



**ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ ГОРОДА МОСКВЫ**  
**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ**  
**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ**  
**«КОЛЛЕДЖ СВЯЗИ №54»**  
**ИМЕНИ П.М. ВОСТРУХИНА**

полное название образовательного учреждения

**ДОПУСКАЮ К ЗАЩИТЕ**

Заместитель директора по УПР *О.В.Корешков*

(дата)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

Разработка информационной системы складского учета и контроля рабочего  
времени сотрудников

Выпускная квалификационная работа должна быть выполнена в виде:  
дипломной работы и демонстрационного экзамена

студентом группы ЗИСП11-1

Полиной Андреевной Смирновой

(И.О.Фамилия)

(подпись, дата)

Основная профессиональная образовательная программа по специальности  
09.02.07 Информационные системы и программирование

Форма обучения очная

Руководитель преподаватель

Кристина Александровна Бастрыкина

Руководитель мастер производственного обучения

Кристина Александровна Бастрыкина

(ученая степень, должность, И.О.Фамилия)

(подпись, дата)

Председатель предметной (междисциплинарной, модульной) комиссии  
Виктор Александрович Плаксо

(И.О.Фамилия)

(подпись, дата)

Москва  
2020



ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ ГОРОДА МОСКВЫ  
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ  
«КОЛЛЕДЖ СВЯЗИ №54»  
ИМЕНИ П.М. ВОСТРУХИНА

полное название образовательного учреждения

УТВЕРЖДАЮ

Заместитель директора по УПР *О.В. Корешков*

(дата)

**ЗАДАНИЕ НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ**

Студенту

*Смирновой Полине Андреевне*

(фамилия, имя, отчество полностью)

I. Тема выпускной

*Разработка информационной системы складского*

квалификационной работы

*учета и контроля рабочего времени сотрудников*

II. Срок сдачи студентом законченной работы

*9 апреля 2020 г.*

III. Исходные данные

*Microsoft SQL Server*

*Visual Studio 2016*

**IV. Перечень подлежащих разработке вопросов**

1. *Анализ и классификация информационных систем*

2. *Обоснование проблемы*

3. *Описание объекта*

4. *Разработка информационной системы складского учета и контроля рабочего времени сотрудников (практическая часть)*

5.

**V. Перечень графического / иллюстрационного материала**

1. *Скриншоты экрана (практическая часть)*

2. *Презентация*

3.

4.

5.

6.

**VI. Дата выдачи задания «9» апреля 2020 г.**

Руководитель

Бастрыкина К.А

(подпись)

Задание принял к исполнению

Смирнова П.А

(подпись)

«9» апреля 2020 г.



**ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ ГОРОДА МОСКВЫ**  
**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ**  
**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ**  
**«КОЛЛЕДЖ СВЯЗИ №54»**  
**ИМЕНИ П.М. ВОСТРУХИНА**

полное название образовательного учреждения

**КАЛЕНДАРНЫЙ ПЛАН ВКР**

(с указанием сроков выполнения отдельных этапов)

для специальности 09.02.07 Информационные системы и программирование

(группа 3-ИСП11-1, 3-ИСП11-2, 3-ИСП11-3, 3-ИСП11-4вб)

<b>Сроки преддипломной практики</b>		4 недели с 20.04.2020 по 17.05.2020
	Выбор темы, руководителя, оформление заявления	С 03.02.2020 по 07.02.2020
	Утверждение темы ВКР	10.02.2020
	Выполнение задания по теме ВКР	с 18.05.2020 по 14.06.2020
	Предоставление отчета по практике руководителю	с 11.05.2020 по 15.05.2020
	Аттестация по практике	16.05.2020
<b>Подготовка ВКР</b>		4 недели с 18.05.2020 по 14.06.2020
	Утверждение задания на ВКР	1 неделя
	Подбор и анализ исходной информации	с 18.05.2020 по 24.05.2020
	Подготовка и утверждение плана (оглавления) ВКР	
	Работа над разделами (главами) и устранение замечаний руководителя ВКР	1 неделя с 25.05.2020 по 31.05.2020
	Согласование содержания ВКР, устранение замечаний	1 неделя с 01.06.2020 по 07.06.2020
	<b>Оформление и представление руководителю полного текста работы. Получение отзыва руководителя ВКР.</b>	<b>1 неделя 08.06.2020</b>
	Предоставление студентом готовой ВКР рецензенту	<b>09.06.2020</b>
	Сдача ДЭ	По отдельному графику
	Предзащита ВКР	<b>10.06-13.06.2020</b>

Руководитель \_\_\_\_\_ (подпись)

Руководитель \_\_\_\_\_ (подпись)

План принял к исполнению «\_\_» \_\_\_\_\_ 2020 г.

\_\_\_\_\_ (подпись студента)

## ОГЛАВЛЕНИЕ

Введение.....	5
Глава I. Анализ процесса учета рабочего времени.....	7
1.1. Изучение вопроса учета времени работников, и определение назначения системы .....	7
1.2. Выбор средств разработки .....	10
1.3. Основные этапы разработки информационной системы.....	14
Выводы.....	15
Глава II. Проектирование и разработка информационной системы учета и контроля рабочего времени сотрудников.....	17
2.1. Определение функций и требований программы .....	17
2.2. Проектирование и разработка базы данных для информационной системы учета и контроля рабочего времени.....	17
2.3. Проектирование и разработка информационной системы.....	19
Выводы.....	28
Заключение.....	29
Список литературы.....	30

## ВВЕДЕНИЕ

Одним из важнейших условий выполнения плана производства, увеличения выработки продукции на каждого члена трудового коллектива, а также рационального использования трудовых ресурсов является экономное и эффективное использование рабочего времени. От того, насколько полно и рационально используется рабочее время, зависят эффективность работы, выполнение всех технико-экономических показателей. Поэтому анализ использования рабочего времени является важной составной частью аналитической работы на промышленном предприятии.

Учет рабочего времени сотрудников обязателен согласно действующему законодательству РФ, а потому организовать его в самой простой форме необходимо каждому работодателю. Но что, если не ограничиваться табелем для бухгалтерии, а использовать современные средства, анализируя не только количество, но и качество. Получить полную картину загрузки каждого работника и мотивировать его к более сознательному и дисциплинированному труду важно для любого бизнеса. Разнообразные программы и системы наблюдения позволяют вести учет времени работы сотрудников так, как удобно работодателю.

Актуальность данной работы заключается в необходимости учёта и контроля рабочего времени сотрудников любой организации, способной оказывать качественную поддержку для успешного ведения управленческой деятельности в частности и коммерческой деятельности в целом.

Объектом дипломной работы являются управленческие информационные процессы организаций, связанные с учётом и контролем рабочего времени сотрудников. Предметом – информационная система учёта и контроля рабочего времени сотрудников.

Целью работы является повышение эффективности управленческой деятельности предприятий за счет внедрения разработанной информационной

системы учёта и контроля рабочего времени сотрудников. Для достижения поставленной цели необходимо выполнить следующие задачи:

- проанализировать процесс учета рабочего времени;
- выбрать программные средства разработки и рассмотреть их особенности;
- определить требования к информационной системе;
- спроектировать и реализовать базу данных;
- разработать интерфейс для созданной базы данных.

Выпускная квалификационная работа состоит из введения, теоретической части, практической части, заключения и списка использованной литературы.

