федеральное государственное бюджетное образовательное учреждение высшего образования

«**Вологодский государственный университет**»

университетский колледж

Информационные системы и программирование

(наименование специальности)

**КУРСОВОЙ ПРОЕКТ**

Дисциплина: «МДК 01.01 Разработка программных модулей»

Наименование темы: «Разработка модуля информационной системы почтовой службы»

Код работы КП 09.02.07.13.00.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Код специальности, код УК, регистрационный номер по журналу, год

Руководитель Преподаватель Дегтярев Михаил Евгеньевич

(уч. степень, звание, должность. Ф.И.О)

Выполнил (а) обучающийся Дронтусов Виктор Евгеньевич

(Ф.И.О)

Группа, курс ИСП 31–20, 3 курс

Дата сдачи \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка по защите \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись преподавателя)

Вологда

2023 г.

федеральное государственное бюджетное образовательное учреждение высшего образования

«**Вологодский государственный университет**»

университетский колледж

**Задание на курсовой проект**

По МДК 01.01 Разработка программных модулей

Обучающемуся Дронтусову Виктору Евгеньевичу

Специальности 09.02.07 Информационные системы и программирование

Тема: Разработка модуля информационной системы почтовой службы

Исходные данные к курсовому проекту: Интегрированная среда разработки Microsoft Visual Studio Community 2022, СУБД Microsoft SQL Server

Содержание и объём курсового проекта:

Введение

Аналитический обзор

Разработка программного обесечения

Тестирование

Руководство пользователя

Заключение

Список использованных источников

Приложения

Председатель П(Ц)К\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Руководитель курсового проекта\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Обучающийся\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Дата выдачи задания«\_\_»\_\_\_20\_\_г.

Срок выполнения«\_\_»\_\_\_20\_\_г.

Курсовой проект защищен с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

(Подпись руководителя)

Содержание

Введение 4

1. аналитический обзор 5
   1. Описание требований 5
   2. Выбор средств разработки 5
   3. Описание базы данных 9
2. разработка программного обеспечения 12
   1. Разработка графического интерфейса 12
   2. Разработка программного функционала 41
3. тестирование 52
   1. Тестирование логики программы 52
   2. Тестирование интерфейса программы 52
4. руководство пользователя 53
   1. Введение 53
   2. Назначение и условия применения программного обеспечения 53
   3. Подготовка к работе 54
   4. Описание операций 54

заключение 63

список использованных источников 64

приложение 1 65

приложение 2 73

приложение 3 77

ВВЕДЕНИЕ

Актуальность курсовой работы заключается в том, что современные почтовые службы сталкиваются с огромным потоком информации, связанным с доставкой почты, и не всегда могут справиться с ним эффективно. Приложение, разработанное для управления этими данными, может помочь значительно снизить нагрузку на сотрудников и улучшить общую эффективность работы службы.

Приложение для почтовой службы, облегчит доступ к данным и поможет просто и эффективно просматривать, редактировать и удалять данные. Вместо того, чтобы сотрудники тратили время на ручноую запись данных и поиск нужной информации, приложение позволит им быстро и легко получать доступ ко всем необходимым данным, включая информацию об отправлениях, адресах и получателях.

Кроме того, разработка приложения для почтовой службы может помочь улучшить качество и скорость управления доставкой, а также повысить удобство использования почтовых услуг для пользователей.

Целью курсового проекта является разработка модуля информационной системы почтовой службы.

Задачи на проект:

* описание требований;
* выбор средств разработки;
* разработка графического интерфейса;
* разработка программного функционала;
* тестирование;
* написание руководство пользователя.

1. АНАЛИТИЧЕСКИЙ ОБЗОР
   1. Описание требований

Приложение имеет функционал, позволяющий работать с данными различным пользователям. Для этого в приложении предусмотрена система разграничения прав доступа пользователей.

Одной из ключевых функций приложения является эффективная работа с заказами, включающая возможность просмотра, добавления, редактирования и удаления заказов. Так же приложение обеспечивает удобный и интуитивно понятный интерфейс для пользователя, позволяющий легко и быстро освоиться с функционалом и получить доступ к необходимой информации.

Подробные требования к программному обеспечению находятся в приложении 1.

* 1. Выбор средств разработки

Сначала необходимо выбрать язык, на котором будет происходить разработка. Из популярных языков можно выделить: C#, C++, Python, Java.

