотрудника, после выбора кнопки «Да» на рисунке 18, программа выдаст вам следующее.(см. рис. 20)

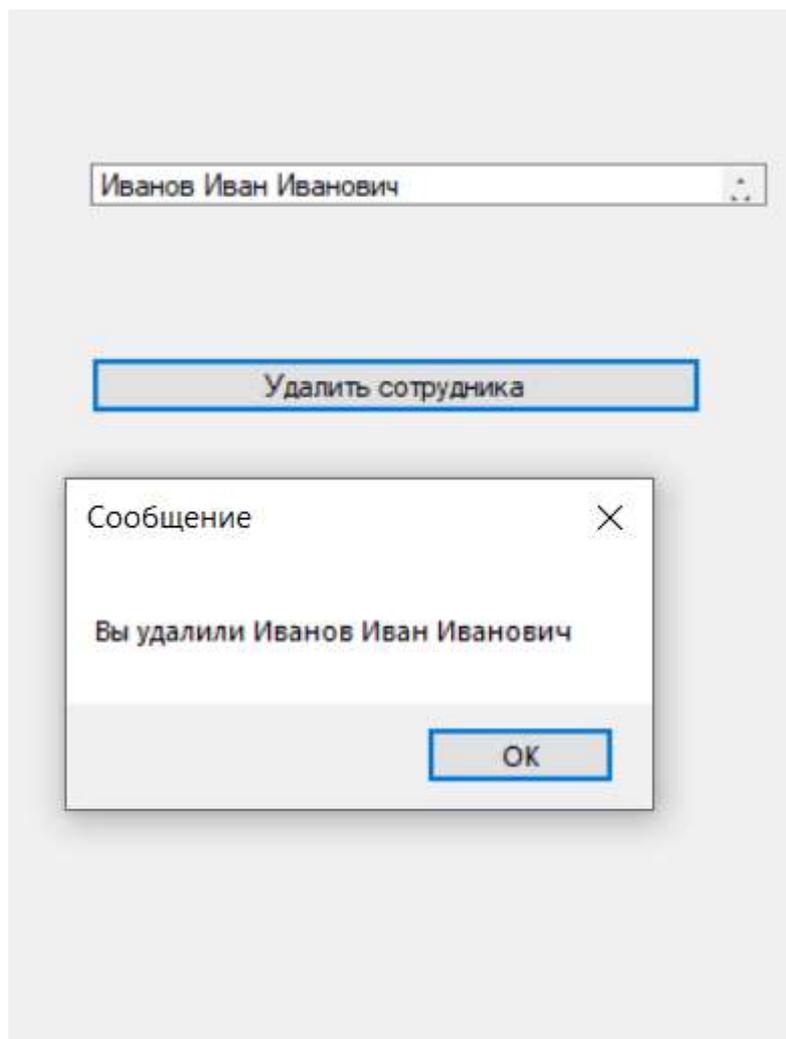


Рисунок 20 – Сообщение об успешном удалении сотрудника из БД

Естественно после такого, в пункте показать информацию, при наличие одного сотрудника, программа сообщит вам следующее.(см. рис. 21)