					09.02.07 – ЗИСП11-1	Лист
						6
Изм.	Лист	№ докум.	Подпись	Дата		

## ГЛАВА 1. АНАЛИЗ ПРОЦЕССА УЧЕТА РАБОЧЕГО ВРЕМЕНИ

### 1.1. Изучение вопроса учета времени работников, и определение назначения системы

Одним из важнейших условий выполнения плана производства, увеличения выработки продукции на каждого члена трудового коллектива, а также рационального использования трудовых ресурсов является экономное и эффективное использование рабочего времени. От того, насколько полно и рационально используется рабочее время, зависят эффективность работы, выполнение всех технико-экономических показателей. Поэтому анализ использования рабочего времени является важной составной частью аналитической работы на промышленном предприятии.

С помощью анализа изучаются тенденции развития, глубоко и системно исследуются факторы изменения результатов деятельности, обосновываются планы и управленческие решения, осуществляется контроль за их выполнением, выявляются резервные повышения эффективности производства, оцениваются результаты деятельности предприятия, вырабатывается стратегия его развития и т.д.

Эффективное управление трудовыми ресурсами невозможно без адекватной информации. Поэтому в процессе деятельности предприятия регулярно собираются данные, характеризующие различные аспекты состояния трудовых ресурсов, и проводится их детальный анализ. В нем рассматривается информация о различных сторонах управления трудовыми ресурсами предприятия - производительности, издержках на рабочую силу, профессиональном обучении, динамике рабочей силы и т.д.

От обеспеченности предприятия трудовыми ресурсами и эффективности их использования зависят объём и своевременность выполнения всех работ, эффективность использования оборудования, машин механизмов и как

					09.02.07 – ЗИСП11-1	Лист
Изм.	Лист	№ докум.	Подпись	Дата		7

результат - объём производства продукции, её себестоимость, прибыль и ряд других экономических показателей.

Учет рабочего времени — это методика подсчета часов, которые сотрудник фактически посвятил своим рабочим обязанностям. Сведения вносятся в табель, который законодательство обязывает вести каждое предприятие независимо от его размера.

Все графики работы, утверждаемые разнообразными работодателями, не должны противоречить требованиям Трудового кодекса РФ: как по вопросам выбора режима рабочего времени, так и в контексте организации его наглядного учета. Учет рабочего времени – это непреложная обязанность каждой организации и естественная часть ее работы, позволяющая формировать корректные платежные ведомости, адекватно оценивать производительность и КПД работы сотрудников, а также своевременно узнавать о случаях, когда они не соблюдают режим рабочего дня.

В соответствии с действующими нормативными актами нашей страны, под рабочим временем понимается то время, на протяжении которого работник занимается осуществлением своих трудовых обязанностей в соответствии с положениями, очерченными в трудовом договоре, и инструкциями внутреннего распорядка организации.

Теоретически сотрудник может по каким-то причинам прозаниматься время, не совпадающее с условиями подписанного трудового договора или постановлениями внутреннего распорядка. Если речь в этом случае будет идти о работе сверх нормы, то она все равно будет считаться официально потраченным на труд временем.

При этом, если сверхурочная работа окажется произведенной в нарушение законодательства по требованию нанимателя, то последний может быть привлечен к правовой ответственности.

Продолжительность рабочего времени также predetermined ТК России и, по состоянию на данный момент, равняется 40 часам в неделю.



Каждый работодатель в числе своих обязанностей имеет учет времени, отработанного сотрудником, а также должен прописывать режим рабочего времени и временные возможности для отдыха в строчках трудового договора. Этот режим может не конкретизироваться в составе договора только в том случае, если для сотрудника он будет полностью отвечать коллективным правилам, которые применяются в организации.

Режим рабочего времени – это комплексное, составное понятие, которое включает в себя сразу несколько аспектов упорядочения трудовой деятельности работника:

- продолжительность рабочей недели. Вариантом может быть несколько: неполный формат, пять рабочих дней и два свободных, шесть трудовых будней и один выходной, а также распределение дней отдыха по скользящему графику;
- продолжительность ежедневной смены (включая вариант для рабочего дня неполного формата);
- время, когда сотрудник должен начать производство своих трудовых обязанностей, и когда он должен это закончить;
- время начала и завершения перерывов в течение трудового дня;
- количество смен, в рамках которых работают сотрудники организации, в сутки;
- график чередования рабочих и освобожденных от работы дней;
- категории работников, для которых учреждается ненормированный рабочий день.

Все эти параметры указываются в составе правил внутреннего трудового распорядка. Но, как уже было отмечено выше, если для какого-то сотрудника режим рабочего времени разнится с корпоративными правилами, которые действуют в организации, то все это должно быть указано в его трудовом договоре.

Также, для учета рабочего времени в нашей стране установлен нормативный документ «Табель учета рабочего времени», который представлен на рисунке 1.1.

Унифицированная форма № Т-12  
Утверждена Постановлением Госкомстата России от 05.01.2004 № 1

Форма по ОКУД 0301007  
по ОКПО 00270015

Код

ООО «Мир дерева»  
наименование организации  
Отдел менеджмента и маркетинга  
наименование структурного подразделения

Номер документа 8  
Дата составления 30.06.2016

Отчетный период  
с 01.06.2016 по 30.06.2016

ТАБЕЛЬ  
учета рабочего времени  
и расчета оплаты труда  
УСЛОВНЫЕ ОБОЗНАЧЕНИЯ

	Код	
	буквенный	цифровой
Продолжительность работы в дневное время	Я	01
Продолжительность работы в ночное время	Н	02
Продолжительность работы в выходные и нерабочие праздничные дни	РВ	03
Продолжительность сверхурочной работы	С	04
Продолжительность работы вахтовым методом	ВМ	05
Служебная командировка	К	06
Повышение квалификации с отрывом от работы	ПК	07
Повышение квалификации с отрывом от работы в другой местности	ПМ	08
Ежегодный основной оплачиваемый отпуск	ОТ	09
Ежегодный дополнительный оплачиваемый отпуск	ОД	10
Дополнительный отпуск в связи с обучением с сохранением среднего заработка работникам, совмещающим работу с обучением	У	11
Сокращенная продолжительность рабочего времени для обучающихся без отрыва от производства с частичным сохранением заработной платы	УВ	12
Дополнительный отпуск в связи с обучением без сохранения заработной платы	УД	13
Отпуск по беременности и родам (отпуск в связи с усыновлением новорожденного ребенка)	Р	14
Отпуск по уходу за ребенком до достижения им возраста трех лет	ОЖ	15
Отпуск без сохранения заработной платы, предоставляемый работнику по разрешению работодателя	ДО	16
Отпуск без сохранения заработной платы при условиях, предусмотренных действующим законодательством Российской Федерации	ОЗ	17
Ежегодный дополнительный отпуск без сохранения заработной платы	ДВ	18