C# - это объектно-ориентированный язык программирования, разработанный компанией Microsoft. Он широко используется для создания настольных приложений для операционных систем Windows. Основными плюсами C# являются: большое количество инструментов и библиотек, которые могут упростить и ускорить процесс разработки, статическая типизация, которая позволяет выявить ошибки на этапе компиляции, система сборки мусора, которая автоматически освобождает память, используемую приложением, что уменьшает вероятность утечек памяти.

Однако C# также имеет некоторые минусы: в отличие от некоторых других языков программирования, например, C и C++, C# не позволяет использовать непосредственно указатели и работать с памятью напрямую, что может существенно ограничивать возможности разработчика в некоторых случаях, привязка к платформе .NET, означает, что при разработке приложений на C# необходимо устанавливать и поддерживать окружение .NET на всех целевых компьютерах, что может быть неудобно в некоторых ситуациях.

C++ –более низкоуровневый, что является и минусом и плюсом одновременно, позволяет эффективно управлять памятью. Но сложнее в освоении, требует больше времени на разработку, больше возможностей для ошибок.

Python – прост в освоении, хорошо подходит для быстрого прототипирования, большое количество библиотек. Низкая производительность из-за интерпретатора, динамическая типизация так же повышает риск ошибок.

Java – имеет более низкую производительность из-за JVM. В данный момент плохо подходит для настольных приложений, так как имеет только устаревшие платформы для их создания.

Выбор был сделан в пользу языка C#, так как он является достаточно простым в освоении, сложнее допустить ошибки и хорошо подходит для использования под платформу Windows, которая как раз необходима.

Далее необходимо выбрать фреймворк или библиотеку, так как это упростит разработку и позволит создать более качественное приложение.

WPF – это технология, которая предоставляет богатые возможности для создания современных и привлекательных пользовательских интерфейсов. WPF основывается на языке программирования C# и платформе .NET, что обеспечивает хорошую поддержку и интеграцию с другими продуктами Microsoft.

WPF предоставляет богатый функционал для работы с графикой, анимациями, мультимедиа, стилями и шаблонами. Он также обладает мощной системой привязки данных и гибкой архитектурой модулей, что делает его лучшим выбором для разработки современных настольных приложений под платформу Windows.

Windows Forms (WF) - это более старая технология разработки настольных приложений от Microsoft, которая также основывается на языке программирования C#. WF предоставляет более простой и прямолинейный подход к созданию пользовательских интерфейсов, что делает его хорошим выбором для быстрой разработки небольших приложений, но имеет множество ограничений.

QT - это кроссплатформенный фреймворк для разработки настольных и мобильных приложений на C++. Он обладает широким функционалом для создания пользовательских интерфейсов и поддерживает множество операционных систем, но имеет более высокую сложность написания интерфейса и могут быть проблемы с совместимостью на некоторых операционных системах.

JavaFX - это технология разработки настольных приложений на языке Java. Он предоставляет мощные возможности для создания пользовательских интерфейсов. Но является устаревшим и не востребованным, не имеет поддержки сторонних библиотек и в целом менее производителен.

PyQT – фреймворк для создания настольных приложений на Python. Он основывается на QT и позволяет создавать кроссплатформенные приложения на Python. Однако, Python не является идеальным выбором для разработки настольных приложений из-за ограничений в производительности и сложности интеграции с некоторыми платформами.

Для разработки был выбран WPF, так как это новый и достаточно мощный инструмент разработки настольных приложений. Так же имеет хорошую совместимость с Windows, что необходимо.

После выбора языка и инструмента для разработки настольного приложения, необходимо определиться с интегрированной средой разработки. Самыми популярными средами для разработки на C# и WPF являются Microsoft Visual Studio и JetBrains Rider.

Microsoft Visual Studio предоставляет множество инструментов и функций для разработки приложений, таких как поддержка IntelliSense, встроенный отладчик, систему управления версиями, графический дизайнер и т.д.

Кроме того, Microsoft Visual Studio обладает лучшей интеграцией с другими продуктами и технологиями от Microsoft. Это позволяет разработчикам более эффективно работать с различными платформами и сервисами Microsoft.

