**РОСЖЕЛДОР**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ОБРАЗОВАНИЯ**

**СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ПУТЕЙ СООБЩЕНИЯ (СГУПС)**

|  |  |  |  |
| --- | --- | --- | --- |
| **К защите:** |  |  | |
| **Заведующий кафедрой** | **Информационные** | |
| **технологии транспорта** | | |
|  | д-р техн. наук, проф. | |
|  |  | В. И. Хабаров | |
| *подпись* |  | *инициалы, фамилия* | |
|  |  |  | |
| *дата* |  |  | |

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Тема:** | Разработка информационной системы учета профзаболеваний | | | | | |
|  | работников ОАО «РЖД» | | | | | |
|  | |  | БР.БИСТ.08.2022 |  |  |
|  | |  | *шифр документа* |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **Выполнил** |  |  |  | **Руководитель** |
|  |  | А.В. Дейнес |  |  |  | канд. техн. наук, доц.  В. Г. Кобылянский |
| *подпись* |  | *инициалы, фамилия* |  | *подпись* |  | *инициалы, фамилия* |
|  |  |  |  |  |  |  |
| *дата* |  |  |  | *дата* |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Консультанты по разделам** |  |  |  |  |
| 1 |  |  |  |  |
|  |  |  |  |  |
| 2 |  |  |  |  |
|  |  |  |  |  |
| 3 |  |  |  |  |
|  |  |  |  |  |
| Нормоконтролер работы |  |  |  | ст. преп.  Т. А. Распопина |
|  |  | *подпись* |  |  |
|  |  |  |  |  |
|  |  | *дата* |  |  |

**2022 г.**

**СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ПУТЕЙ СООБЩЕНИЯ (СГУПС)**

Факультет: Бизнес-информатики

Кафедра: Информационные технологии транспорта

Направление: 09.03.02 «Информационные системы и технологии»

Профиль: Интеллектуальные транспортные системы

|  |  |
| --- | --- |
|  | ***УТВЕРЖДАЮ****: зав. кафедрой «Информационные технологии транспорта»*  д-р техн. наук, проф.  В. И. Хабаров |
|  | *«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_ 2022 г.* |

**З А Д А Н И Е**

**на выполнение выпускной квалификационной работы**

|  |  |  |  |
| --- | --- | --- | --- |
| студенту | Дейнес Анне Васильевне | | |
|  |  | | |
| 1. Тема «Разработка информационной системы учета профзаболеваний работников ОАО «РЖД»» утверждена приказом № 203/с от «30» мая 2022 г. | | | |
| 2. Задание выдано «12» мая 2022 г. | | | |
| 3. Срок сдачи законченной работы на кафедру «17» июня 2022 г. | | | |
| 4. Исходные данные: данные, полученные в ходе прохождения преддипломной практики | | | |
| 5. Содержание расчетно-пояснительной записки | | | |
| Наименование разделов и вопросов | | Примерное количество страниц | График (сроки) выполнения |
| Введение | | 2 | 13.05.2022 |
| Аналитическое исследование | | 5 | 13.05.2022 |
| Проектирование информационной системы | | 5 | 15.05.2022 |
| Создание приложения | | 2 | 28.05.2022 |
| Заключение | | 1 | 01.06.2022 |

6. Содержание и объемы графической части

|  |  |  |
| --- | --- | --- |
| Наименование графического документа (чертежа, схемы, графика) | Количество  листов  формата А1 | График  (сроки)  выполнения |
| Презентация PowerPoint | 15 | 05.05.2022 |

7. Консультанты по разделам

|  |  |  |
| --- | --- | --- |
| Наименование  раздела | Фамилия, И. О.  консультанта | Подпись консультанта,  дата выдачи задания |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| Руководитель |  | В. Г. Кобылянский |
|  | *(подпись, фамилия, И.О.)* |  |
| Задание к использованию принял |  | А. В. Дейнес |
|  | *(подпись студента)* |  |

УДК 004.41

**АННОТАЦИЯ**

В работе 49 страниц, 40 рисунков, 10 источников.

Ключевые слова: *информационная система,* *автоматизация процессов, программное обеспечение, оптимизация, профессиональные заболевания.*

В работе представлен прототип информационной системы учета профессиональных заболеваний и отравлений работников ОАО «РЖД», предназначенная для автоматизации процесса сбора данных.

**ABSTRACT**

In the work 49 pages, 40 drawings , 10 sources .

Keywords*: information system prototype, process automation, software, optimization, occupational diseases.*

The paper presents a prototype of an information system for recording occupational diseases and poisonings of employees of Russian Railways, designed to automate the data collection process.

**ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ**

АИС – автоматизированная информационная система.

БД – база данных.

ВКР – выпускная квалификационная работа.

ИС – информационная система.

ЕКАСУТР – Единая Корпоративная Автоматизированная Система Управления Трудовыми Ресурсами.

SAP (System Analysis Program Development) – автоматизированная система решений.

ОАО «РЖД» – Открытое акционерное общество «Российские железные дороги».

СУБД – система управления базами данных.

ARIS (Architecture of Integrated Information Systems) – методология для моделирования бизнес-процессов организаций.

UML (Unified Modeling Language) – унифицированный язык моделирования.

ОС – операционная система.

C# – объектно-ориентированный язык программирования высокого уровня.

.NET – программная платформа компании Microsoft.

Visual Studio – среда разработки от компании Microsoft.

# СОДЕРЖАНИЕ

[Введение 6](#_Toc106156115)

[1 Описание предметной области 8](#_Toc106156116)

[1.1 Учет и анализ профессиональных заболеваний 8](#_Toc106156117)

[1.2 Общие сведения об информационных системах 9](#_Toc106156118)

[1.3 Анализ существующих информационных систем 10](#_Toc106156119)

[1.4 Анализ существующих бизнес - процессов 12](#_Toc106156123)

[2 Моделирование бизнес-процессов 14](#_Toc106156124)

[2.1 Общие сведения о языке моделирования UML 14](#_Toc106156125)

[2.2 Диаграмма вариантов использования 15](#_Toc106156126)

[2.3 Диаграмма последовательности 18](#_Toc106156127)

[2.4 Диаграмма активностей 21](#_Toc106156128)

[2.5 Диаграмма классов 22](#_Toc106156129)

[3 Архитектура ИС 24](#_Toc106156124)

[3.1 Обоснование выбора системы управления базами данных 24](#_Toc106156125)

[3.2 Обоснование выбора языка программирования](#_Toc106156126) 26

[3.3 Обоснование выбора инструментальных возможностей программной реализации 2](#_Toc106156127)7

[3.4 Описание интерфейса операционной системы](#_Toc106156126) 28

[3.5 Разработка функционала ПО](#_Toc106156126) 29

[4 Руководство пользователя 32](#_Toc106156124)

[4.1 Описание системы 32](#_Toc106156125)

[4.2 Требования к системе 33](#_Toc106156126)

[4.3 Запуск системы 34](#_Toc106156127)

[Заключение 48](#_Toc106156132)

[Список использованных источников](#_Toc106156132) 49

# ВВЕДЕНИЕ

Темой выпускной квалификационной работы, является разработка информационно – справочной системы учета профессиональных заболеваний работников, которая предназначена для автоматизации процесса учета профессиональных заболеваний и отравлений в структурных подразделениях ОАО «РЖД», обеспечения контроля за выполнением мероприятий по ликвидации и предупреждению профессиональных и отравлений.

В связи с ростом профессиональных заболеваний, а также повышением осведомленности общественности об их последствиях, большое значение приобрело внимание к различным аспектам заболеваний и улучшение охраны труда и техники безопасности. Следовательно, появилась потребность в соответствующих инструментах по управлению информацией, таких как информационные системы, которые предназначены для распознавания моделей появления профессиональных заболеваний, а также для принятия решений об их профилактике, раннем обнаружении и лечении. Информационные системы учета профессиональных заболеваний являются богатым источником информации для принятия решений в области здравоохранения. Подобные информационные системы содержат набор данных о профессиональных заболеваниях и отравлениях и работниках имеющих профессиональные заболевания. Эти данные имеют решающее значение для распознавания и предотвращения возникновения профессиональных заболеваний и отравлений. Применяются такие ИС для различных целей, таких как сбор статистической информации, диагностики и профилактики.

Актуальность темы, как проектирование части информационной системы. Данный проект должен позволить обеспечить ввод информации о профессиональных заболеваниях и отравлениях, информацию о работниках имеющие профессиональные заболевания, формировать отчетность, а также выводить эту отчетность на печать.

Объектом исследования являются случаи профессиональных заболеваний и отравлений в структурных подразделениях ОАО «РЖД».

Предметом исследования является проектирование информационной системы учета профессиональных заболеваний и отравлений работников в структурных подразделениях ОАО «РЖД».

Целью данной работы является создание программного обеспечения для автоматизации процесса сбора данных о случаях профессиональных заболеваний и отравлений в структурных подразделениях ОАО «РЖД», контроль за выполнением мероприятий по ликвидации и предупреждению профессиональных заболеваний и отравлений, обеспечение доступа к информации руководству и причастным работникам.

Задачи, которые необходимо решить для достижения поставленной цели:

* + описать предметную область;
  + смоделировать бизнес-процессы информационной системы;
  + спроектировать и реализовать информационную систему учета организационно-технических мероприятий;
  + разработка программного обеспечения;
  + разработка документации.

В качестве методов исследования, посредством которых выполнялась работа, использовались: методы синтеза, анализа и сравнения, с помощью которых рассматриваются объект и предмет исследования, а также выделяются аспекты при рассмотрении существующих решений по данной тематике, методы моделирования, использующиеся для построения диаграмм бизнес-процессов.

Отчет по выполненной работе состоит из введения, четырех разделов, в которых описана предметная область, смоделированы бизнес-процессы, разработан функционал приложения и описание интерфейса. А также заключение, список использованных источников и приложения с кодом программы.