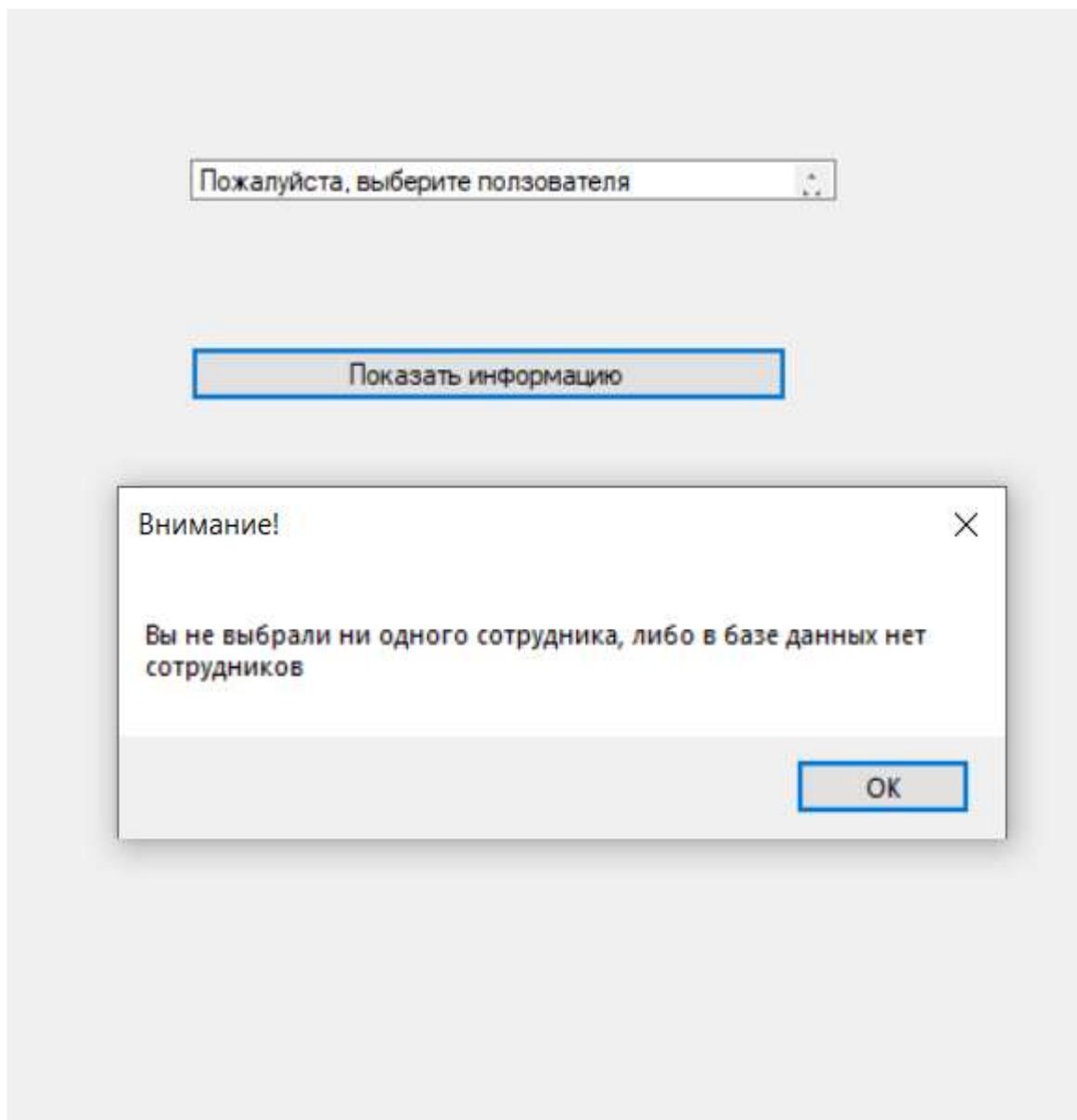


Рисунок 21 – Сообщение об отсутствии сотрудников

Но теперь вы снова можете добавить сотрудника, которого удалили, как это уже было показано на рисунке 13.

Так же у нас остался еще один раздел, это «учет рабочего времени», кое вы видели на рисунке 11.

Что ж, при переходе в новое окно, вы можете увидеть уже знакомый список, предлагающий вас выбрать сотрудника, также выбрать количество часов, соответственно, часы не могут уйти в минус, так как такого просто не может быть. Вы можете взаимодействовать с тремя кнопками: «Внести отработанные часы», «Показать информацию», «Назад».

Учет рабочего времени

Иванов Иван Иванович

Выберите количество часов

Июнь 2024

Пн	Вт	Ср	Чт	Пт	Сб	Вс
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

Сегодня: 17.06.2024

Рисунок 22 – Пример для внесения отработанных часов

В случае если вы не вносили за этот месяц, либо будете делать это первый раз, то после нажатия «Показать информацию», программа вас уведомит, что на данный момент информации нет.(см. рис. 23)

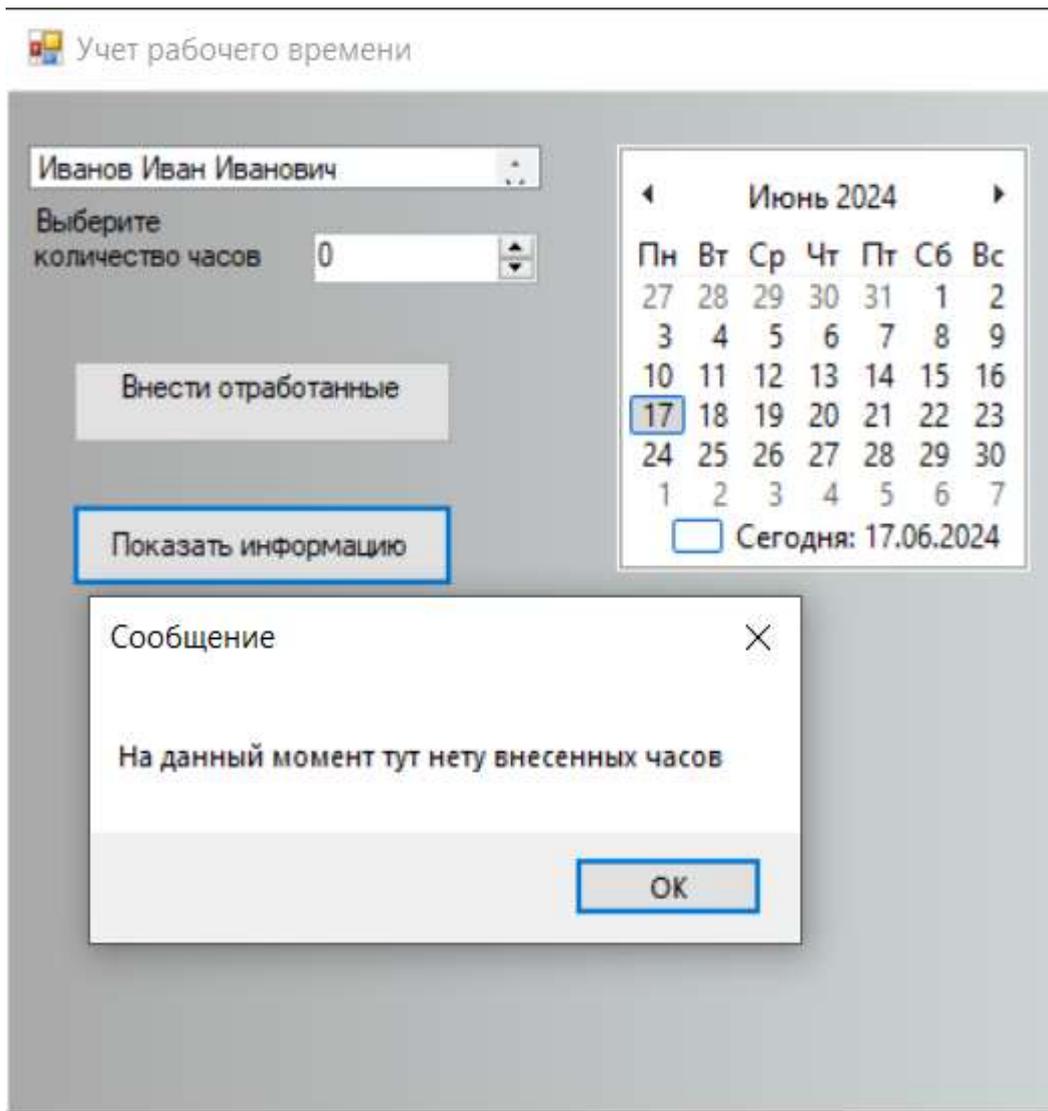


Рисунок 23 – Сообщение об отсутствие данных

Что же будет если мы внесем данные нашему сотруднику, они запишутся и сохранятся, что вы сразу же сможете проверить.(см. рис. 24)