Несмотря на то, что JetBrains Rider имеет свои преимущества, такие как высокая производительность и кросс-платформенность, он уступает Microsoft Visual Studio в общем функционале и интеграции с другими продуктами от Microsoft. Поэтому, выбрана интегрированная среда разработки Microsoft Visual Studio.

Следующим шагом является выбор средств управления баз данных (СУБД). Можно выделить несколько популярных СУБД: MS SQL Server, MySQL, Oracle.

MS SQL Server - это коммерческая СУБД, разработанная Microsoft. Она широко используется по всему миру. MS SQL Server предлагает множество функций и возможностей, включая интеграцию с .NET. Кроме того, MS SQL Server обеспечивает высокую производительность и масштабируемость.

MySQL - это бесплатная и открытая СУБД, которая широко используется в веб-разработке и веб-хостинге. Она имеет свободную лицензию и доступна для загрузки из интернета. MySQL имеет некоторые возможности, включая транзакции и поддержку хранимых процедур. Однако, она может быть менее масштабируемой и менее надежной, чем MS SQL Server.

Oracle - это коммерческая СУБД, которая широко используется в крупных предприятиях. Oracle предлагает высокую масштабируемость и надежность. Однако, Oracle может быть дороже и сложнее в использовании, чем MS SQL Server.

Выбор сделан в пользу в пользу Microsoft SQL Server, так как данная СУБД предлагает большие возможности, простоту использования и хорошую производительность.

* 1. Описание базы данных

В базе данных содержатся таблицы:

* addresses – адреса
* cities – города
* countries – страны
* delivery – доставки(посылки)
* delivery\_history – история доставок
* houses – дома
* pick\_up\_points – точки выдачи
* positions – должности
* reviews – отзывы
* shifts – рабочие смены
* storages – склады
* streets -улицы
* tariffs – тарифы доставки
* users – пользователи
* worker\_images – изображения сотрудников
* workers – сотрудники
* workers\_in\_delivery – сотрудники, участвующие в доставке
* workers\_in\_pick\_up\_points – сотрудники в точках выдачи
* workers\_in\_storages – сотрудники на складах

Таблицы addresses, countries, cities, streets, houses вместе являются полным адресом, состоит из названия страны, города, улицы, номера дома и почтового индекса. Адреса используются в различных таблицах базы данных таких, как users, deliveries, storages и другие.

Для представления пользователя в базе данных создана таблица users которая, содержит ФИО, номер телефона, номер и серию паспорта, адрес проживания и адрес по паспорту. Используется для формирования доставки, может является получателем или отправителем.

Таблицы workers, worker\_images, positions служат для отображения информации о работниках. В них входит информация о сотруднике, как о пользователе, благодаря связи один к одному, должность, логин, пароль и фотография сотрудника.

Таблица shifts содержит данные о начале и окончании рабочей смены. Используется в промежуточных таблицах workers\_in\_pick\_up\_points и workers\_in\_storages.

Таблица pick\_up\_points показывает точки выдачи и их адреса. Может использоваться, как точка выдачи и как точка принятия посылок. Для данной таблицы существует промежуточная workers\_in\_pick\_up\_points, которая позволяет создать связь многие ко многим между workers и pick\_up\_points. Помимо этого в промежуточной таблице хранится смена сотрудника, в которую он работает.

Таблицы storages отображает информацию о складе, а именно его номер и адрес. Для данной таблицы создана промежуточная workers\_in\_storages, которая создаёт связь между workers и storages. В промежуточной таблице используется смена сотрудника.

Доставки в базе данных отображены в таблице delivery. Она содержит в себе информацию о тарифе, цене, отправителе, получателе, точки принятия посылки и точки выдачи. Так же есть таблица с историей доставок, которая называется delivery\_history и содержит идентификатор доставки, время, статус и адрес. Для возможности сохранять отзывы о доставке имеется таблица reviews, которая содержит оценку, комментарий и идентификатор доставки.

ER-диаграмма базы данных представлена в соответствии с рисунком 1.