# Описание предметной области

# Учет и анализ профессиональных заболеваний

Профессиональное заболевание – заболевание, возникающее в результате воздействия вредного производственного фактора, который может повлечь временную или стойкую утрату трудоспособности работника.

Согласно Постановлению Правительства РФ № 967 профессиональные заболевания, которые могут повлечь временную или стойкую утрату трудоспособности, делятся на:

* острые (представляют собой результат однократного воздействия на работника вредных производственных факторов);
* хронические (представляют собой результат длительного воздействия на работника этих факторов).

Острое профзаболевание может развиться в течение одного рабочего дня или смены. Его еще называют отравлением.

Обстоятельствами и условиями возникновения хронических профзаболеваний в основном являются:

* несовершенство технологических процессов;
* конструктивные недостатки средств труда;
* несовершенство рабочих мест и СИЗ;
* неприменение и отсутствие СИЗ [1].

Специфика трудовой деятельности работников железнодорожного транспорта, характеризуется воздействием на организм вредных факторов, которые приводят к ухудшению здоровья и возникновению профессиональных заболеваний.

На рисунке 1.1 представлена структура профессиональных заболеваний на железнодорожном транспорте в РФ.

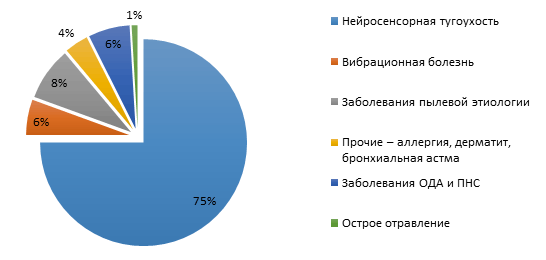


Рисунок 1.1 – Структура профессиональных заболеваний на железнодорожном транспорте в РФ

Согласно данным из структуры профессиональных заболеваний, существенно преобладает нейросенсорная тугоухость – поражение слухового аппарата из – за воздействия шума (75%), далее идут заболевания пылевой этиологии (8%), вибрационная болезнь (6%), заболевания опорно – двигательного аппарата (6%), аллергия (4%), острое отравление (1%).

# Общие сведения об информационных системах

Система – представляет собой совокупность элементов или объектов, которые связаны друг с другом и имеют некоторые общие признаки, свойства, назначения, условия существования. То есть эти элементы образуют определенную целостность [2].

Информационная система – это система, которая предназначена для сбора, хранения и обработки данных, а так же для предоставления информации. Частями информационной системы являются аппаратное обеспечение, программное обеспечение, соединения и информация компьютерной сети, пользователи информационной системы. ИС включает ресурсы для совместно используемой или обрабатываемой информации, а также людей, которые управляют системой. Люди считаются частью системы, потому что без них системы не работали бы правильно [3].

При проектировании информационной системы необходимо сначала определить цели проекта.

Цель проекта – создание программного обеспечения для введения данных по организационным и техническим мероприятиям.

Основная задача любого успешного проекта заключается в том, чтобы на момент запуска системы и в течение всего времени ее эксплуатации можно было обеспечить:

* + требуемую функциональность системы и степень адаптации к изменяющимся условиям ее функционирования;
  + требуемую пропускную способность системы;
  + требуемое время реакции системы на запрос;
  + безотказную работу системы в требуемом режиме, иными словами – готовность и доступность системы для обработки запросов пользователей;
  + простоту эксплуатации и поддержки системы;
  + необходимую безопасность.

Производительность является главным фактором, определяющим эффективность системы.

# Анализ существующих информационных систем

# В настоящее время в ОАО «РЖД» существует автоматизированный учет случаев профессиональных заболеваний и отравлений. Данные о случаях профессиональных заболеваний и работниках, которые имеют профессиональные заболевания, хранятся в системе ЕК АСУТР, которая была разработана на основе SAP технологий. Программный комплекс обеспечивает ввод информации в систему, а также обеспечивает предоставление отчетной информации пользователям. Пользователями системы являются руководители и специалисты центрального аппарата ОАО «РЖД», руководители и специалисты служб охраны труда и промышленной безопасности железных дорог, инженеры по охране труда, администраторы и технологи. Специалисты по охране труда структурных подразделений обеспечивают ввод необходимой информации в систему. А руководители и специалисты могут просматривать отчетные формы.

# На рисунке 1.2 показано ведение профзаболевания.

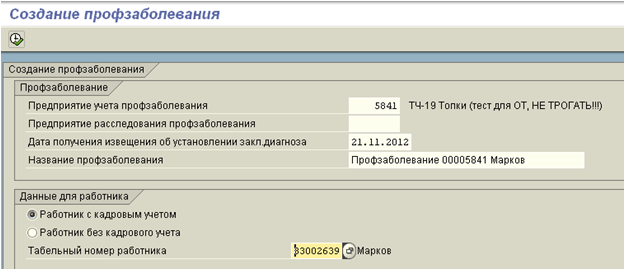


Рисунок 1.2 – Ведение профзаболевания

# На рисунке 1.3 показано добавление данных о работнике, имеющем профзаболевание.

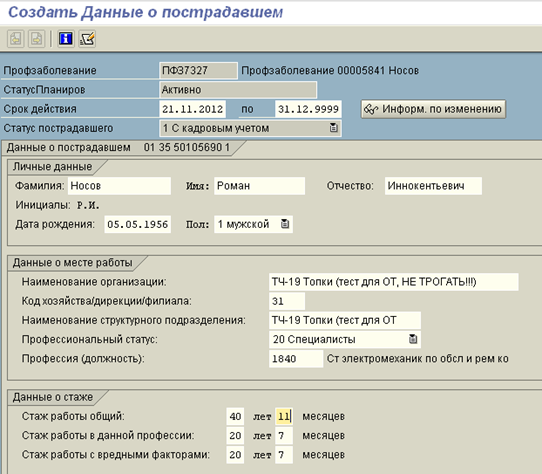


Рисунок 1.3 – Добавление данных о работнике с профзаболеванием

# Анализ существующих бизнес-процессов

Задача «Учет и анализ профессиональных заболеваний» входит в состав информационной интегрированной системы управления охраной труда в ОАО «РЖД» и предназначена для автоматизации учета и анализа информации о профессиональных заболеваниях, формирования отчетной документации, выработки профилактических рекомендаций.

Задача предназначена для специалистов и руководителей всех уровней управления.

Основные направления деятельности специалистов по охране труда, подлежащие автоматизации.

– учет профессиональных заболеваний;

– анализ профессиональных заболеваний;

– формирование отчетности по профессиональным заболеваниям для разных уровней управления.

Выявление профессионального заболевания происходит в тот момент, когда работник обращается к врачу. После постановки предварительного диагноза, направляется извещение в центр санитарно-эпидемиологического надзора, а так же сообщается работодателю в специальной форме, которая установлена Министерством здравоохранения Российской Федерации.

После получения извещения, центр санитарно-эпидемиологического надзора приступает к выяснению причин и обстоятельств, которые способствовали возникновению профессионального заболевания у работника, а так же составляется характеристика условий труда.

Полученная информация направляется работодателю, которая в дальнейшем заносится в информационную систему, предназначенную для учета профессиональных заболеваний. После занесения происходит формирование базы данных, где содержатся данные о работнике с профзаболеванием, причинах возникновения профессионального заболевания, дате постановки диагноза, сведения о трудоспособности работника, стаже работы, стаже работы с вредными факторами. А так же происходит формирование справочника, содержащего информацию о вредных факторах, наименование профзаболеваний.

На основании, полученных данных работодатель обязан организовать расследование причин и обстоятельств, из-за которых работником было получено профессиональное заболевание. Для этого необходимо создать комиссию, которая будет заниматься расследованием причин. В состав комиссии входят:

– специалист по охране труда;

– представитель учреждения здравоохранения;

– представитель работодателя.

Комиссия после проведения расследования и установления причин и обстоятельств, способствовавших возникновению профессионального заболевания у работника, составляет акт и направляет работодателю. Работодатель в свою очередь обязан в установленные сроки разработать и принять меры по предотвращению возникновения профессионального заболевания.

После составления акта о профессиональном заболевании специалист по охране труда структурного подразделения вносит в систему данные о запланированных мероприятиях по ликвидации и предупреждению профессиональных заболеваний

По мере выполнения мероприятий специалист по охране труда структурного подразделения вносит в систему данные об этом.

# Моделирование бизнес-процессов

# Общие сведения о языке моделирования UML

Унифицированный язык моделирования (Unified Modeling Language, UML) – язык моделирования общего назначения, который предназначен для того чтобы можно было визуализировать, специфицировать, конструировать и документировать системы. UML не является языком программирования, а представляет собой скорее визуальный язык, основной целью которого является определение стандартного способа визуализации того, как была спроектирована система. Диаграммы UML используются, чтобы изобразить поведение и структуру системы [4].

Грэди Буч, Ивар Якобсон и Джеймс Рамбо создали унифицированный язык моделирования в 1995 году, работая в Rational Software. В 1997 году Object Management Group приняла UML в качестве стандарта для своих членов, в том числе таких, как Hewlett-Packard, IBM и Apple Computer.

В UML выделяют девять типов диаграмм:

* + диаграммы классов (Class diagram);
  + диаграммы объектов (Object diagram);
  + диаграммы вариантов использования (Usecase diagram);
  + диаграммы последовательностей (Sequence diagram);
  + диаграммы кооперации (Collaboration diagram);
  + диаграммы состояний (State diagram);
  + диаграммы деятельности (Activity diagram);
  + диаграммы компонентов (Component diagram);
  + диаграммы развертывания (Deployment diagram).

Нотация UML может быть полезна на этапе сбора требований проекта. Например, диаграммы вариантов использования покажут пользователям, что они должны делать, а также то, что система сделает для них. Также можно использовать диаграммы как основу для тестирования приложения (создания планов тестирования) и разработки руководств пользователя.

# Диаграмма вариантов использования

Диаграмма вариантов использования – представляет собой поведенческую диаграмму UML, которая часто используется для описания поведения системы.