	Код	
	буквенный	цифровой
Временная нетрудоспособность (кроме случаев, предусмотренных кодом «Т») с назначением пособия согласно законодательству	Б	19
Временная нетрудоспособность без назначения пособия в случаях, предусмотренных законодательством	Т	20
Сокращенная продолжительность рабочего времени против нормальной продолжительности рабочего дня в случаях, предусмотренных законодательством	ЛЧ	21
Время вынужденного прогула в случае признания увольнения, перевода на другую работу или отстранения от работы незаконными с восстановлением на прежней работе	ПВ	22
Невыходы на время исполнения государственных или общественных обязанностей согласно законодательству	Г	23
Прогулы (отсутствие на рабочем месте без уважительных причин в течение времени, установленного законодательством)	ПР	24
Продолжительность работы в режиме неполного рабочего времени по инициативе работодателя в случаях, предусмотренных законодательством	НС	25
Выходные дни (еженедельный отпуск) и нерабочие праздничные дни	В	26
Дополнительные выходные дни (оплачиваемые)	ОВ	27
Дополнительные выходные дни (без сохранения заработной платы)	НВ	28
Забастовка (при условиях и в порядке, предусмотренных законом)	ЗВ	29
Неявка по невыясненным причинам (до выяснения обстоятельств)	НН	30
Время простоя по вине работодателя	РП	31
Время простоя по причинам, не зависящим от работодателя и работника	НП	32
Время простоя по вине работника	ВП	33
Отстранение от работы (недопущение к работе) с оплатой (пособием) в соответствии с законодательством	НО	34
Отстранение от работы (недопущение к работе) по причинам, предусмотренным законодательством, без начисления заработной платы	НБ	35
Время приостановки работы в случае задержки выплаты заработной платы	НЗ	36

Рисунок 1.1. Табель учета рабочего времени, форма Т-12

Важно также понимать, что эти параметры хоть и устанавливаются работодателем, однако должны соответствовать детерминированным требованиям, предусмотренным 110-ой статьей ТК РФ. В частности, вне зависимости от длительности рабочей недели каждый служащий должен отдыхать на протяжении не менее чем 42 часов подряд. Исключение из этого правила составляет только вахтовый метод, и то он предполагает наличие хотя бы одного выходного дня в неделю.

Таким образом можно выявить основное назначение разрабатываемой системы: учет рабочего времени сотрудников, с датой и временем начала работы и окончания, где сотрудники сами будут указывать время работы, а администраторы смогут получить доступ к этой информации.

## 1.2. Выбор средств разработки

Важнейшим элементом в процессе разработки приложения является выбор правильной IDE, зависящий не только от платформы, но и уровня собственной подготовки.

IDE – система программных средств, используемая программистами для разработки программного обеспечения.

Существует огромное множество платформ для разработки приложений. Каждая из платформ имеет интегрированную среду разработки, предоставляющую инструменты, позволяющие разработчику программировать, тестировать и внедрять приложения на целевую платформу.

Среда разработки включает в себя:

- текстовый редактор;
- транслятор (компилятор и/или интерпретатор);
- средства автоматизации сборки;
- отладчик.

Чтобы средства разработки и технологии отвечали требованиям разработчиков, в корпорации Майкрософт была создана совершенно новая модель программирования для доступа к данным, основанная на .NET Framework. Построение на основе .NET Framework гарантирует единообразие доступа к данным: компоненты используют систему общих типов, общие шаблоны разработки и соглашения о пространствах имен.

В .NET Framework поддерживается прямая и обратная совместимость. В контексте .NET Framework обратная совместимость означает, что любое приложение, созданное в .NET Framework более ранней версии, будет выполняться и в более поздней версии. Прямая совместимость означает возможность выполнения приложения, созданного в более поздней версии .NET Framework SDK v 2.0, в .NET Framework более ранней версии.

Классы ADO.NET были разработаны для поддержки возможностей новой модели программирования: интеграции с XML, единого представления

данных с возможностью комбинирования данных из различных источников, а также средств оптимизации взаимодействия с базой данных, представленных в .NET Framework.

Структура ADO.NET создана для решения задач современной модели разработки приложений. В то же время модель программирования по возможности приближена к ADO, что упрощает переход разработчиков ADO к новой среде. ADO.NET является неотъемлемой частью .NET Framework, оставаясь понятной программистам ADO.

Microsoft Visual Studio – полнофункциональная интегрированная среда разработки с поддержкой популярных языков программирования, среди которых C, C++, VB.NET, C#, F#, JavaScript, Python.

Специально для этой платформы Microsoft разработала язык программирования C#.

Язык программирования C# – популярный объектно-ориентированный язык программирования, который с выходом .Net Core в 2016 году, стал кроссплатформенным. Он обеспечивает создание безопасного и верифицируемого кода, применение которого возможно в любом современном программном комплексе. Созданные на языке C# приложения могут успешно работать на любом компьютере под управлением операционной системы Windows, Linux и MacOS независимо от архитектуры устройств.

C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию, поддерживает полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML.

Также, среда разработки Visual Studio занимает первое место среди бесплатных IDE для программирования на C# в этом году (рисунок 1.2).









BEST C# IDES		PRICE	PLATFORMS	BRACKET MATCHING	INTEGRATED DEBUGGER
93		JetBrains Rider	€139/FIRST YEAR	-	-
91		Visual Studio	FREE / PAID	Windows Mac	yes
90		Visual Studio Code	FREE	Windows, macOS, Linux	Yes
89		MonoDevelop	FREE	Windows Linux Mac	-
--		browxy	FREE	-	yes
--		SharpDevelop	OPEN SOURCE	-	-
--		scriptcs + any text editor	-	-	-
--		Unity	-	-	-

Рисунок 1.2. Самые популярные IDE для разработки на C#

По этим причинам было решено использовать среду разработки Visual Studio, а также язык программирования C#, написанный специально для этой среды разработки.

Далее необходимо выбрать СУБД разработки – систему управления базами данных.

Современные СУБД в основном являются приложениями Windows, так как данная среда позволяет более полно использовать возможности персональной ЭВМ, нежели среда DOS.

Среди наиболее ярких представителей систем управления базами данных можно отметить PostgreSQL, Microsoft SQL Server и Oracle, используемые в приложениях, построенных по технологии «клиент-сервер».

Фактически, у любой современной СУБД существует аналог, выпускаемый другой компанией, имеющий аналогичную область применения и возможности, любое приложение способно работать со многими форматами представления данных, осуществлять экспорт и импорт данных благодаря наличию большого числа конвертеров. Общепринятыми, также, являются технологии, позволяющие использовать возможности других приложений, например, текстовых процессоров, пакетов построения графиков и т.п., и

встроенные версии языков высокого уровня (чаще - диалекты SQL и / или VBA) и средства визуального программирования интерфейсов разрабатываемых приложений.

При выборе СУБД основными факторами являются её гибкость, совместимость с выбранной средой разработки и языком программирования. Поскольку MS SQL Server и C# разработаны одной компанией и имеют высокую совместимость и простоту совместного использования, а также необходимую гибкость – выбор пал на СУБД Microsoft SQL Server.

### 1.3. Основные этапы разработки информационной системы

Для начала разработки информационной системы необходимо рассмотреть основные этапы её создания.

Спецификация (определение требований к программе): на данном этапе происходит подробное описание исходных данных, осуществляется формулировка требований к получаемому результату, рассматриваются всевозможные поведения программы при возникновении особых случаев (к примеру, если ввели неверные данные), происходит разработка диалоговых окон, которые обеспечат взаимодействие пользователя и самой программы.

Разработка алгоритма: на этом этапе программист определяет последовательность необходимых действий, которые впоследствии нужно выполнить для получения желаемого результата.

Если возникает ситуация, когда поставленную задачу можно решить несколькими способами, то само собой, возможны множество разных вариантов алгоритма решения. Тогда разработчик программы по некоторому немаловажному критерию (к примеру, скорость решения алгоритма) делает выбор более подходящего решения.

Результат данного этапа разработки программы – подробное словесное описание алгоритма программы, либо блок-схема алгоритма.