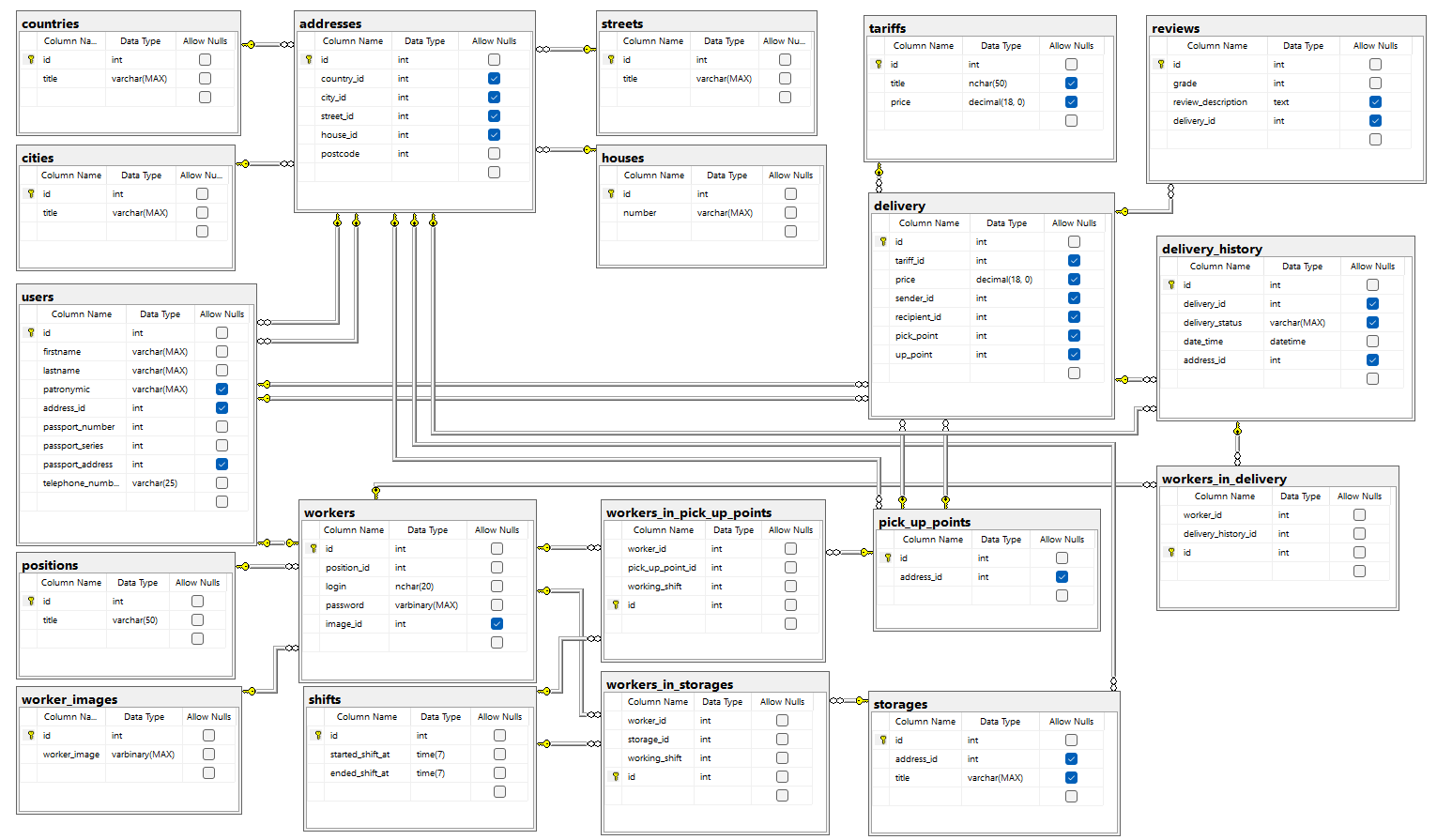


Рисунок 1 – ER-диаграмма.

1. РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
   1. Разработка графического интерфейса

При разработке приложения сначала было необходимо написать стили для различных элементов. Они предназначены для уменьшения повторений в коде и большей его читаемости. Стили написаны для следующих элементов: кнопок, выпадающих списков, таблиц данных, иконок, полей ввода, текстовых полей, окна. Для рассмотрения выбран пример со стилем кнопки управления окном – ControlButton.