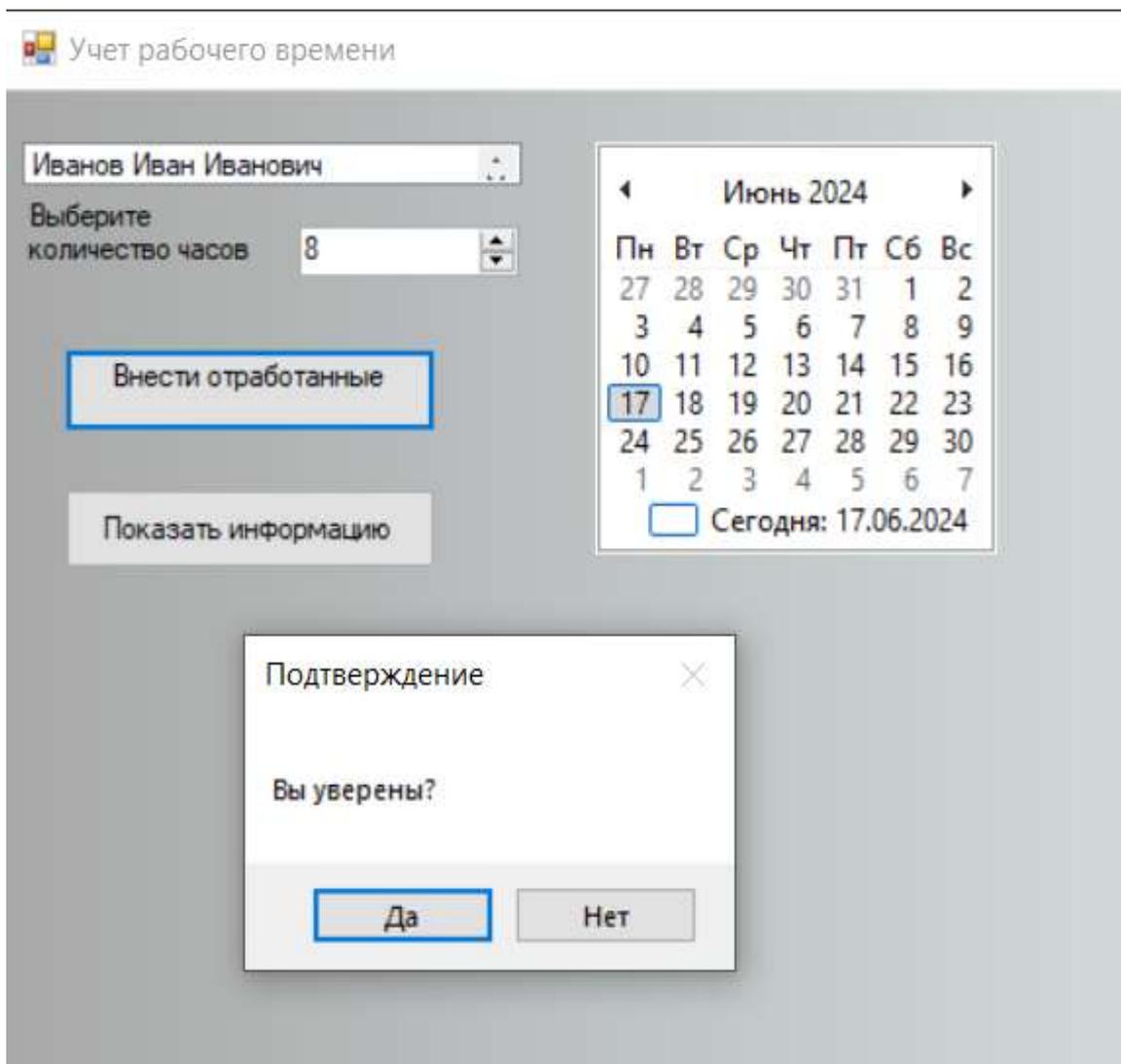


Рисунок 24 – Ожидание подтверждения операции

Здесь такая же система, как и в прошлом модуле «Добавление и удаление сотрудников», вас спросят хотите ли внести данные. Если вы соглашаетесь вы увидите следующее.(см. рис. 25)