Кодирование: после проведения спецификации и составления алгоритма решения, используемый алгоритм в итоге будет записан на необходимом языке программирования. Результатом этапа кодирования является готовая программа.

Отладка: На данном этапе программист занимается отладкой программы, то есть поиском и устранением ошибок. Последние делятся на две группы: алгоритмические и синтаксические (ошибки в тексте исходной программы). Из этих двух групп ошибок наиболее легко устранить синтаксические ошибки, тогда как алгоритмические ошибки определить достаточно трудно.

Этап отладки считается законченным лишь тогда, когда исходная программа работает корректно и правильно при одном или двух наборах первичных данных.

Тестирование программы очень важно, поскольку в большинстве случаев программисты создают программы не для личного применения, а чтоб их программой пользовались другие. На этапе тестирования разработчик проверяет поведение программы при большом числе наборов входных данных, как верных, так и специально подобранных неверных.

## Выводы

Одним из важнейших факторов успешной работы любой организации является экономное и эффективное использование рабочего времени. Рабочее время – это время, которое работник тратит на выполнение своих обязанностей согласно должностной инструкции. Оплата труда рассчитывается с учетом отработанных часов.

Процесс подсчета рабочих часов в нашем законодательстве предусматривает специальные нормативные документы. Разрабатываемое приложение предназначено для автоматизации этого процесса.

Для разработки было решено использовать язык программирования С#, написанный специально для среды разработки Visual Studio. На текущий момент данная IDE является наиболее используемой для этого языка программирования.

					09.02.07 – ЗИСП11-1	Лист
						16
Изм.	Лист	№ докум.	Подпись	Дата		



## ГЛАВА II. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ УЧЕТА И КОНТРОЛЯ РАБОЧЕГО ВРЕМЕНИ СОТРУДНИКОВ

В I главе ВКР упоминалось, что учет и контроль рабочего времени сотрудников является неотъемлемой и важной частью работы всего предприятия в целом.

Поэтому, целью второй главы выпускной квалификационной работы является создание приложения для учета и контроля рабочего времени сотрудников.

### 2.1. Определение функций и требований программы

Программа должна иметь следующие основные возможности:

- 1) авторизация пользователей;
- 2) распределение прав пользователей в зависимости от роли (работник, менеджер, администратор);
- 3) создание, хранение, вывод и изменение данных о всех сущностях системы.

Для работы в системе пользователь должен иметь компьютер со следующими техническими характеристиками:

- процессор с тактовой частотой 2,4 ГГц, с двумя и более ядер;
- оперативная память 2 Гб;
- не менее 5 Гб свободного места на жестком диске;
- наличие Ethernet.

Далее, разработаем базу данных для информационной системы.

### 2.2. Проектирование и разработка базы данных

Выделим основные сущности информационной системы:

- пользователи системы;

- роли пользователей;
- рабочее время;

Приведенные сущности должны быть отражены в базе данных, а также она должна соответствовать третьей нормальной форме.

У каждого пользователя системы должны быть имя, фамилия, отчество, пароль, фото и роль. Создадим таблицу user с соответствующими полями, как показано на рисунке 2.1. Также, исходя из требований первой нормальной формы – необходимо создать суррогатный ключ – id для каждой сущности.

Имя столбца	Тип данных	Разрешить ...
id	bigint	<input type="checkbox"/>
firstName	nvarchar(30)	<input type="checkbox"/>
lastName	nvarchar(30)	<input type="checkbox"/>
email	nvarchar(30)	<input type="checkbox"/>
password	nvarchar(30)	<input type="checkbox"/>
role	bigint	<input type="checkbox"/>
avatar	varbinary(MAX)	<input checked="" type="checkbox"/>

Рисунок 2.1. Сущность user

Далее, таблица ролей. У каждой роли есть id и название. Всего существует три роли пользователей – сотрудники, менеджеры, администраторы. Создадим и сразу заполним данную таблицу (рисунок 2.2).

id	name
1	employee
2	manager
3	admin

Рисунок 2.2. Сущность role

Сущность рабочего времени (рисунок 2.3) содержит данные о начале работы, окончании, статусе и пользователе, который выполняет работу. Статус может быть «На работе» и «Завершен».

Имя столбца	Тип данных	Разрешить ...
id	bigint	<input type="checkbox"/>
startWork	datetime	<input type="checkbox"/>
endwork	datetime	<input checked="" type="checkbox"/>
[user]	bigint	<input type="checkbox"/>
status	bigint	<input type="checkbox"/>

Рисунок 2.3. Сущность accounting

Законченная схема базы данных представлена на рисунке 2.4.

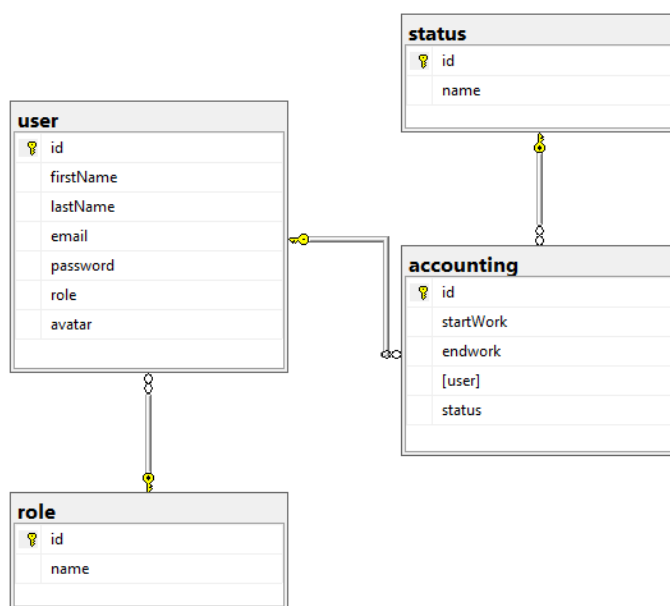


Рисунок 2.4. Диаграмма базы данных

Далее, займемся разработкой самого приложения на языке программирования С# в среде разработки Visual Studio.

## 2.3 Проектирование и разработка информационной системы

В первую очередь необходимо предоставить пользователю возможность авторизоваться в системе. Приложение производит поиск в базе данных и, если находит пользователя, в зависимости от его роли, открывает главное окно приложения. Также, для всего приложения необходимо выбрать единый стиль, логотип (рисунок 2.5.) и набор цветов.



Рисунок 2.5. Логотип программы

Для фона, а также цвета кнопок, используем цвет «Pure blue», для заголовка «Strong blue», поля ввода и текст – «White».

Архитектура приложения будет страничной. Разметим страницу авторизации, как показано на рисунке 2.6.