Диаграммы вариантов использования применяются для сбора требований к системе, которые включают внутренние и внешние воздействия. Когда система анализируется для определения ее функциональных возможностей, подготавливаются сценарии и определяются участники (актеры).

Диаграммы вариантов использования определяют события системы и их потоки. Но диаграмма вариантов использования никогда не описывает, как они реализованы. Эти схемы используются на очень высоком уровне проектирования. Этот высокоуровневый дизайн совершенствуется снова и снова, чтобы получить полное и практическое представление о системе [5].

Вкратце, можно сказать, что целями диаграммы вариантов использования являются:

– сбор требований к системе;

– получение внешнего образа системы;

– определение внешних и внутренних факторов, которые влияют на систему;

– показать взаимодействия между требованиями и действующими лицами.

Диаграмма вариантов использования состоит из трех компонентов: актеры, варианты использования, ассоциации. Все компоненты более подробно описаны ниже.

Актер на диаграмме вариантов использования – представляет собой любой объект, который выполняет роль в данной системе. Это может быть человек, организация или внешняя система.

Вариант использования – представляет собой функцию или действие в системе. Изображается в виде овала. Название варианта очень важно, оно должно быть подобрано таким образом, чтобы можно было определить выполняемые функции.

Ассоциации – это связь между актером и вариантами использования.

На рисунке 2.1 показана диаграмма вариантов использования пользователя. Пользователь обеспечивает ввод необходимой информации в систему, может редактировать введенную информацию, формирует отчетность а также может отправлять отчеты на распечатку, просматривает списки с работниками, имеющие профзаболевания.

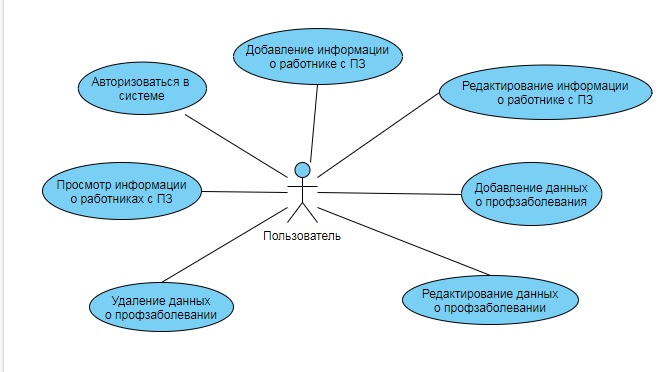


Рисунок 2.1 – Диаграмма вариантов использования пользователя

На рисунке 2.2 представлена диаграмма вариантов использования для администратора. Администратор занимается добавлением новых пользователей, изменяет и удаляет информацию о них, просматривает списки пользователей, просматривает информацию о работниках с профзаболеваниями, изменяет и удаляет ее, просматривает списки работников, имеющих профзаболевания. Также администратор может просматривать и редактировать справочник с профзаболеваниями.

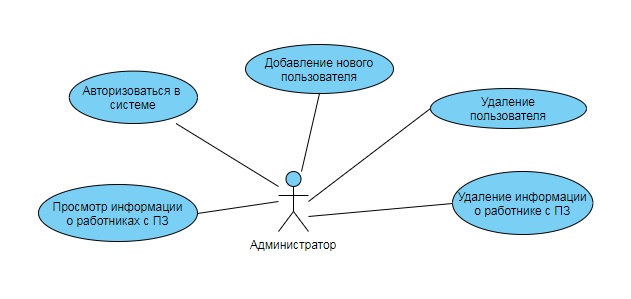


Рисунок 2.2 – Диаграмма вариантов использования администратора

На рисунке 2.3 представлена Sub-диаграмма для добавления новых пользователей в систему.

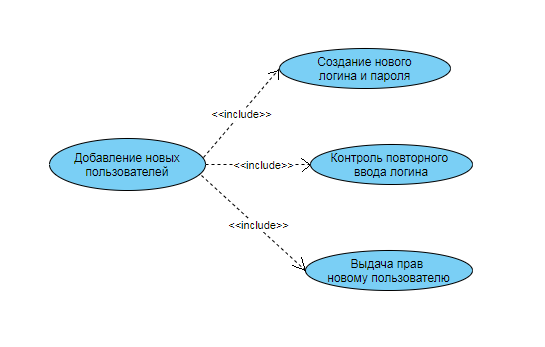


Рисунок 2.3 – Sub-диаграмма «Добавление новых пользователей»

На рисунке 2.4 представлена диаграмма вариантов использования для пользователя – аналитика. Пользователь-аналитик занимается анализом информации, формирует отчетность, следит за проведением мероприятий по ликвидации и предупреждению профзаболеваний.

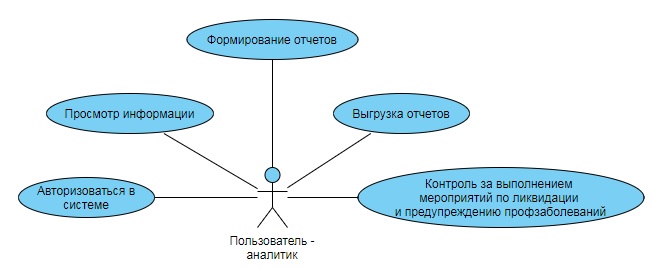


Рисунок 2.4 – Диаграмма вариантов использования пользователя-аналитика

# Диаграмма последовательности

Диаграммы последовательности описывают взаимодействия между объектами с точки зрения обмена сообщениями во времени. Также их называют диаграммами событий. Диаграмма последовательности состоит из группы объектов и классов, представленных линиями жизни, которые были задействованы в сценарии, и последовательности сообщений, передающиеся между объектами и классами. Диаграммы также могут отображать управляющие структуры между объектами [6].

Основная цель диаграммы последовательности – определить последовательности событий, которые приводят к желаемому результату. Акцент делается не столько на самих сообщениях, сколько на порядке их появления; тем не менее, большинство диаграмм последовательностей сообщают, какие сообщения передаются между объектами системы, а также порядок, в котором они происходят. Диаграмма передает эту информацию по горизонтали и вертикали: вертикальное измерение показывает сверху вниз временную последовательность сообщений/вызовов по мере их возникновения, а горизонтальное измерение показывает слева направо экземпляры объектов, которым отправляются сообщения.

Диаграмма последовательности состоит из следующих компонентов: линия жизни, актер, сообщения, фрагменты, активность.

Линии жизни представляют роли или экземпляры объектов, которые участвуют в моделируемой последовательности. При рисовании диаграммы последовательности элементы обозначения линии жизни размещаются в верхней части диаграммы. Линии жизни нарисованы в виде прямоугольника с пунктирной линией, спускающейся от центра к нижнему краю.

Первое сообщение диаграммы последовательности всегда начинается сверху и для удобства чтения обычно располагается в левой части диаграммы. Последующие сообщения добавляются к диаграмме немного ниже предыдущего сообщения. Чтобы показать объект (то есть линию жизни), отправляющий сообщение другому объекту, рисуется линия к принимающему объекту сплошной стрелкой. Имя сообщения или метода помещается над линией со стрелкой.

Активность обозначает время, необходимое объекту для выполнения задачи. Когда объект занят выполнением процесса или ожидает ответного сообщения, используется тонкий прямоугольник, который располагается вертикально на линии жизни.

Фрагменты представляют собой логические группы, представленные прямоугольником, которые содержат условные структуры, влияющие на поток сообщений.

Актер – роль, которую пользователь или другие субъекты.

На рисунке 2.5 представлена диаграмма последовательности для процесса взаимодействия пользователя с программным обеспечением, ввод запроса и получения ответа.

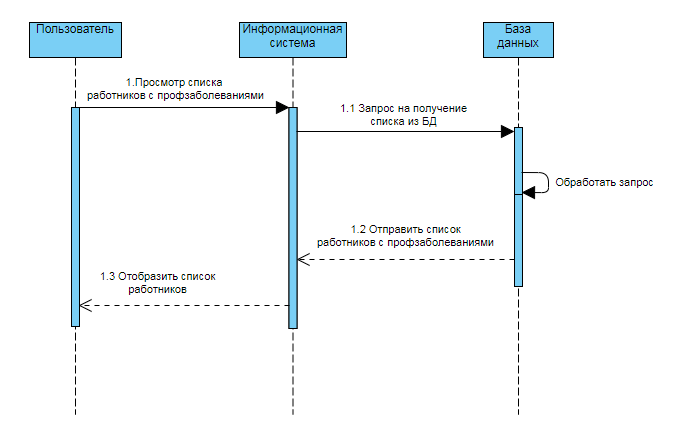


Рисунок 2.5 – Просмотр списка работников с профессиональными заболеваниями

На рисунке 2.6 представлена диаграмма последовательности для процесса составления базы данных.

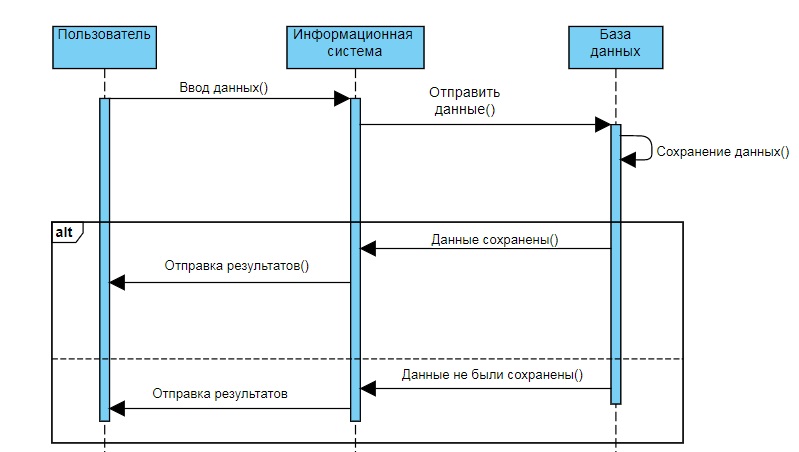


Рисунок 2.6 – Формирование базы данных

# Диаграмма активностей

Диаграмма активностей – тип диаграммы, которая иллюстрирует поток управления в системе и обращается к этапам выполнения варианта использования. Диаграмма активностей фокусируется на состоянии потока и последовательности, в которой он происходит. То есть описывается или изображается то, что вызывает конкретное событие.