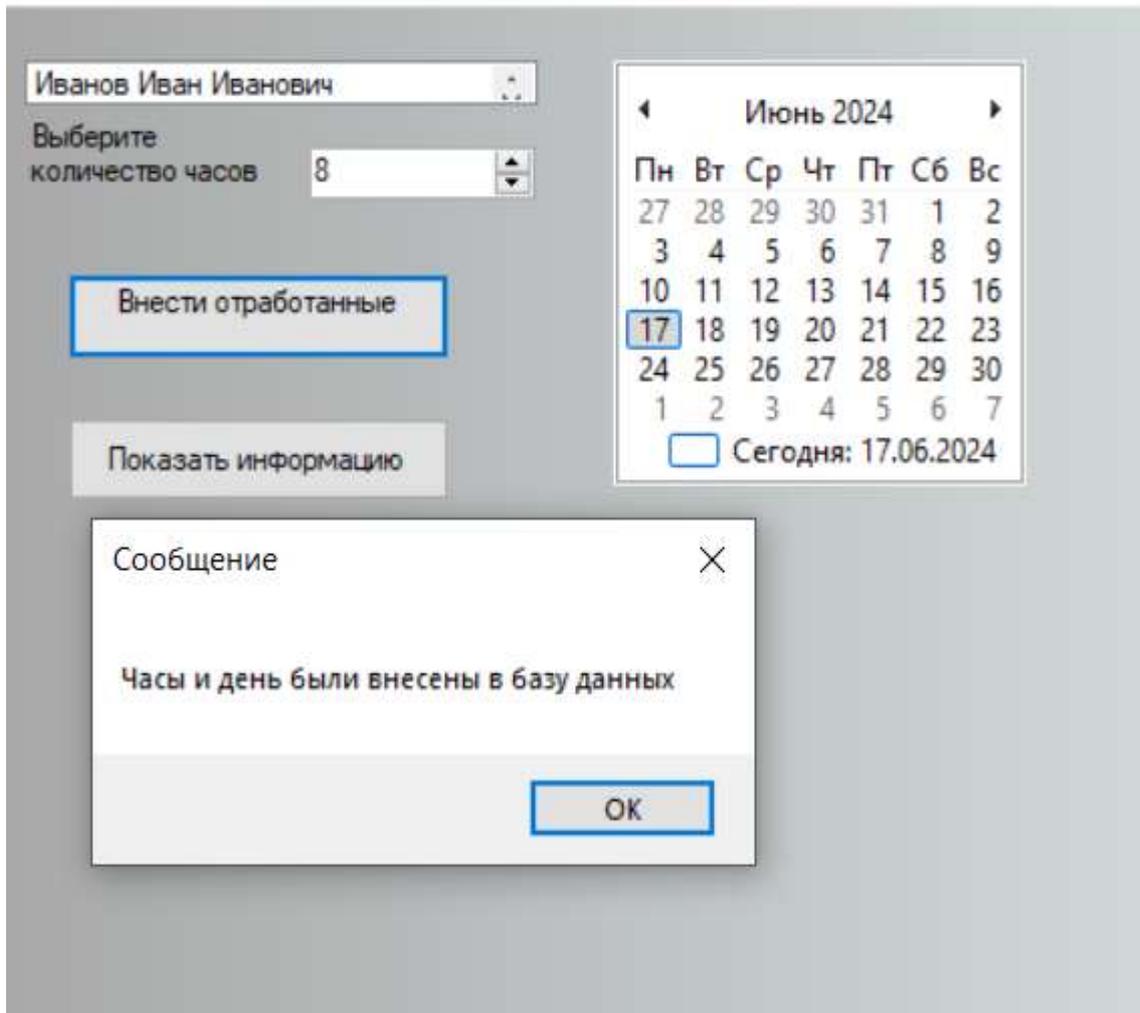


Рисунок 25 – Сообщение о внесении часов в базу данных

Давайте теперь проверим это через кнопку «Показать информацию».(см. рис. 26)