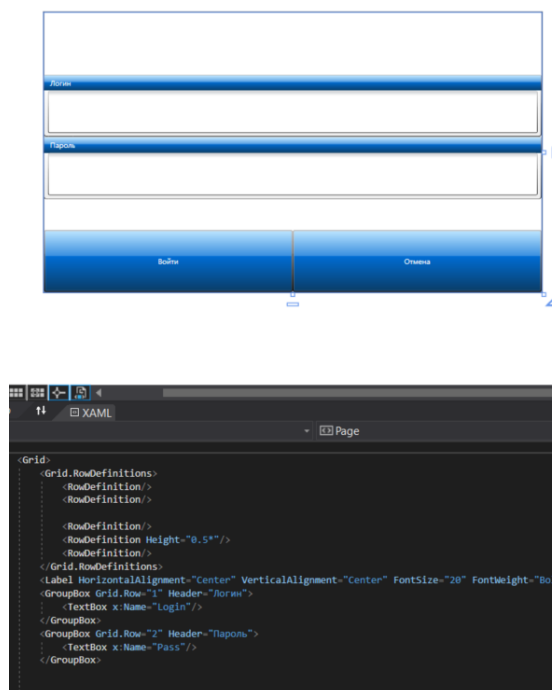


Рисунок 2.6. Разметка страницы авторизации

Подключим базу данных, используя Entity Framework. В рамках приложения сформируется модель базы данных и сразу откроется диаграмма базы данных, как показано на рисунке 2.7.

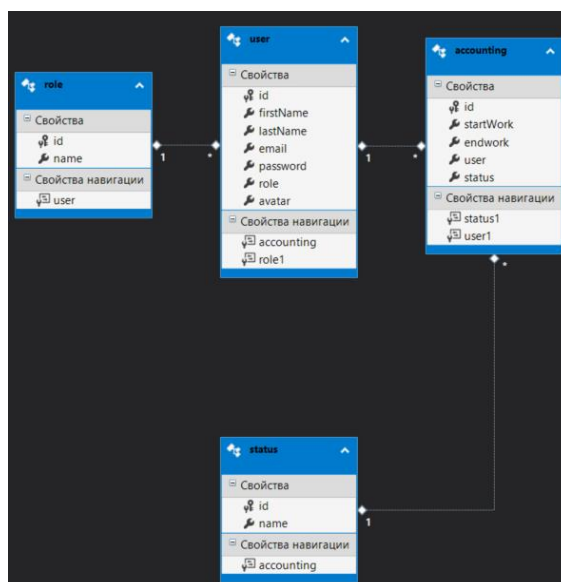


Рисунок 2.7. Модель базы данных

При авторизации будем проверять, что пользователь существует в бд и, в зависимости от его роли, переходить на соответствующую главную страницу. Код представлен на рисунке 2.8.

```

public partial class Authorization : Page
{
    TimeManagetEntities context = new TimeManagetEntities();

    public static user User { get; set; }

    public Authorization()
    {
        InitializeComponent();
    }

    private void Exit(object sender, RoutedEventArgs e)
    {
        Application.Current.Shutdown();
    }

    private void LogInSystem(object sender, RoutedEventArgs e)
    {
        var user = context.user.Where(i => i.email == Login.Text &&
            i.password == Pass.Text);
        if (user.Count() > 0)
        {
            User = user.FirstOrDefault();
            switch (user.FirstOrDefault().role)
            {
                case 1:
                    NavigationService.Navigate(new EmployeeManagment());
                    break;
                case 2:
                    NavigationService.Navigate(new Managers());
                    break;
                case 3:
                    NavigationService.Navigate(new Admin());
                    break;
                default:
                    break;
            }
        }
        else
        {
            MessageBox.Show("введите верные данные");
        }
    }
}

```

Рисунок 2.8. Код страницы авторизации

Далее, создадим главную страницу для сотрудников (рисунок 2.9.), а также другие страницы.

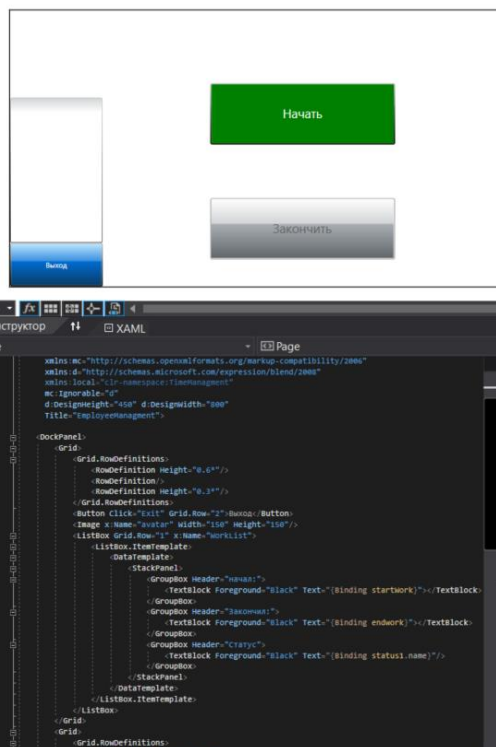


Рисунок 2.9. Разметка главного окна сотрудника

Изм.	Лист	№ докум.	Подпись	Дата

09.02.07 – ЗИСП11-1

Лист

21

При запуске приложения форма авторизации выглядит так, как показано на рисунке 2.10.

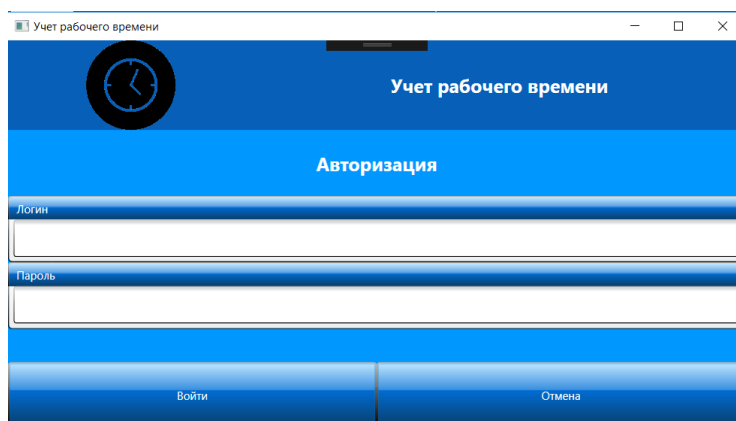


Рисунок 2.10. Страница авторизации

Добавим функционал на главную страницу сотрудника. Слева будет отображаться фотография сотрудника из базы, а также список его рабочих часов с указанием даты и времени начала работы, а также статусом. Ниже расположена кнопка «Выход». По центру разместим две кнопки с надписями «Начать» и «Закончить» (рисунок 2.11.).

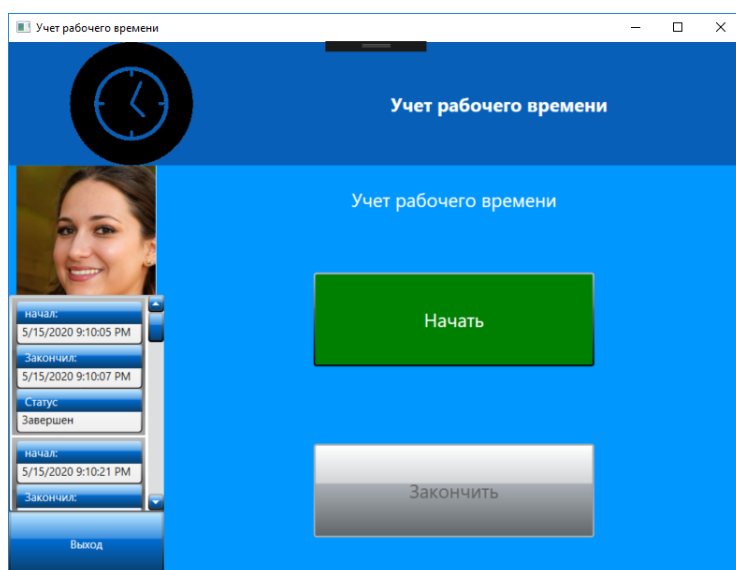


Рисунок 2.11. Страница сотрудника

При нажатии кнопки начать запускается таймер отсчета времени – сколько прошло времени с момента нажатия на кнопку (рисунок 2.12).