Диаграмма активностей – это, по сути, блок-схема, представляющая поток от одного действия к другому. Действие можно описать как работу системы.

Поток управления переходит от одной операции к другой. Этот поток может быть последовательным, разветвленным или параллельным.

Активность - это особая операция системы. Диаграммы активности используются не только для визуализации динамической природы системы, но они также используются для построения исполняемой системы с использованием методов прямого и обратного проектирования.

Целями диаграммы активности являются:

– описание потока активности;

– описание последовательности от одного действия к другому;

– описание разветвлений в системе.

Диаграммы активностей в основном используются в качестве блок-схемы, которая состоит из действий, выполняемых системой. Диаграммы активностей – это не совсем блок-схемы, поскольку они обладают некоторыми дополнительными возможностями.

Диаграмма активностей состоит из следующих компонентов: деятельность, ассоциация, условия, ограничения.

Действие – это спецификация параметризованной последовательности поведения. Действие отображается в виде прямоугольника с закругленными углами, охватывающего все действия, потоки управления и другие элементы, составляющие действие.

Ассоциация – показывает поток управления от одного действия к другому. Обозначается линией со стрелкой.

Условия – принятие решения, имеет защитные условия, которые позволят управлять потоком, если защитное условие выполнено.

Диаграмма активностей для поиска информации о работнике с профзаболеванием на рисунке 2.7.

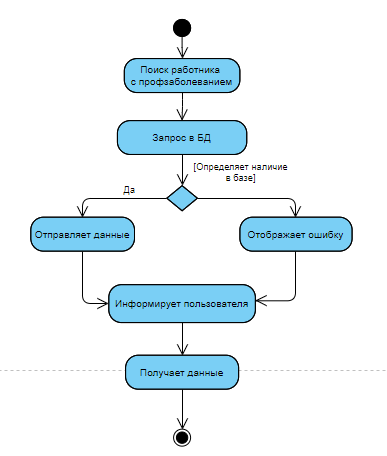


Рисунок 2.7 – Диаграмма активностей поиск информации о работнике с профзаболеванием

# Диаграмма классов

Диаграмма классов отображает статическое представление приложения. Он представляет типы объектов, находящихся в системе, и отношения между ними. Класс состоит из своих объектов, а также может наследовать от других классов. Диаграмма классов используется для визуализации, описания, документирования различных аспектов системы, а также для создания исполняемого программного кода.

Основное назначение диаграмм классов - построить статическое представление приложения. Это единственная диаграмма, которая широко используется для построения, и ее можно отображать с помощью объектно-ориентированных языков. Это одна из самых популярных диаграмм UML.

Ниже приведены цели диаграмм классов:

– анализирует и создает статическое представление системы;

– описывает основные обязанности системы;

– предоставляет общую схему системы для лучшего понимания.

Диаграмма классов состоит из следующих компонентов: класс, отношения.

Класс – это представление похожих объектов, которые имеют одинаковые отношения, атрибуты, операции и семантику.

Отношения – это семантическая связь между двумя и более классами, при этом изменения в одном классе вызывают изменения в другом.

Диаграмма классов представлена на рисунке 2.8.

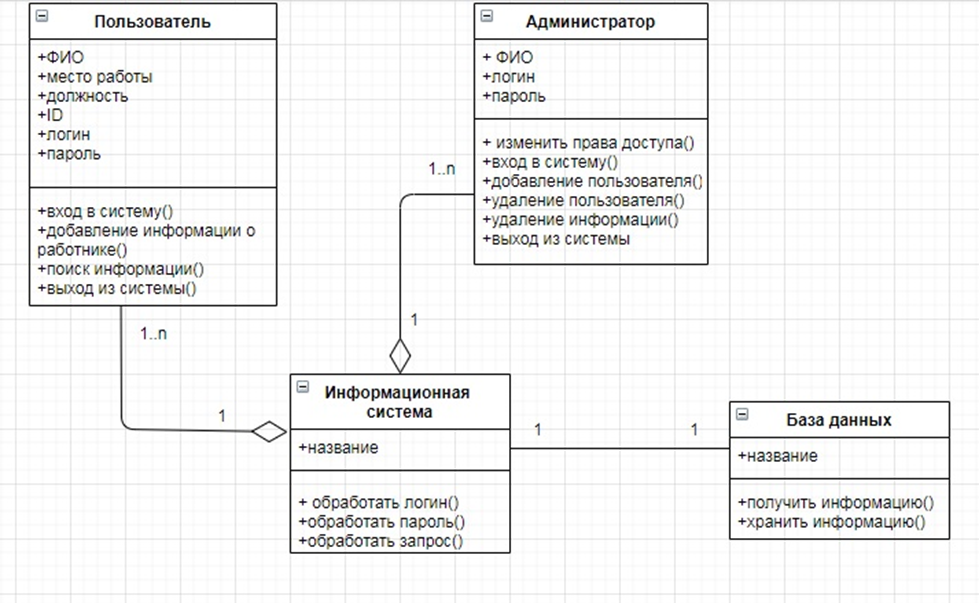


Рисунок 2.8 – Изменение содержимого итоговой таблицы

**3 Архитектура ИС**

Информационная система учета профессиональных заболеваний создавалась в два этапа. Во время первого этапа была создана информационная база данных, где хранятся данные о профессиональных заболеваниях и работниках, имеющие профессиональные заболевания. Вторым этапом была разработка прикладного приложения, связанного с базой данных.

Таким образом, необходимо рассмотреть два программных продукта, которые были использованы при создании информационной системы.

**3.1 Обоснование выбора системы управления базами данных**

Система управления базами данных (СУБД) – это программное обеспечение, предназначенное для хранения и обработки данных. Она состоит из группы программ, которые позволяют манипулировать данными. СУБД позволяет пользователям создавать свои собственные базы данных, которые соответствуют их требованиям [7].

Среди наиболее ярких представителей систем управления базами данных, построенные на технологии «клиент-сервер», были выделены следующие:

– Oracle;

– MySQL;

– Microsoft SQL Server;

– Microsoft Access;

– SQLite;

– MS SQL Server;

– Microsoft Visual FoxPro.

В качестве СУБД была выбрана Microsoft SQL Server.

Microsoft SQL Server – это система управления реляционными базами данных. Данная СУБД построена на основе SQL, который является стандартным языком программирования, предназначенный для взаимодействия с реляционными базами данных. Microsoft SQL Server связана с Transact-SQL, или T-SQL, реализацией SQL от Microsoft. Проще говоря, MS SQL используется, когда необходимо хранить или извлекать данные из других программных приложений [8].

Основным компонентом SQL Server является компонент Database Engine. Чтобы пользователи могли взаимодействовать с реляционной СУБД, их запрос на допустимом языке запроса/базы данных должен быть переведен в запрос SQL, прежде чем механизм SQL сможет его обработать. Механизм хранения SQL записывает и извлекает данные с сервера хранилища данных, что часто выполняется путем преобразования данных в совместимый формат. Для извлечения данных обработчик запросов принимает, анализирует и выполняет команды SQL для хранилища данных для пересылки на сервер приложений. Сервер приложений обрабатывает запрос SQL и отправляет его на веб-сервер, где клиент может получить доступ к информации через таблицы данных SQL.

Механизм хранения SQL-сервера — это программное обеспечение, которое используется для работы, создания, чтения и обновления данных между диском и памятью. SQL-сервер сопоставляет базу данных с файлами, в которых хранятся объекты базы данных, таблицы и индексы. Затем эти файлы можно хранить в файловой системе FAT или NTFS. Существует три основных типа файлов базы данных SQL Server:

* первичный файл данных;
* вторичный файл данных;
* файл журнала транзакций.

Microsoft SQL Server имеет множество различных функций и является мощным инструментом, именно поэтому для реализации информационной системы была выбрана эта СУБД.

**3.2 Обоснование выбора языка программирования**

Для реализации проекта был выбран объектно-ориентированный язык программирования C#.

По своей сути C# – это объектно-ориентированный, универсальный и безопасный язык программирования. C# предоставляет обширную поддержку для разработки объектно-ориентированных приложений. Однако предметная область объектно-ориентированного программирования обширна [9].

С помощью C# программисты могут создавать различные надежные, долговечные и безопасные приложения, работающие в .Net. Известно, что этот фреймворк позволяет создавать многоплатформенные приложения, что очень полезно, поскольку экономит время разработки, если возникает необходимость запускать приложения на разных платформах.

Существует множество причин почему C# позволяет создавать надежные приложения. Во-первых, сборщик мусора .Net автоматически выделяет и освобождает память приложения. С защитой типов, допускающих значение NULL, также существует минимальный риск того, что код вызовет исключение System.NullReferenceException во время выполнения, которое обычно отражает ошибку разработчика. Кроме того, поскольку в C# предусмотрена обработка исключений, он обеспечивает эффективное обнаружение и устранение ошибок.

Поскольку C# принадлежит к семейству языков C, он упрощает написание эффективного кода, тем самым повышая производительность во время выполнения. Более того, этот язык программирования постоянно совершенствуется новыми функциями и имеет множество доступных ресурсов для повышения уровня опыта разработчиков.

C# – это объектно-ориентированный язык, поэтому обслуживание, разработка и отладка становятся проще. Среди других преимуществ перед процедурным программированием — четкая структура программ и полное повторное использование приложений с меньшим количеством кода и более коротким временем.

**3.3 Обоснование выбора инструментальных возможностей программной реализации задачи**

IDE расшифровывается как интегрированная среда разработки. И она используется для создания среды для разработчиков и предоставления комплексных функций для разработки и тестирования программного обеспечения. Как правило, IDE содержит редактор исходного кода, некоторые инструменты автоматизации и отладчик для тестирования.