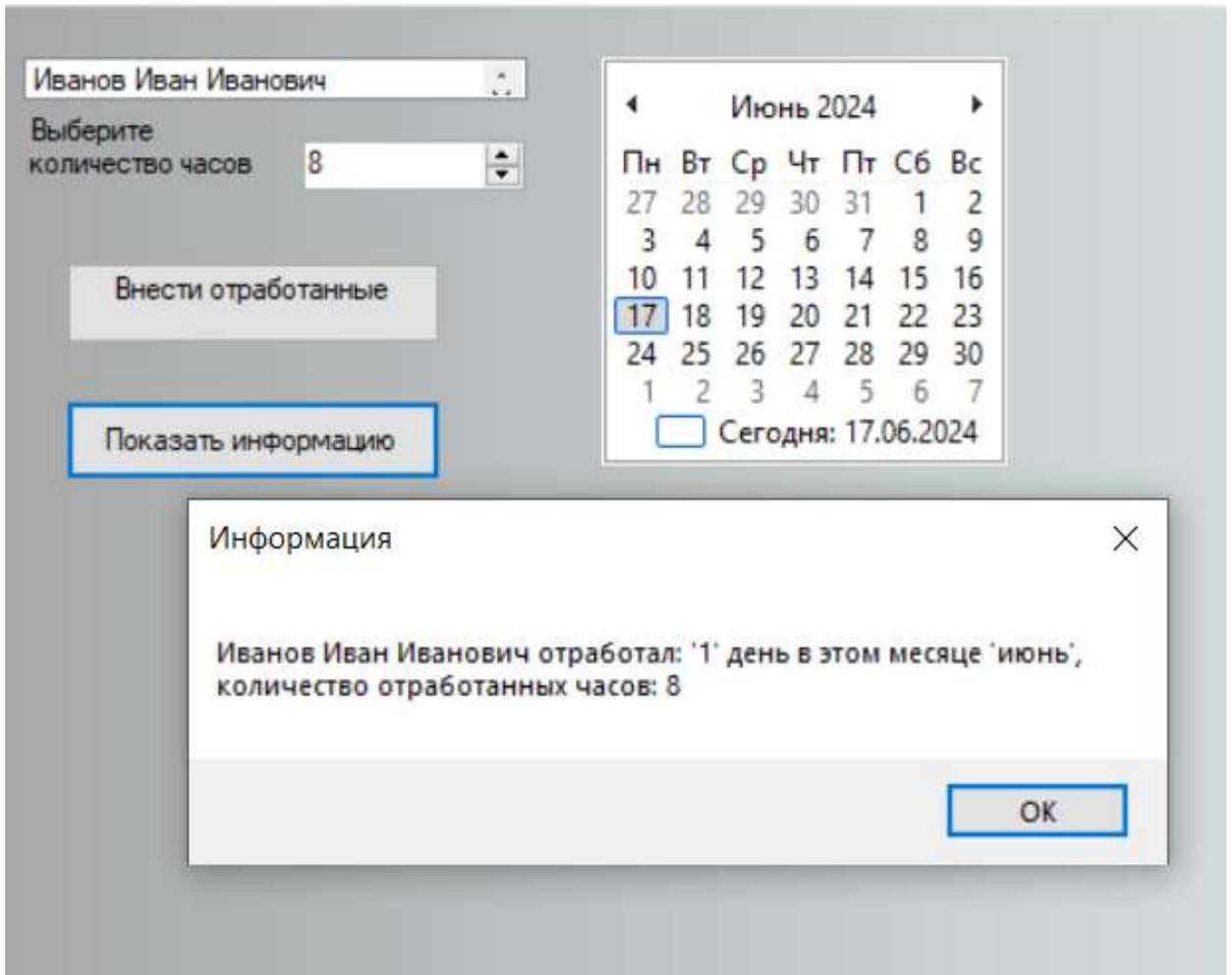


Рисунок 26 – Информация об отработанных часах сотрудника

ЗАКЛЮЧЕНИЕ

Цель данной дипломной работы выполнена полностью, благодаря решению следующих задач:

– В процессе разработки были изучены существующие методы учета рабочего времени, проведен анализ их преимуществ и недостатков, а также изучено трудовое законодательство Российской Федерации в сфере учета рабочего времени.

– Проведен анализ существующих систем и приложений применяемых для учета рабочего времени. Было изучено несколько примеров программ, которые в настоящее время занимают ведущие позиции в сфере учета и автоматизации рабочего времени.

– Проведен анализ программного обеспечения, используемое предприятием ООО «АТП №1». На данный момент удобство и эффективность программы 1С: Бухгалтерия предприятия является очень даже неплохим аналогом прочим программам в этой сфере.

Результатом дипломной работы стало создание функционального и удобного в использовании приложения, которое позволяет учитывать рабочее время каждого сотрудника, анализировать производительность и эффективность труда. Благодаря всем этим возможностям разработанное мною приложение, позволяет организациям, на которых она будет развернута, значительно уменьшить рутинную работу отдела кадров и бухгалтерии.

В дальнейшем развитии моего приложения, предполагается добавление новых функциональных возможностей, улучшение интерфейса и повышение стабильности работы последнего.

Таким образом, данное приложение может стать незаменимым инструментом для учета рабочего времени сотрудников и повышения эффективности работы организации в целом.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Балдин, К. В. Информационные системы в экономике : Учебное пособие / Российская таможенная академия. - 1-е изд. - Москва : ООО "Научно-издательский центр ИНФРА-М", 2024. - 218 с. - (Высшее образование). - ISBN 978-5-16-019321-2. - Текст : непосредственный. - URL: <https://znanium.com/catalog/document?id=435544> (дата обращения: 30.05.2024).

2. Варфоломеева, А. О. Информационные системы предприятия : Учебное пособие / Российский экономический университет им. Г.В. Плеханова. - 2-е изд., перераб. и доп. - Москва : ООО "Научно-издательский центр ИНФРА-М", 2024. - 330 с. - (Высшее образование: Бакалавриат). - ISBN 978-5-16-012274-8. - Текст : непосредственный. - URL: <https://znanium.ru/catalog/document?id=446764> (дата обращения: 30.05.2024).

3. Плахотникова, М. А. Информационные технологии в менеджменте : учебник и практикум для вузов / М. А. Плахотникова, Ю. В. Вертакова. - 2-е изд., пер. и доп. - Москва : Юрайт, 2024. - 326 с. - (Высшее образование). - ISBN 978-5-534-07333-1. - Текст : непосредственный. - URL: <https://urait.ru/bcode/535632> (дата обращения: 30.05.2024).

4. Трофимов, В. В. Информационные технологии в экономике и управлении : учебник для вузов / В. В. Трофимов [и др.] ; ответственный редактор В. В. Трофимов. - 4-е изд., пер. и доп. - Москва : Юрайт, 2024. - 556 с. - (Высшее образование). - ISBN 978-5-534-18678-9. - Текст : непосредственный. - URL: <https://urait.ru/bcode/545322> (дата обращения: 30.05.2024).

5. Черников, Б. В. Информационные технологии управления : Учебник / Российский экономический университет им. Г.В. Плеханова. - 2-е изд., перераб. и доп. - Москва : Издательский Дом "ФОРУМ", 2024. - 368 с. - (Высшее образование: Бакалавриат). - ISBN 978-5-8199-0782-5. - Текст : непосредственный. -

URL: <https://znanium.com/catalog/document?id=438483> (дата обращения: 30.05.2024).

6. Одинцов, Б. Е. Современные информационные технологии в управлении экономической деятельностью (теория и практика) : Учебное пособие / Финансовый университет при Правительстве Российской Федерации. - 1-е изд. - Москва : Вузовский учебник, 2023. - 373 с. - ISBN 978-5-9558-0517-7. - Текст : непосредственный. -

URL: <http://znanium.com/catalog/document?id=415448> (дата обращения: 30.05.2024).