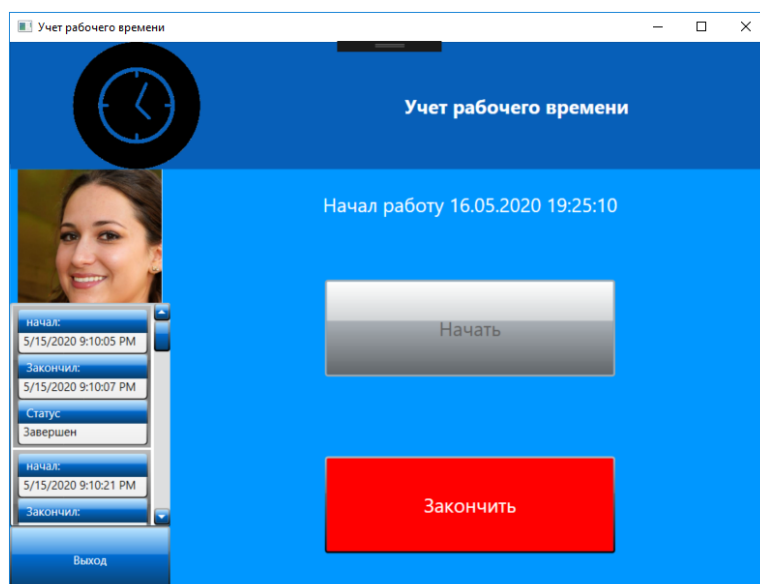


Рисунок 2.12. Страница сотрудника, начавшего работать

Чтобы закончить работу необходимо нажать кнопку «Закончить», при этом на экране появится информация о том, сколько минут проработал сотрудник. Это можно увидеть на рисунке 2.13.

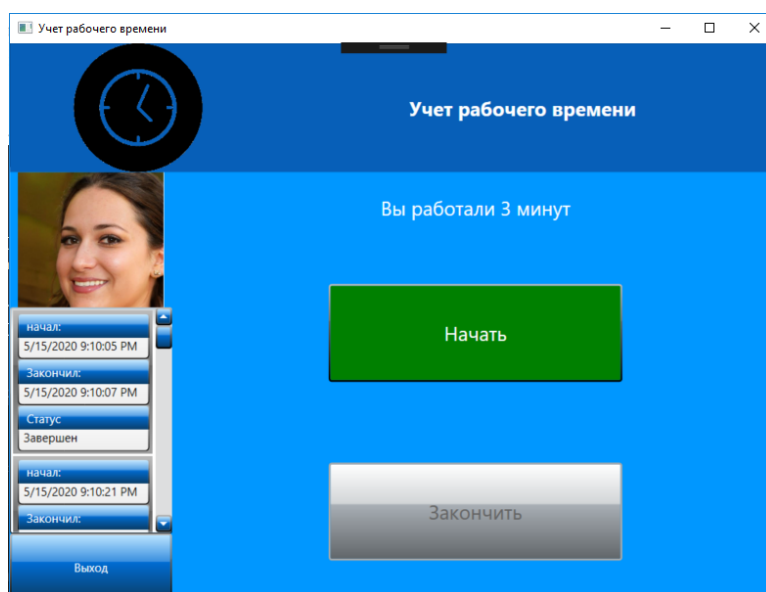


Рисунок 2.13. Страница сотрудника, закончившего работать

При начале работы – данные об этом заносятся в базу данных со статусом «На работе». При нажатии на кнопку «Закончить» статус рабочего времени меняется на «Завершен».

Создадим страницу менеджера. В левой части окна будет отображаться фото менеджера, кнопка «Выйти» и фильтр на рабочее время «Только в работе». По центру отобразим список всех рабочих часов, с указанием

фамилии сотрудника, даты и времени начала работы, даты и времени окончания работы, а также статусом.

Конечный вид страницы после запуска приложения и авторизацией под менеджером представлен на рисунке 2.14.

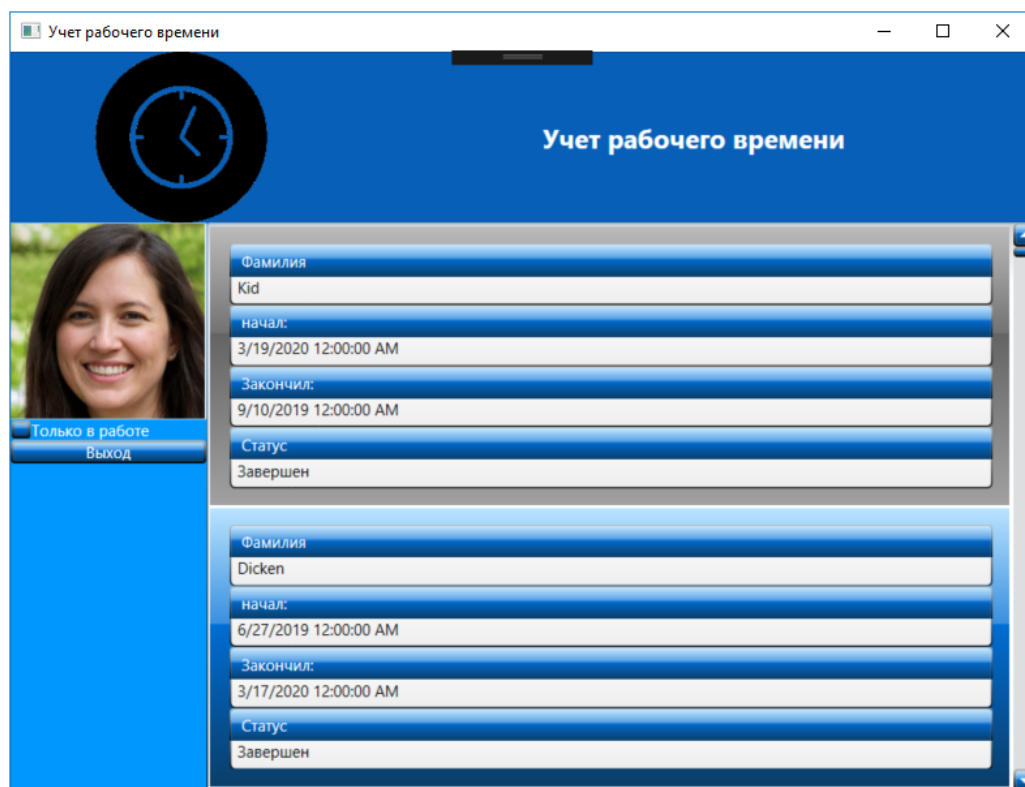


Рисунок 2.14. Страница менеджера

Далее, создадим и разметим страницу администратора. В левой части окна будет располагаться фото администратора, кнопки «Добавить», «Изменить» и «Выход». По середине будет располагаться список всех пользователей системы с указанием фото, фамилии, имени, почты и роли.