В качестве интегрированной среды разработки была выбрана Microsoft Visual Studio 2019.

Помимо многих традиционных функций IDE, Visual Studio также предлагает набор опций, специфичных для экосистемы Microsoft, благодаря выбору специальных подключаемых модулей Azure. Возможность создавать и поддерживать .NET Framework (не .NET Core) является еще одним важным преимуществом Visual Studio.

Visual Studio поддерживает множество языков программирования, в том числе серверные языки. А средства обработки данных SQL Server (SSDT) всегда используются для создания серверных приложений. Новая модель доставки данных доступна в Visual Studio 2019 для разработки проектов, связанных с SQL, таких как серверные приложения, проекты служб анализа, проекты служб отчетов, проекты служб интеграции [10].

Выбор данной среды разработки обусловлен тем, что она имеет понятный интерфейс, широкий набор инструментов для разработки приложений, есть возможность работы с данными различного типа, имеется интеграция с различными языками программирования. Так же существует возможность создания нового проекта базы данных и импортировать схему базы данных из существующей базы данных, файла сценария .sql или приложения уровня данных.

**3.4 Описание интерфейса информационной системы**

В ходе построения диаграмм, описывающих бизнес-процессы, были выделены ключевые сущности, между которыми были установлены связи. На основе выделенных сущностей, была построена ER-диаграмма. Результат построения диаграммы можно увидеть на рисунке 3.1.

Диаграмма ER означает диаграмму отношений сущностей, также известную как ERM, — это диаграмма, которая отображает взаимосвязь наборов сущностей, хранящихся в базе данных. Другими словами, ER-диаграммы помогают объяснить логическую структуру баз данных.



Рисунок 3.1 – ER-диаграмма

Основываясь на построенной диаграмме, была реализована рабочая база данных и создано прикладное приложение.

**3.5 Разработка функционала ПО**

Описание классов, методов и полей.

Класс MainWindow содержит описание главного окна.

Поля:

* MainWindow – экземпляр класса MainViewModel, предназначенный для доступа полям и методам.

Методы:

* Window\_Loaded() – метод, который срабатывает при первой загрузке окна;
* btnSubmit\_Click () – метод, срабатывающий при нажатии кнопки.

Класс CreateEmployee содержит описание формы добавления нового работника с профзаболеванием.

Поля:

* IDNumber – представляет из себя целочисленную переменную, где хранится идентификатор для записи в базу данных;
* Name, Surname, Patronymic, Citizenship, Email, PlaceOfWork, Post, Disease, CauseOfDisease, Disability – строковая переменная для хранения информации о работнике в текстовом формате;
* DateOfBirth, EmploymentDate, DateOfReceiving, BeginningDate, EndDate – переменные в формате даты;
* Age, PhoneNumber, TotalExperience, Experience, ExperienceFactors – целочисленные переменные, хранящие данные о работнике в числовом формате.

Методы:

* GetBothGender() – метод, который срабатывает при выборе пола;
* GetAllAvailableDiseases() – метод, который срабатывает при изменении поля с диагнозом;
* SaveExecute() – метод, срабатывающий при нажатии кнопку сохранения данных о работнике;
* CloseExecute() – метод, который закрывает форму добавления нового работника.
* DeleteExecute() – метод, который срабатывает при нажатии на кнопку удаления данных.

Класс updateEmployee содержит описание формы, где можно изменить данные о работнике с профзаболеванием.

Поля:

* updateIDNumber – представляет из себя целочисленную переменную, где хранится идентификатор для записи в базу данных;
* updateName, updateSurname, updatePatronymic, updateCitizenship, updateEmail, updatePlaceOfWork, updatePost, updateDisease, updateCauseOfDisease, updateDisability – строковая переменная для хранения информации о работнике в текстовом формате;
* updateDateOfBirth, updateEmploymentDate, updateDateOfReceiving, updateBeginningDate, updateEndDate – переменные в формате даты;
* updateAge, updatePhoneNumber, updateTotalExperience, updateExperience, updateExperienceFactors – целочисленные переменные, хранящие данные о работнике в числовом формате.

Методы:

* GetBothGender() – метод, который срабатывает при выборе пола;
* GetAllAvailableDiseases() – метод, который срабатывает при изменении поля с диагнозом;
* SaveExecute() – метод, срабатывающий при нажатии кнопку сохранения данных о работнике;
* CloseExecute() – метод, который закрывает форму изменения данных о работнике.

Класс CreateUser содержит описание формы добавления нового пользователя.

Поля:

* IDNumber – представляет из себя целочисленную переменную, где хранится идентификатор для записи в базу данных;
* Name, Surname, Patronymic, Citizenship, Email, PlaceOfWork, Post, Username – строковая переменная для хранения информации о пользователе системы в текстовом формате;
* DateOfBirth, EmploymentDate – переменные в формате даты;
* Age, PhoneNumber, Password – целочисленные переменные, хранящие данные о пользователе в числовом формате.

Поля:

* SaveExecute()– метод, срабатывающий при нажатии кнопку сохранения данных о пользователе;
* CloseExecute()– метод, который закрывает форму добавления нового пользователя;
* DeleteExecute() – метод, который срабатывает при нажатии на кнопку удаления данных.

Класс CreateDiseases содержит описание формы добавления нового пользователя.

Поля:

* IDNumber – представляет из себя целочисленную переменную, где хранится идентификатор для записи в базу данных;
* TypeOfDisease, HarmfulFactor, Symptoms – строковая переменная для хранения информации о заболевании в текстовом формате.

**4 Руководство пользователя**

Данный раздел представляет собой рабочую документацию информационной системы учета профессиональных заболеваний работников ОАО «РЖД»

**4.1 Описание системы**

4.1.1 Наименование системы.

Наименование системы – «Информационная система учета профессиональных заболеваний работников ОАО «РЖД».

4.1.2 Область применения.

Данная информационная система предназначена для автоматизации учета и анализа случаев профессиональных заболеваний в ОАО «РЖД».

4.1.3 Краткое описание возможностей ИС.

Информационная система учета профессиональных заболеваний должна обеспечить автоматизацию процесса сбора данных о случаях профессиональных заболеваний и отравлений, контроль за выполнением мероприятий по ликвидации и предупреждению профессиональных заболеваний, получение разнообразного вида аналитических справок.

4.1.4 Уровень подготовки пользователей.

Пользователями системы являются администраторы и технологи, инженеры по охране труда структурных подразделений ОАО «РЖД», руководители и специалисты служб охраны труда и промышленной безопасности, руководители и специалисты служб охраны труда и промышленной безопасности железных дорог, руководители и специалисты центрального аппарата Управления ОАО «РЖД».

До начала работы с программами ИС персонал, обеспечивающий работоспособность комплекса, должен пройти подготовку, обеспечивающую полноценное использование ОС Windows, средств Microsoft Office.

**4.2 Требования к системе**

4.2.1 Требования к защите информации от несанкционированного доступа.

Доступ к функциям и данным ИС разрешается только зарегистрированным пользователям.

Любые данные, предоставляемые ИС, могут быть переданы причастным работникам, не относящимся к числу пользователей, полностью или частично. Ответственность за правомерность передачи данных несет работник, предоставляющий информацию. Если в выходных документах будет содержаться информация, которая не должна передаваться, пользователь системы должен перед передачей удалить ее из передаваемого документа.

4.2.2 Требования к доступности системы.

Доступ к ИС должен быть обеспечен стандартными средствами системного программного обеспечения. Доступ к справочной системе, к вводу информации должен быть обеспечен через режим идентификации. Должен быть разграничен доступ по типу пользователей: администраторов, работников функциональных филиалов и структурных подразделений ОАО «РЖД».

4.2.3 Требования по обеспечению информационной безопасности

Так как в системе будут содержаться персональные данные, а так же данные медицинского о работниках, необходимо обеспечить защиту этих данных.

Система защиты информации ИС должна соответствовать законодательству Российской Федерации, нормативным документам ФСБ и ФСТЭК России и организационно-распорядительным документам ОАО «РЖД» по обеспечению информационной безопасности и защите персональных данных.

**4.3 Запуск системы**

4.3.1 Авторизация в системе.

При входе в систему появляется окно авторизации пользователя, где необходимо ввести логин и пароль, что видно на рисунке 4.1.

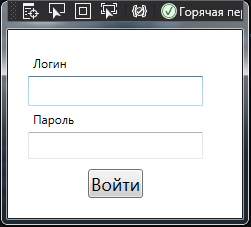


Рисунок 4.1 – Вход в систему

Если логин или пароль будут введены неверно, то появится окно, сообщающее это, что продемонстрировано на рисунке 4.2. После этого необходимо повторить попытку.

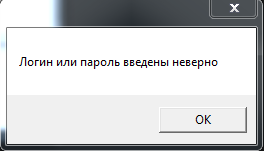


Рисунок 4.2 – Сообщение об ошибке

После входа в систему под учетными данными администратора появляется окно со всеми функциями системы это видно на рисунках 4.3 – 4.4, так как администратор имеет расширенные права доступа к информационной системе. В расширенные права входит добавление новых пользователей, просмотр и редактирование списка пользователей системы, добавление новых профзаболеваний, добавление новых работников с профзаболеваниями, а так же просмотр и редактирование списка работников с профзаболеваниями.