7. Балдин, К. В. Информационные системы в экономике : Учебник / Российская таможенная академия. - 10-е изд. - Москва : Издательско-торговая корпорация "Дашков и К", 2022. - 394 с. - ISBN 978-5-394-04783-1. - Текст : непосредственный. -

URL: <https://znanium.com/catalog/document?id=431494> (дата обращения: 30.05.2024).

8. Карпузова, В. И. Информационные технологии в менеджменте : Учебное пособие / Российский государственный аграрный университет - МСХА им. К.А. Тимирязева. - 2-е изд., доп. - Москва : Вузовский учебник, 2021. - 301 с. - (ВО - Бакалавриат). - ISBN 978-5-9558-0315-9. - Текст. Электронный. - URL: <https://znanium.com/catalog/product/1047207> (дата обращения: 30.05.2024). - Режим доступа: по подписке.

9. Дмитриева, И. М. Бухгалтерский учет с основами МСФО : учебник и практикум для вузов / И. М. Дмитриева. - 7-е изд., пер. и доп. - Москва : Юрайт, 2024. - 304 с. - (Высшее образование). - ISBN 978-5-534-16323-0. - Текст : непосредственный. - URL: <https://urait.ru/bcode/535647> (дата обращения: 04.06.2024).

10. Проданова, Н. А. Бухгалтерский учет на предприятиях малого бизнеса : учебное пособие для вузов / Н. А. Проданова, Е. И. Зацаринная ; ответственный редактор Е. И. Зацаринная. - 3-е изд., пер. и доп. - Москва : Юрайт, 2024. - 220 с. - (Высшее образование). - ISBN 978-5-534-15782-6. -

Текст : непосредственный. - URL: <https://urait.ru/bcode/537169> (дата обращения: 04.06.2024).

11. Мизиковский, И. Е. Бухгалтерский учет, анализ и аудит в кредитной сфере при цифровизации экономики / Мизиковский И. Е., Дружиловская Т. Ю., Дружиловская Э. С. - Нижний Новгород : ННГУ им. Н. И. Лобачевского, 2023. - 284 с. - Текст : непосредственный. - URL: <https://e.lanbook.com/book/344771> (дата обращения: 04.06.2024).

12. Штефан, М. А. Основы аудита : учебник и практикум для вузов / М. А. Штефан, О. А. Замотаева, Н. В. Максимова ; под общей редакцией М. А. Штефан. - 3-е изд., пер. и доп. - Москва : Юрайт, 2023. - 313 с. - (Профессиональное образование). - ISBN 978-5-534-16651-4. - Текст : непосредственный. - URL: <https://urait.ru/bcode/531440> (дата обращения: 04.06.2024).

13. Титова, А. В. Бухгалтерский учет, анализ и аудит. Тесты. Тест 1. Основы бухгалтерского учета / А. В. Титова ; Урал. гос. экон. ун-т, Каф. бухгалт. учета и аудита. - Екатеринбург : [б. и.], 2022. - 5 с. - Текст : электронный. - URL: <http://lib.wbstatic.usue.ru/202212b/20.docx> (дата обращения: 04.06.2024). - Режим доступа: для авторизованных пользователей.

14. Титова, А. В. Бухгалтерский учет, анализ и аудит. Тесты. Тест 2. Методы и приемы анализа хозяйственной деятельности / А. В. Титова ; Урал. гос. экон. ун-т, Каф. бухгалт. учета и аудита. - Екатеринбург : [б. и.], 2022. - 4 с. - Текст : электронный. - URL: <http://lib.wbstatic.usue.ru/202212b/18.docx> (дата обращения: 04.06.2024). - Режим доступа: для авториз. пользователей.

15. Харченко, О. Н. Бухгалтерский учет, налоги, анализ и аудит. Оценочные материалы с решениями : Учебное пособие / Сибирский федеральный университет. - Красноярск : Сибирский федеральный университет, 2021. - 384 с. - (ВО - Бакалавриат). - ISBN 978-5-7638-4246-3. - Текст. Электронный. - URL: <https://znanium.com/catalog/product/1830746> (дата обращения: 04.06.2024). - Режим доступа: по подписке.

16. Молинаро, Э. SQL. Сборник рецептов. Решения и методики построения запросов для разработчиков баз данных : производственно-практическое издание / Энтони Молинаро, Роберт де Грааф. - 2-е изд. - Санкт-Петербург : БХВ-Петербург, 2023. - 592 с. - ISBN 978-5-9775-6759-6. - Текст : непосредственный.

17. Мамонова, Т. Е. Информационные технологии. Лабораторный практикум : Учебное пособие для вузов / Мамонова Т. Е. - Москва : Юрайт, 2022. - 176 с. - (Высшее образование). - ISBN 978-5-9916-7060-9. - Текст. Электронный. - URL: <https://urait.ru/bcode/490340> (дата обращения: 06.06.2024). - Режим доступа: по подписке.

18. Баранова, Е. К. Основы информатики и защиты информации : Учебное пособие / Национальный исследовательский университет "Высшая школа экономики". - 1-е изд. - Москва : Издательский Центр РИОР, 2021. - 183 с. - (ВО - Бакалавриат). - ISBN 978-5-369-01169-0. - Текст. Электронный. - URL: <https://znanium.com/catalog/product/959916> (дата обращения: 06.06.2024). - Режим доступа: по подписке.

19. Плещев, В. В. Основы программирования на языках C++ и C# с примерами и упражнениями : учебное пособие / В. В. Плещев, Е. И. Шишков ; М-во образования и науки Рос. Федерации, Урал. гос. экон. ун-т. - Екатеринбург : [Издательство УрГЭУ], 2021. - 286 с. - Текст : непосредственный. - URL: <http://lib.usue.ru/resource/limit/ump/18/p490708.pdf> (дата обращения:).