Часть кода разметки представлена на рисунке 2.15.



```

<DockPanel>
  <StackPanel>
    <Image VerticalAlignment="Top" Name="avatar" Width="150" ScrollViewer.HorizontalScrollBarVisibility="Hidden"/>
    <Button Click="AddNew">Добавить</Button>
    <Button Click="Update">Изменить</Button>
    <Button Click="Exit">Выход</Button>
  </StackPanel>
  <ListBox x:Name="UserList" >
    <ListBox.ItemsPanel>
      <ItemsPanelTemplate>
        <WrapPanel IsItemsHost="True">
          <WrapPanel.Style>
            <Style TargetType="WrapPanel">
              </Style>
            </WrapPanel.Style>
          </WrapPanel>
        </ItemsPanelTemplate>
      </ListBox.ItemsPanel>
      <ListBox.ItemTemplate >
        <DataTemplate >
          <StackPanel Margin="10">
            <GroupBox Header="Пользователь" >
              <Image Height="150" Source="{Binding avatar}"></Image>
            </GroupBox>
            <GroupBox Header="Фамилия" >
              <TextBlock Foreground="Black" Text="{Binding lastName}"></TextBlock>
            </GroupBox>
            <GroupBox Header="Имя" >
              <TextBlock Foreground="Black" Text="{Binding firstName}"></TextBlock>
            </GroupBox>
            <GroupBox Header="Email" >
              <TextBlock Foreground="Black" Text="{Binding email}"></TextBlock>
            </GroupBox>
            <GroupBox Header="Роль" >
              <TextBlock Foreground="Black" Text="{Binding role1.name}"></TextBlock>
            </GroupBox>
          </StackPanel>
        </DataTemplate>
      </ListBox.ItemTemplate>
    </ListBox>
  </DockPanel>

```

Рисунок 2.15. Код разметки страницы администратора

При нажатии на кнопку «Изменить» - программа в первую очередь проверит выбран ли пользователь, данные которого необходимо изменить. Для выбора пользователя достаточно нажать на него в списке. Если пользователь был выбран – откроется новое окно для редактирования данных пользователя.

При нажатии на кнопку «Добавить» - откроется новое окно для внесения данных нового пользователя. При этом основное окно администратора блокируется.

Вид страницы администратора после написания данного функционала приведен на рисунке 2.16.

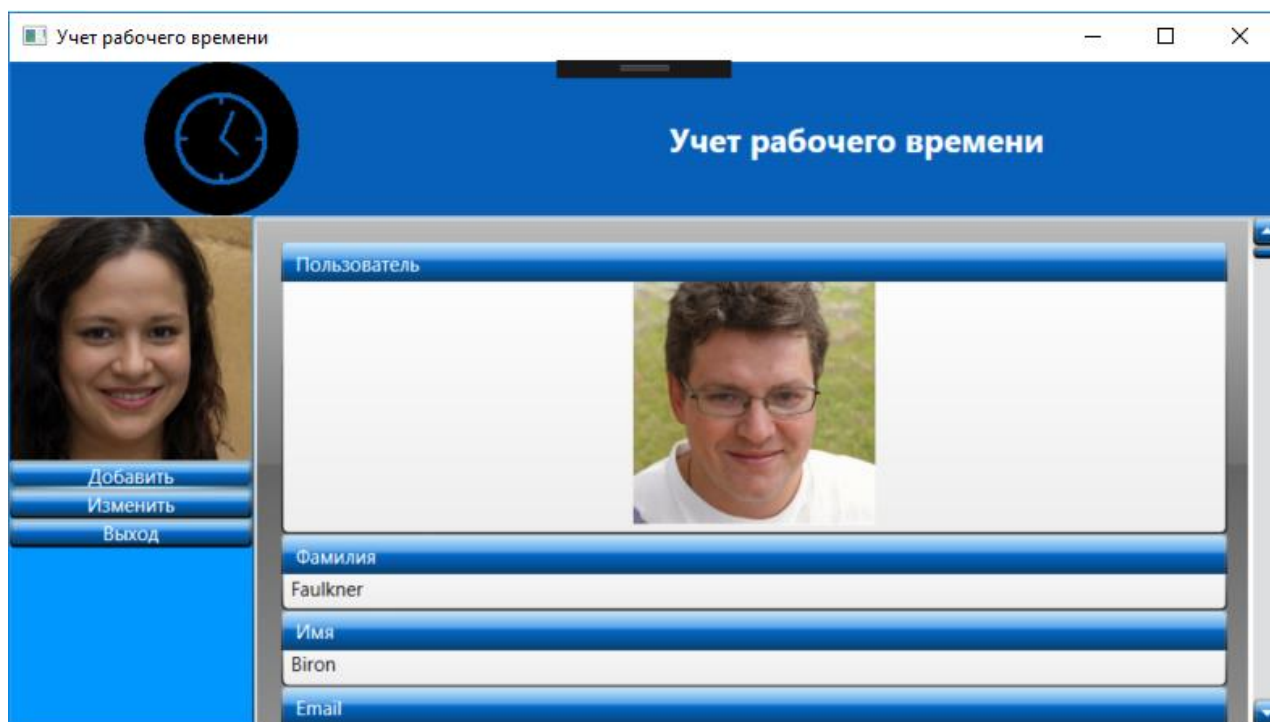


Рисунок 2.16. Страница администратора

Вид окна, открывающегося при нажатии на кнопку «Добавить» представлен на рисунке 2.17.



Рисунок 2.17. Окно добавления нового пользователя

При нажатии на кнопку «Сохранить» программа проверит, что пользователя с такими данными ещё нет в базе и добавит его в список

пользователей. При нажатии на кнопку «Отменить» - приложение закроет окно добавления пользователя и вернет доступ к главному окну.

Чтобы изменить данные существующего пользователя – необходимо кликнуть по пользователю, после чего фон его карточки подцветится синим (рисунок 2.18).

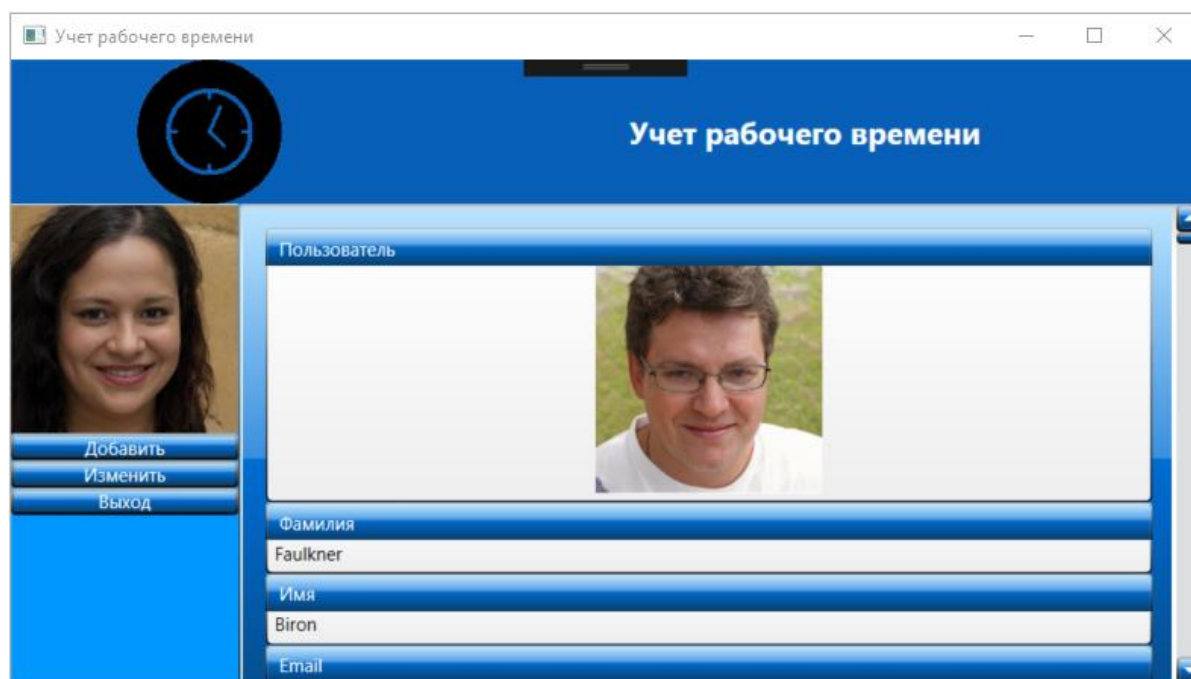


Рисунок 2.18. Изменение фона карточки пользователя

Далее, при нажатии кнопки «Изменить» - откроется окно для редактирования профиля пользователя (рисунок 2.19), где можно изменить любые данные пользователя, нажав на необходимую строку и введя новые данные.