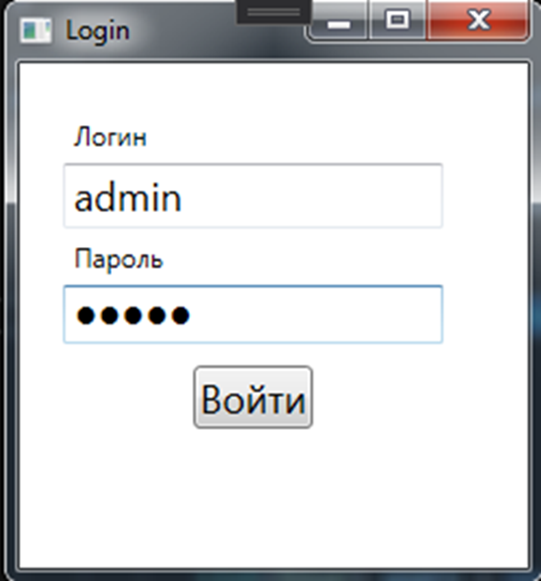


Рисунок 4.3 – Вход в систему под учетными данными администратора

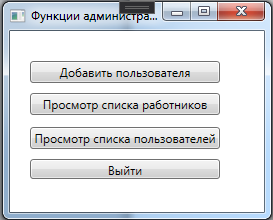


Рисунок 4.4 – Функции администратора

4.3.2 Добавление нового пользователя системы.

Для того чтобы добавить нового пользователя системы необходимо нажать на кнопку «Добавить пользователя». После нажатия появится форма, которую можно увидеть на рисунке 4.5 где необходимо ввести данные нового пользователя:

– идентификационный номер;

– фамилия, имя, отчество;

– дату рождения;

– возраст;

– пол;

– номер телефона;

– электронную почту;

– место работы;

– дату приема на работу;

– логин;

– пароль.

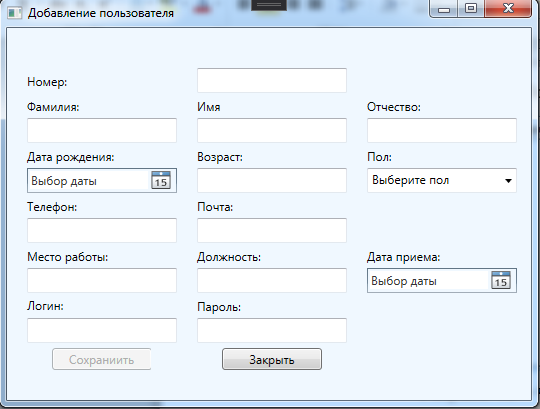


Рисунок 4.5 – Форма добавления нового пользователя.

Если данные будут введены некорректно или будут не введены, то сохранить нового пользователя будет невозможно. Как только все данные будут введены верно, можно будет добавить нового пользователя, который появится в списке. Это продемонстрировано на рисунках 4.6 – 4.8.

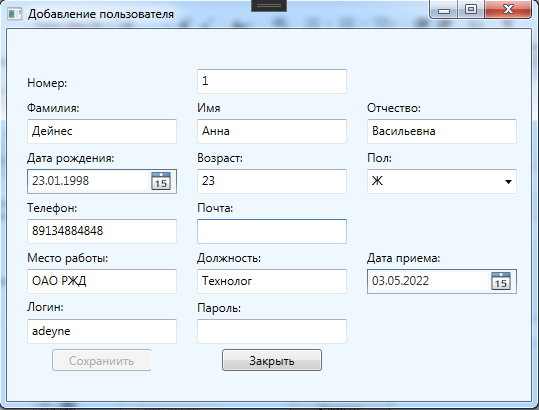


Рисунок 4.6 – Неверно введенные данные

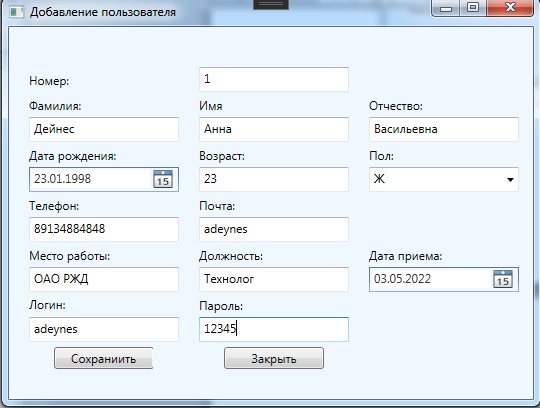


Рисунок 4.7 – Верно введенные данные

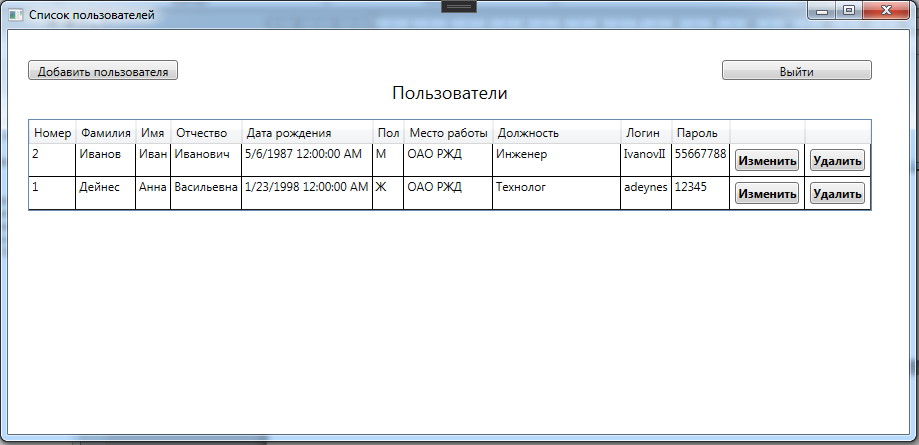


Рисунок 4.8 – Список пользователей

4.3.3 Изменение данных о пользователе.

Для того чтобы изменить данные о пользователе необходимо нажать на кнопку «Изменить», после этого появится аналогичная форма представленная на рисунке 4.7, где можно редактировать нужные данные после чего необходимо нажать на кнопку сохранить. Также существует возможность удаления существующих пользователей для этого необходимо нажать на кнопку «Удалить», после чего появится окно, подтверждающее это действие. Все результаты показаны на рисунках 4.9 – 4.12.

После выполнения необходимых действий, связанных с добавлением, изменением, удалением пользователей, можно нажать на кнопку «Выйти», после чего мы вернемся к форме с функциями администратора.

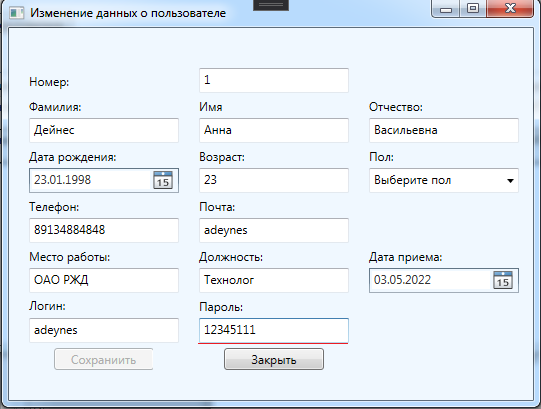


Рисунок 4.9 – Изменение данных о пользователе

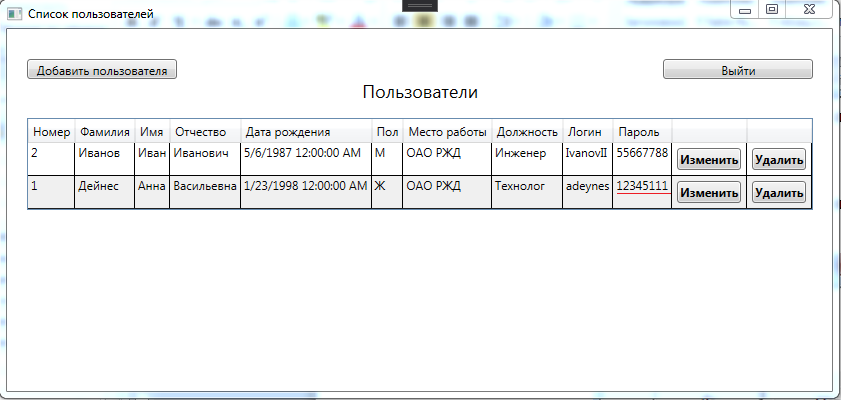


Рисунок 4.10 – Результат изменения данных о пользователе в таблице

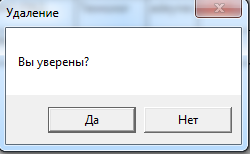


Рисунок 4.11 – Удаление пользователя

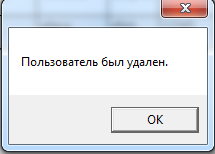


Рисунок 4.12 – Сообщение об удалении пользователя

4.3.4 Добавление нового профзаболевания.

Для того чтобы добавить новое профзаболевание необходимо в форме с функциями администратора нажать на кнопку «Просмотр списка работников», после чего появится новая форма, где необходимо будет перейти во вкладку «Профзаболевания». Результат показан на рисунке 4.13.

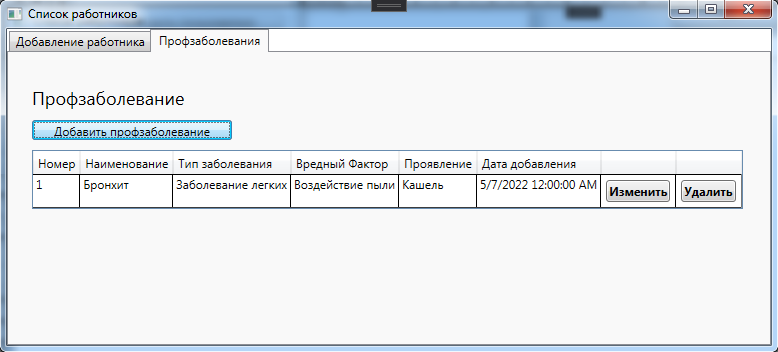


Рисунок 4.13 – Вкладка «Профзаболевания»

В появившейся вкладке можно увидеть таблицу, где будут храниться все внесенные профзаболевания и информация о них.

Для добавления нового профзаболевания следует нажать на кнопку «Добавить профзаболевание», после этого появится новая форма, где необходимо внести следующие данные:

– номер;

– наименование;

– вид заболевания;

– вредный фактор;

– проявление;

– дата добавления.