20. Казанский, А. А. Программирование на Visual C# : учебное пособие для вузов / А. А. Казанский. - 2-е изд., пер. и доп. - Москва : Юрайт, 2023. - 192 с. - (Высшее образование). - ISBN 978-5-534-12338-8. - Текст : непосредственный. - URL: <https://urait.ru/bcode/512404> (дата обращения: 06.06.2024).

21. Огнева, М. В. Программирование на языке C++: практический курс : учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. - Москва : Юрайт, 2023. - 335 с. - (Высшее образование). - ISBN 978-5-534-05123-0. - Текст :

непосредственный. - URL: <https://urait.ru/bcode/515142> (дата обращения: 06.06.2024).

22. Подбельский, В. В. Программирование. Базовый курс C# : учебник для вузов / В. В. Подбельский. - Москва : Юрайт, 2023. - 369 с. - (Высшее образование). - ISBN 978-5-534-10616-9. - Текст : непосредственный. - URL: <https://urait.ru/bcode/511747> (дата обращения: 06.06.2024).

23. Трофимов, В. В. Информатика : учебник для вузов / В. В. Трофимов [и др.] ; ответственный редактор В. В. Трофимов. - 4-е изд., пер. и доп. - Москва : Юрайт, 2023. - 795 с. - (Высшее образование). - ISBN 978-5-534-17577-6. - Текст : непосредственный. - URL: <https://urait.ru/bcode/533353> (дата обращения: 06.06.2024).

24. Тузовский, А. Ф. Объектно-ориентированное программирование : учебное пособие для вузов / А. Ф. Тузовский. - Москва : Юрайт, 2023. - 213 с. - (Высшее образование). - ISBN 978-5-534-16316-2. - Текст : непосредственный. - URL: <https://urait.ru/bcode/530800> (дата обращения: 06.06.2024).

25. Гуриков, С. Р. Введение в программирование на языке Visual C# : Учебное пособие / Московский технический университет связи и информатики. - 1-е изд. - Москва : Издательство "ФОРУМ", 2021. - 447 с. - (ВО - Бакалавриат). - ISBN 978-5-00091-458-8. - Текст. Электронный. - URL: <https://znanium.com/catalog/product/1092167> (дата обращения: 06.06.2024). - Режим доступа: по подписке.

```

CREATE TABLE [dbo].[Members] (
[member_id] INT NOT NULL,
[last_name] VARCHAR (20) NULL,
[first_name] VARCHAR (20) NULL,
[age] INT NULL,
[rank] VARCHAR (3) NULL,
[rank_value] INT NULL,
[sensei] VARCHAR (20) NULL,
[dojo] VARCHAR (20) NULL,
PRIMARY KEY CLUSTERED ([member_id] ASC) );

```

```

public string conString = "Data
Source=(LocalDB)\\MSSQLLocalDB;AttachDbFilename=C:\\dip\\
Desktop\\budaa\\staff_members.mdf; Integrated Security = True;
Connect Timeout = 30";

```

```

private void button1_Click(object sender, EventArgs e)
{
    SqlConnection con = new SqlConnection(conString);

    con.Open();

    if(con.State == System.Data.ConnectionState.Open)
    {
        string q = "insert into Members(member_id,
last_name, first_name," +
        " age, rank, rank_value, sensei, dojo)" +
        " values('" + textBox1.Text.ToString() +
        "','" + textBox2.Text.ToString() + "','" +
textBox3.Text.ToString() + "','" + textBox4.Text.ToString() +
        "','" + comboBox2.Text.ToString() + "','" +
comboBox1.Text.ToString() + "','" + comboBox3.Text.ToString() +
        "','" + textBox8.Text.ToString() + "') ";
        SqlCommand cmd = new SqlCommand(q, con);

```

```

        cmd.ExecuteNonQuery();
        MessageBox.Show("Connection was successful!");
    }

    con.Close();
}

private void button3_Click(object sender, EventArgs e)
{
    this.Visible = false;
    Form3 form3= new Form3();
    form3.ShowDialog();
}

private void button3_Click(object sender, EventArgs e)
{
    this.Visible = false;
    Form3 form3= new Form3();
    form3.ShowDialog();
}

SqlConnection con = new SqlConnection("Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\dip\
Desktop\budaa\staff_members.mdf;Integrated
Security=True;Connect Timeout=30");
SqlCommand cmd;
SqlDataAdapter adapt;

con.Open();
DataTable dt = new DataTable();
adapt = new SqlDataAdapter("select * from Members", con);
adapt.Fill(dt);
dataGridView1.DataSource = dt;
con.Close();

private void button1_Click(object sender, EventArgs e)

```

```

        {
            string SqlText;

            SqlText = "SELECT " + textBox1.Text.ToString() + "
FROM Members WHERE " + textBox2.Text.ToString() + " ORDER BY " +
textBox3.Text.ToString();

            SqlDataAdapter da = new SqlDataAdapter(SqlText,
conString);

            DataSet ds = new DataSet();
            da.Fill(ds, "[Members]");

            dataGridView1.DataSource =
ds.Tables["[Members]"].DefaultView;
        }

public string conString = "Data
Source=(LocalDB)\\MSSQLLocalDB;AttachDbFilename=C:\\dip\\
Desktop\\budaa\\staff_members.mdf;Integrated Security=True;
Connect Timeout=30";
SqlConnection con = new SqlConnection("Data
Source=(LocalDB)\\MSSQLLocalDB;AttachDbFilename=
C:\\dip\\
Desktop\\budaa\\staff_members.mdf;Integrated Security=True;
Connect Timeout=30");
SqlCommand cmd;
SqlDataAdapter adapt;

private void Form7_Load(object sender, EventArgs e)
{
    this.membersTableAdapter.Fill(this.iOGKF_membersData
Set.Members);
    con.Open();
    DataTable dt = new DataTable();
    adapt = new SqlDataAdapter("select * from Members",
con);