Рисунок 2.19. Окно изменения данных пользователя

Чтобы сохранить данные – достаточно нажать кнопку «Сохранить» (рисунок 2.20.). Программа проверит данные на корректность и изменит их в базе данных.



Рисунок 2.20. Кнопка «Сохранить» в окне изменения данных пользователя

## Выводы

Целью большинства коммерческих организаций является извлечение прибыли и увеличение эффективности. Чтобы повысить эффективность работы предприятия – в первую очередь необходимо вести учет действительного рабочего времени сотрудников предприятия. Благодаря разработанному приложению менеджер, а также сам сотрудник, смогут увидеть все часы, потраченные на работу и из этого рассчитывать заработную плату.

Само приложение состоит из базы данных, приведенной к третьей нормальной форме, а также непосредственно программы, написанной в среде разработки Visual Studio на языке программирования C# и предоставляет различным ролям пользователей различные права в системе.

## ЗАКЛЮЧЕНИЕ

Учет рабочего времени ведется работодателем в таблице учета рабочего времени, в котором ежедневно отражается количество отработанных часов (дней) сотрудниками организации.

Можно выделить несколько причин – почему необходимо ввести автоматизированный учет рабочего времени. Время – это самый ценный ресурс, и, к сожалению, не всегда сотрудники компании распоряжаются им правильно. Согласно данным PERCo, после введения в компаниях автоматизированного учета рабочего времени процент опозданий и прогулов значительно уменьшается. Улучшение трудовой дисциплины положительно влияет на прибыль за счет роста производительности труда.

В решении этих проблем может помочь разработанная в данной выпускной квалификационной работе система автоматизированного учета рабочего времени. Система работает просто: сотрудник заходит в систему и нажимает одну кнопку, чтобы начать работать, и другую кнопку, чтобы закончить. Система запоминает эту информацию и позволяет формировать отчеты на ее основе.

## СПИСОК ЛИТЕРАТУРЫ

- 1) Абрамян, Михаил Visual C# на примерах / Михаил Абрамян. - М.: БХВ-Петербург, 2015. - 572 с.
- 2) Албахари, Джозеф C# 6.0. Справочник. Полное описание языка / Джозеф Албахари, Бен Албахари. - М.: Вильямс, 2018. - 406 с.
- 3) Вагнер, Билл. Эффективное программирование на C# / Билл Вагнер. – М.: ЛОРИ, 2017. – 320 с.
- 4) Ватсон, Б. C# 4.0 на примерах (C# 4.0. How-To) / Б. Ватсон. - М.: БХВ-Петербург, 2017. - 608 с.
- 5) Голдштейн, Саша Оптимизация приложений на платформе .Net / Саша Голдштейн, Дима Зурбалева, Идо Флатов. - М.: ДМК Пресс, 2017. - 524 с
- 6) Гэри, Маклин Холл Адаптивный код на C#. Проектирование классов и интерфейсов, шаблоны и принципы SOLID / Гэри Маклин Холл. - М.: Вильямс, 2015. - 432 с.
- 7) Культин, Н.Б. Microsoft Visual C# в задачах и примерах (+ CD-ROM) / Н.Б. Культин. - М.: БХВ-Петербург, 2017. - 308 с.
- 8) Культин, Никита Борисович Основы программирования в Microsoft Visual C# 2010 (+ CD-ROM) / Культин Никита Борисович. - М.: БХВ-Петербург, 2016. - 201 с.
- 9) Магда, Ю. С. NI Measurement Studio. Практика разработки систем измерения и управления на C# / Ю.С. Магда. - М.: ДМК Пресс, 2017. - 190 с.
- 10) Джеффри Рихтер: CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. 4-е издание/Д. Рихтер. - М.: Питер, 2019. - 436 с.
- 11) Фримен, А. LINQ. Язык интегрированных запросов в C# 2010 для профессионалов / А. Фримен. - М.: Диалектика / Вильямс, 2015. - 961 с.

### Электронные ресурсы

- 1) <https://metanit.com/sharp/tutorial/1.1.php> - программирование на C#
- 2) <https://habr.com/ru/hub/csharp/> - основы программирования на C#
- 3) <http://progopedia.ru/language/csharp/> - самоучитель C#
- 4) <https://professorweb.ru/> - объектно-ориентированное программирование C#
- 5) <http://www.sql-tutorial.ru/> - базы данных MSSQL
- 6) <https://proglib.io/p/practice-ms-sql> - практика по MSSQL



ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ ГОРОДА МОСКВЫ  
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ  
«КОЛЛЕДЖ СВЯЗИ №54»  
ИМЕНИ П.М. ВОСТРУХИНА

**ОТЗЫВ РУКОВОДИТЕЛЯ**

(тема выпускной квалификационной работы)

сотрудников

студента

Смирновой Полины Андреевны

(фамилия, имя, отчество)

1. Актуальность работы:

Четко прослеживается актуальность данной работы, обусловленная необходимостью выполнения работ, связанных с разработкой информационных систем.

2. Отличительные положительные стороны работы:

В первой главе ВКР подробно изучен теоретический материал, касающийся информационных систем: свойства, виды и этапы разработки.

Вторая глава является логическим продолжением первой и довершает целостность ощущений от прочтения ВКР. Автор наглядно демонстрирует процесс разработки информационной системы с помощью интегрированной среды разработки Visual Studio 2016

3. Практическое значение

Продемонстрирован процесс разработки информационной системы

4. Недостатки и замечания

Работа выполнена в соответствии с требованиями, предъявляемыми к содержанию и оформлению ВКР, и заданием, выданным руководителем 20.03. 2020 г.

5. Оценка образовательных достижений студента

При выполнении ВКР Смирнова П.А проявила инициативу и самостоятельность в проведении

исследований. Показала себя как вдумчивый, опытный и инициативный студент, который способен решать различные сложные задачи в области научных исследований как теоретического, так и экспериментального характера.

6. Выводы

ВКР Смирновой Полины Андреевны актуальна, отличается значимой теоретической и практической ценностью

Выпускная квалификационная работа может быть допущена к защите и заслуживает оценки **«отлично»**

Руководитель Бастрыкина Кристина Александровна

(фамилия, имя, отчество)

подпись

преподаватель ГБПОУ Колледжа связи №54.

им. П.М Вострухина

ученая степень, ученое звание, должность, место работы