После введения всех данных необходимо нажать на кнопку «Сохранить», после чего текущая форма закроется, а новое профзаболевание появится в таблице. Результаты продемонстрированы на рисунках 4.14 – 4.15.

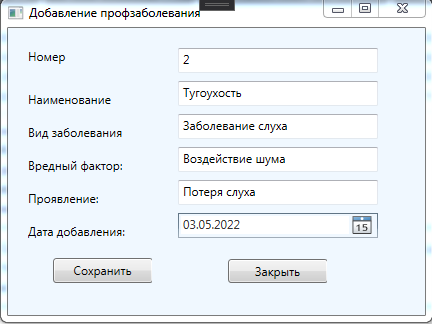


Рисунок 4.14 – Добавление профзаболевания

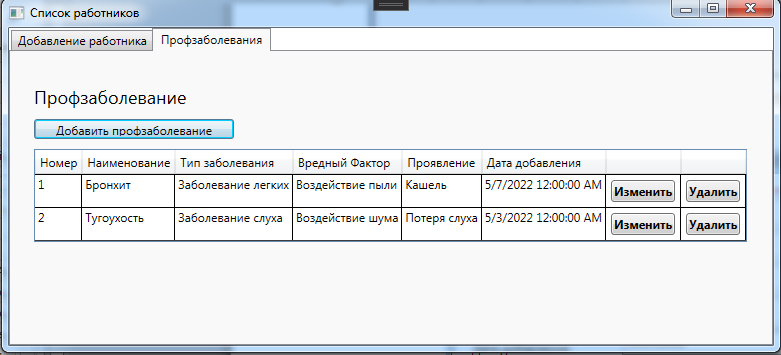


Рисунок 4.15 – Результат добавления нового профзаболевания

Если какие то данные будут введены некорректно или не введены вовсе, то данное поле подсветится красной рамкой и данные будет невозможно сохранить, это продемонстрировано на рисунке 4.16.

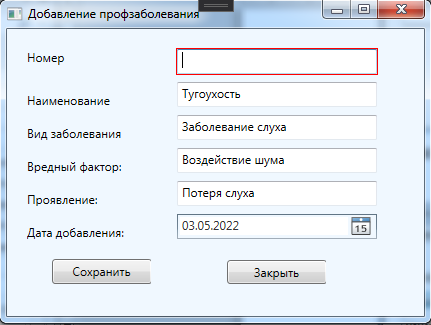


Рисунок 4.16 – Некорректно введенные данные

Для изменения или удаления данных необходимо нажать на соответствующие кнопки напротив интересующего профзаболевания. При нажатии на кнопку «Изменить» вновь появится форма, где можно изменить данные о существующем профзаболевании, после чего необходимо нажать на кнопку «Сохранить», появится сообщение, что данные были изменены и форма закроется. Изменения отобразятся в таблице.

Результаты представлены на рисунках 4.17 – 4.19.

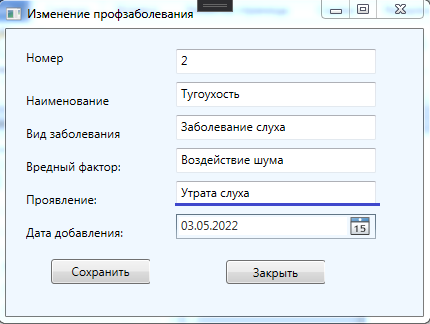


Рисунок 4.17 – Изменение профзаболевания

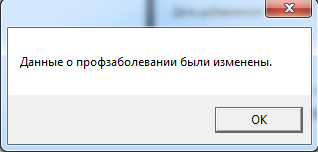


Рисунок 4.18 – Сообщение об успешном изменении данных о профзаболевании

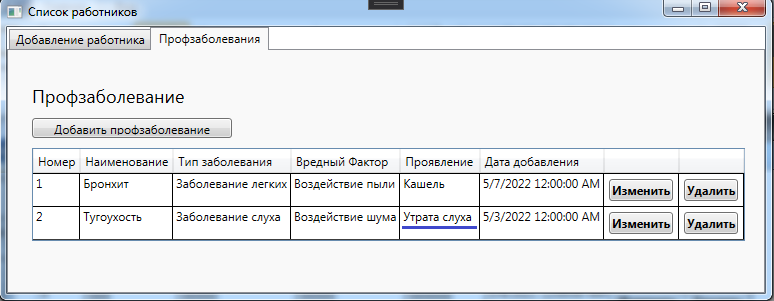


Рисунок 4.19 – Результат изменения данных о профзаболевании в таблице

Для удаления профзаболевания, необходимо нажать на кнопку «Удалить» после чего появится окно, подтверждающее это действие и сообщение об успешном удалении, после чего профзаболевание исчезнет из таблицы, что продемонстрировано на рисунках 4.20 – 4.21.

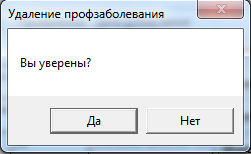


Рисунок 4.20 – Удаление профзаболевания

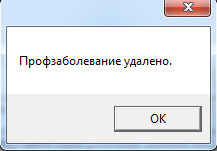


Рисунок 4.21 – Сообщение об удалении профзаболевания

4.3.5 Добавление работника имеющего профзаболевание.

После заполнения таблицы профзаболеваний, можно добавить работников с профзаболеваниями. Для этого необходимо переместиться во вкладку «Добавление работника», где также есть таблица со списком всех работников, имеющих профзаболевания. Далее необходимо нажать на кнопку «Добавить работника», после чего появится форма, которую нужно будет заполнить. В которую следует внести следующие данные:

– номер;

– фамилия, имя, отчество;

– дата рождения;

– пол;

– место рождения;

– номер телефона;

– почта;

– место работы;

– должность;

– дата приема на работу;

– общий стаж;

– стаж в данной профессии;

– стаж работы с вредными факторами;

– заболевание;

– форма заболевания;

– дата получения;

– наименование лечебного учреждения;

– утрата трудоспособности;

– срок утраты трудоспособности.

После чего пользователю следует нажать на кнопку «Сохранить», после чего открытая форма закроется, а новый работник с профзаболеванием появится в таблице. Результаты выполнения представлены на рисунках 4.22 – 4.24.

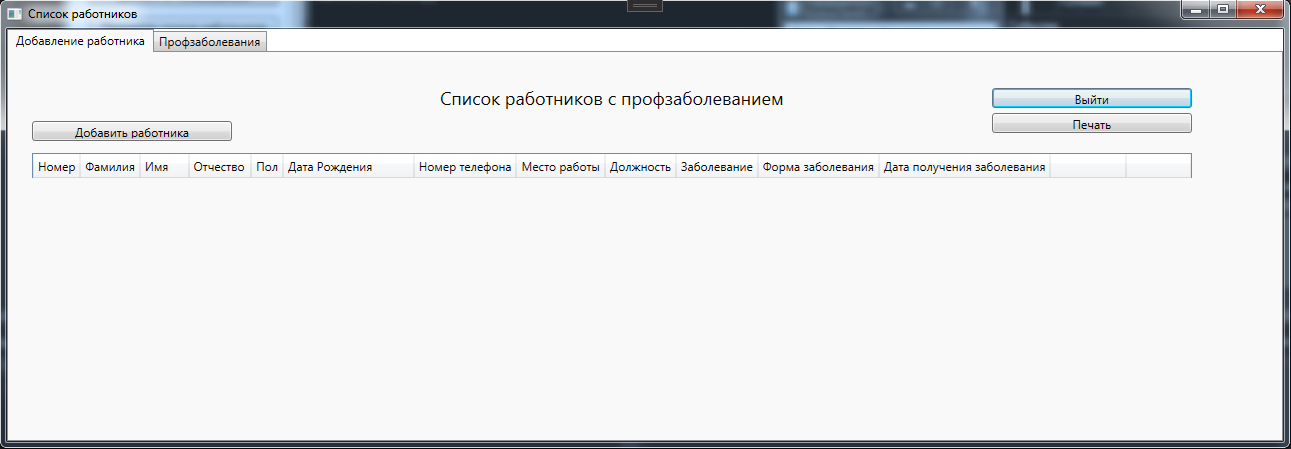


Рисунок 4.22 – Вкладка «Добавление работника»