```

```

        adapt.Fill(dt);
        dataGridView1.DataSource = dt;
        con.Close();
    }

private void button1_Click(object sender, EventArgs e)
{
    if (textBox1.Text != "" && textBox2.Text != "" &&
        textBox3.Text != "" && textBox4.Text != "" && textBox5.Text !=
        "" && textBox6.Text != "" && textBox7.Text != "" &&
        textBox8.Text != "")
    {
        cmd = new SqlCommand("insert into
Members(member_id,last_name,first_name,age,rank,rank_value,sensei,dojo)
values(@member_id,@last_name,@first_name,@age,@rank,@rank_value,
@sensei,@dojo)", con);
        con.Open();
        cmd.Parameters.AddWithValue("@member_id",
textBox1.Text);
        cmd.Parameters.AddWithValue("@last_name",
textBox2.Text);
        cmd.Parameters.AddWithValue("@first_name",
textBox3.Text);
        cmd.Parameters.AddWithValue("@age",
textBox4.Text);
        cmd.Parameters.AddWithValue("@rank",
textBox5.Text);
        cmd.Parameters.AddWithValue("@rank_value",
textBox6.Text);
        cmd.Parameters.AddWithValue("@sensei",
textBox7.Text);
        cmd.Parameters.AddWithValue("@dojo",
textBox8.Text);
        cmd.ExecuteNonQuery();
    }
}

```

```

        con.Close();
        MessageBox.Show("Record Inserted Successfully");
        DisplayData();
        ClearData();
    }
    else
    {
        MessageBox.Show("Please Provide Details!");
    }
}

private void button4_Click(object sender, EventArgs e)
{
    if (textBox1.Text != "" && textBox2.Text != "" &&
        textBox3.Text != "" && textBox4.Text != "" && textBox5.Text !=
        "" && textBox6.Text != "" && textBox7.Text != "" &&
        textBox8.Text != "")
    {
        cmd = new SqlCommand("update Members set
        member_id=@member_id,last_name=@last_name,first_name=@first_name
        ,age=@age,rank=@rank,rank_value=@rank_value,sensei=@sensei,dojo=
        @dojo where member_id=@member_id", con);
        con.Open();
        cmd.Parameters.AddWithValue("@member_id",
        textBox1.Text);
        cmd.Parameters.AddWithValue("@last_name",
        textBox2.Text);
        cmd.Parameters.AddWithValue("@first_name",
        textBox3.Text);
        cmd.Parameters.AddWithValue("@age",
        textBox4.Text);
        cmd.Parameters.AddWithValue("@rank",
        textBox5.Text);
        cmd.Parameters.AddWithValue("@rank_value",
        textBox6.Text);

```

```

        cmd.Parameters.AddWithValue("@sensei",
textBox7.Text);

        cmd.Parameters.AddWithValue("@dojo",
textBox8.Text);

        cmd.ExecuteNonQuery();
        MessageBox.Show("Record Updated Successfully");
        con.Close();
        DisplayData();
        ClearData();
    }
    else
    {
        MessageBox.Show("Please Select Record to
Update");
    }
}

private void button5_Click(object sender, EventArgs e)
{
    if (textBox1.Text != null)
    {
        cmd = new SqlCommand("delete Members where
member_id=@member_id", con);
        con.Open();
        cmd.Parameters.AddWithValue("@member_id",
textBox1.Text);

        cmd.ExecuteNonQuery();
        con.Close();
        MessageBox.Show("Record Deleted Successfully!");
        DisplayData();
        ClearData();
    }
    else
    {

```

```

        MessageBox.Show("Please Select Record to
Delete");
    }
}

private void DisplayData()
{
    con.Open();
    DataTable dt = new DataTable();
    adapt = new SqlDataAdapter("select * from Members",
con);
    adapt.Fill(dt);
    dataGridView1.DataSource = dt;
    con.Close();
}

private void ClearData()
{
    textBox1.Text = "";
    textBox2.Text = "";
    textBox3.Text = "";
    textBox4.Text = "";
    textBox5.Text = "";
    textBox6.Text = "";
    textBox7.Text = "";
    textBox8.Text = "";
}

public Form7()
{
    InitializeComponent();
    dataGridView1.RowHeaderMouseClick += new
DataGridViewCellEventHandler(OnRowHeaderMouseClick);
}

```

```

private void OnRowHeaderMouseClick(object sender,
DataGridViewCellMouseEventArgs e)
{
    textBox1.Text =
dataGridView1.Rows[e.RowIndex].Cells[0].Value.ToString();
    textBox2.Text =
dataGridView1.Rows[e.RowIndex].Cells[1].Value.ToString();
    textBox3.Text =
dataGridView1.Rows[e.RowIndex].Cells[2].Value.ToString();
    textBox4.Text =
dataGridView1.Rows[e.RowIndex].Cells[3].Value.ToString();
    textBox5.Text =
dataGridView1.Rows[e.RowIndex].Cells[4].Value.ToString();
    textBox6.Text =
dataGridView1.Rows[e.RowIndex].Cells[5].Value.ToString();
    textBox7.Text =
dataGridView1.Rows[e.RowIndex].Cells[6].Value.ToString();
    textBox8.Text =
dataGridView1.Rows[e.RowIndex].Cells[7].Value.ToString();
}

```