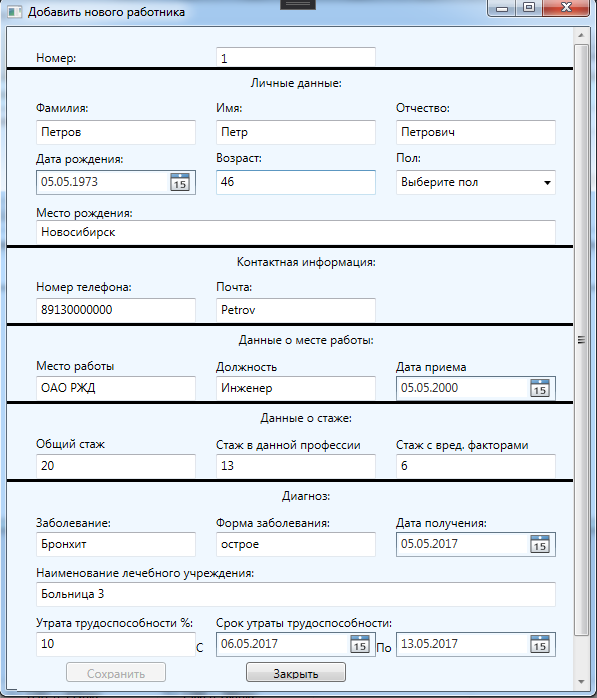


Рисунок 4.23 – Форма добавления нового работника

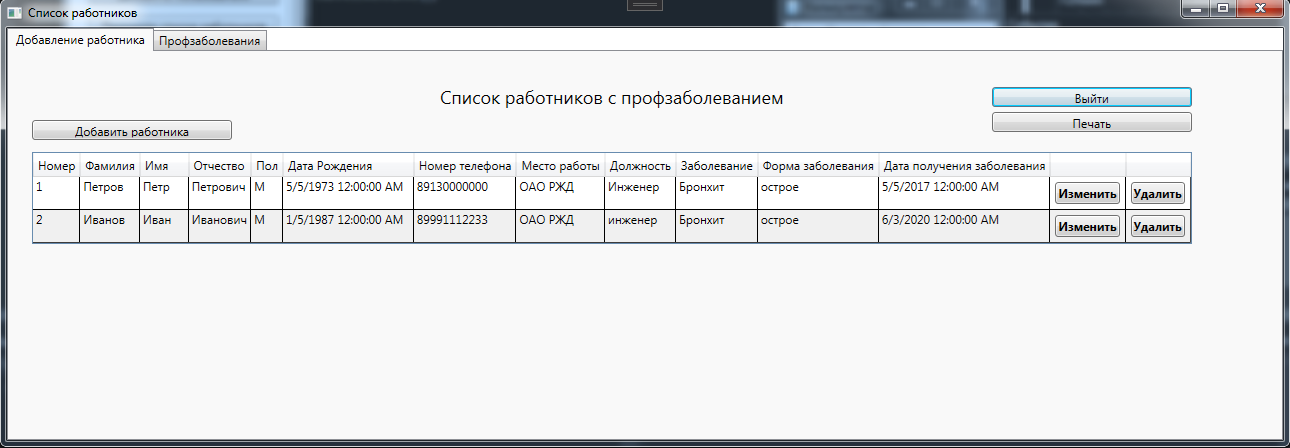


Рисунок 4.24 – Результат добавления нового работника

Если возникла необходимость изменить данные о работнике, то нужно нажать на кнопку «Изменить», после чего появится форма, где можно вносить изменения, которые требуются. После чего следует нажать на кнопку «Сохранить», после чего форма закроется и появится сообщение, что данные были успешно сохранены. Результаты продемонстрированы на рисунках 4.25 – 4.27.

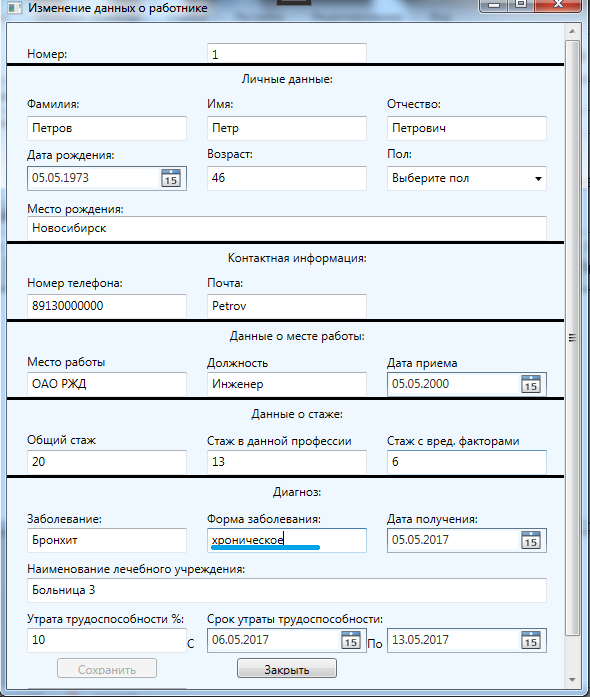


Рисунок 4.25 – Изменение данных

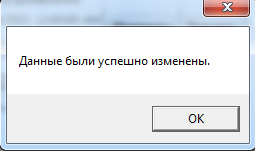


Рисунок 4.26 – Сообщение об изменении данных

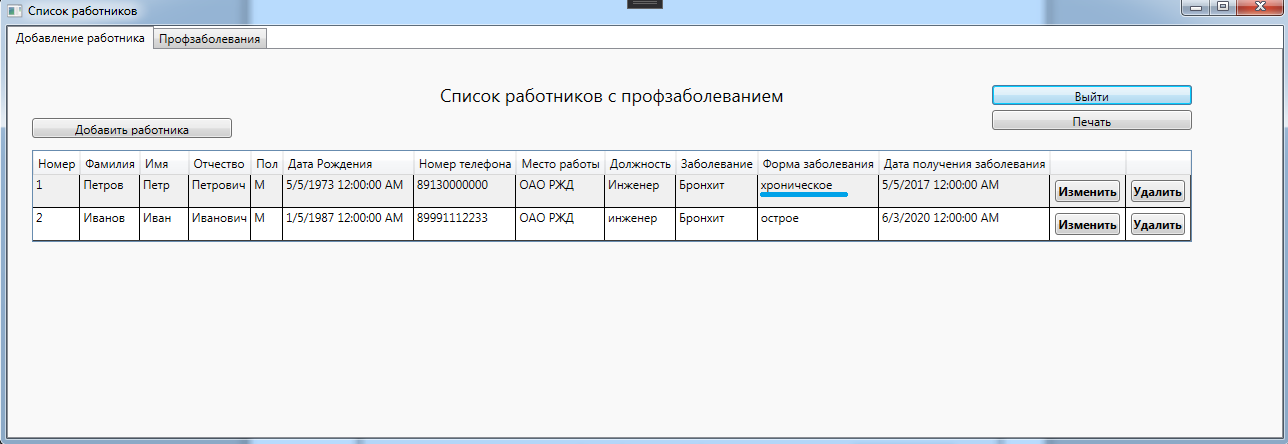


Рисунок 4.27 – Результат изменения данных

Так же имеется возможность выгрузки данных из таблицы о работниках имеющих профессиональные заболевания. Для этого пользователю системы необходимо нажать на кнопку «Печать», после чего данные из таблицы выгрузятся в таблицу в формате Excel, где пользователь так же может вносить изменения. Результат выгрузки данных из таблицы представлен на рисунке 4.28.

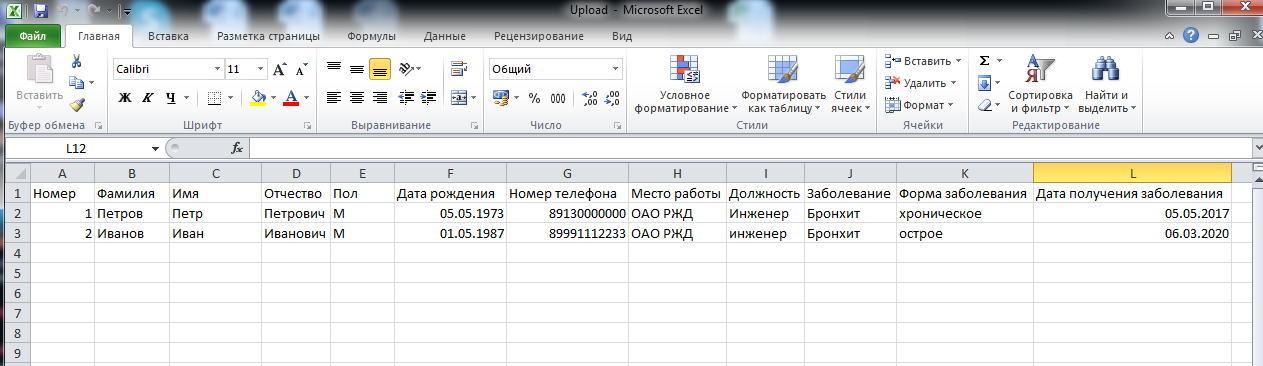


Рисунок 4.28 – Скаченная таблица в формате Excel

# ЗАКЛЮЧЕНИЕ

Данная выпускная квалификационная работа была направлена на создание информационной системы учета профессиональных заболеваний работников ОАО «РЖД».

В ходе выполнения работы было произведено аналитическое исследование, моделирование бизнес-процессов и разработка информационной системы учета профессиональных заболеваний работников ОАО «РЖД».

Были исследованы и изучены существующие информационные системы, а так же был описан бизнес-процесс для существующей системы.

Разработанная информационная система отвечает всем требованиям, предъявляемые на этапе постановки задачи. В ходе разработки была спроектирована рабочая база данных и создан интерфейс программы.

В работе произведен анализ предметной области, подробно изучены бизнес-процессы работы пользователей, при помощи языка UML были построены соответствующие диаграммы бизнес-процессов, описан функционал разработанной системы и был описан интерфейс пользователя.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Положение о расследовании и учете профессиональных заболеваний [Электронный ресурс]. – Режим доступа: https://base.garant.ru/182775/ (дата обращения 08.05.2022).
2. Информационная система [Электронный ресурс]. – Режим доступа: https://dic.academic.ru/dic.nsf/ruwiki/6014 (дата обращения 10.05.2022).
3. Вейцман В.М. Автоматизированная разработка корпоративных информационных систем: Учебное пособие [Текст] / В.М. Вейцман. – Ярославль: МУБиНТ, 2018. – 290 с.
4. Истоки возникновения UML [Электронный ресурс]. – Режим доступа: https://www.lucidchart.com/pages/ru/uml (дата обращения 12.05.2022).
5. UML Use Case Diagrams [Электронный ресурс]. – Режим доступа: https://www.uml-diagrams.org/use-case-diagrams.html (дата обращения 18.05.2022).
6. Теория и практика UML. Диаграмма последовательности [Электронный ресурс]. – Режим доступа: http://www.it-gost.ru/articles/view\_articles/94 (дата обращения 15.05.2022).
7. Евентьев, А.В. Создание и ведение базы данных для автоматизации управления в предметной области / А.В. Евентьев. - Москва: Лаборатория книги, 2018. – С.-117.
8. Архитектура SQL Server[Электронный ресурс]. – Режим доступа: http://citforum.ru/ofis/backoffice/glava2.shtml (дата обращения 20.05.2022).
9. Краткий обзор языка C# [Электронный ресурс]. – Режим доступа: URL: https://docs.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/ (дата обращения 25.05.2022).
10. Microsoft Visual Studio [Электронный ресурс]. – Режим доступа: URL: https://ru.wikipedia.org/wiki/Microsoft\_Visual\_Studio (дата обращения 27.05.2022).