*<Style x:Key="ControlButton" TargetType="Button">*

*<Setter Property="Width" Value="18"/>*

*<Setter Property="Height" Value="18"/>*

*<Setter Property="FocusVisualStyle" Value="{x:Null}"/>*

*<Setter Property="Margin" Value="5 0 5 0"/>*

*<Setter Property="WindowChrome.IsHitTestVisibleInChrome" Value="True" />*

*<Setter Property="Template">*

*<Setter.Value>*

*<ControlTemplate TargetType="Button">*

*<Border Background="{TemplateBinding Background}"*

*CornerRadius="10">*

*<ContentPresenter HorizontalAlignment="Center"*

*VerticalAlignment="Center"/>*

*</Border>*

*</ControlTemplate>*

*</Setter.Value>*

*</Setter>*

*</Style>*

Подключение стилей происходит в файле App.xaml.

*<Application x:Class="DeliveryService.App"*

*xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"*

*xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml">*

*<Application.Resources>*

*<ResourceDictionary>*

*<ResourceDictionary.MergedDictionaries>*

*<ResourceDictionary Source="\Resources\Styles\Colors.xaml" />*

*<ResourceDictionary Source="\Resources\Styles\Buttons.xaml" />*

*<ResourceDictionary Source="\Resources\Styles\Texts.xaml" />*

*<ResourceDictionary Source="\Resources\Styles\TextBoxes.xaml" />*

*<ResourceDictionary Source="\Resources\Styles\ComboBoxes.xaml" />*

*<ResourceDictionary Source="\Resources\Styles\DataGrids.xaml" />*

*<ResourceDictionary Source="\Resources\Styles\Icons.xaml" />*

*<ResourceDictionary Source="\Resources\Styles\Windows.xaml" />*

*</ResourceDictionary.MergedDictionaries>*

*</ResourceDictionary>*

*</Application.Resources>*

*</Application>*

Первым окном, которое видит пользователь является окно входа в систему, в соответствии с рисунком 2. Оно содержит подписанные поля ввода логина, пароля и кнопку «Войти».

*<Window x:Class="DeliveryService.View.LoginWindow"*

*xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"*

*xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"*

*xmlns:d="http://schemas.microsoft.com/expression/blend/2008"*

*xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"*

*xmlns:icons="http://metro.mahapps.com/winfx/xaml/iconpacks"*

*mc:Ignorable="d"*

*xmlns:converters="clr-namespace:DeliveryService.Converters"*

*Title="Вход в систему" Height="350" Width="600"*

*ResizeMode="NoResize"*

*Style="{StaticResource WinChrome}">*

*<Grid>*

*<Grid.RowDefinitions>*

*<RowDefinition Height="30"/>*

*<RowDefinition Height="\*"/>*

*<RowDefinition Height="Auto"/>*

*<RowDefinition Height="Auto"/>*

*<RowDefinition Height="Auto"/>*

*<RowDefinition Height="\*"/>*

*</Grid.RowDefinitions>*

*<!--Control Panel-->*

*<StackPanel Orientation="Horizontal"*

*VerticalAlignment="Center"*

*Margin="10 5"*

*FlowDirection="RightToLeft">*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlCloseColorBrush}"*

*Command="{Binding CloseWindowCommand}">*

*<icons:PackIconMaterial Kind="Close"*

*Width="10"*

*Height="10"/>*

*</Button>*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlMinColorBrush}"*

*Command="{Binding MinimizeWindowCommand}">*

*<icons:PackIconMaterial Kind="Minus"*

*Width="10"*

*Height="10"/>*

*</Button>*

*</StackPanel>*

*<StackPanel Grid.Row="2"*

*HorizontalAlignment="Center">*

*<TextBlock Text="Логин"*

*Margin="10 2"*

*Style="{StaticResource EditField}"/>*

*<TextBox Width="200"*

*Style="{StaticResource TextBoxSearch}"*

*Text="{Binding Login}" />*

*</StackPanel>*

*<StackPanel Grid.Row="3"*

*HorizontalAlignment="Center">*

*<TextBlock Text="Пароль"*

*Margin="10 2"*

*Style="{StaticResource EditField}"/>*

*<PasswordBox x:Name="PbLogin"*

*Width="200"*

*Style="{StaticResource PasswordBoxEdit}"/>*

*</StackPanel>*

*<StackPanel Grid.Row="4">*

*<TextBlock Text="\* Не верный логин или пароль"*

*Style="{StaticResource EditField}"*

*HorizontalAlignment="Center"*

*Margin="0 5 0 0"*

*Visibility="{Binding ErrorMessageVisibility, Converter={converters:VisibilityBoolConverter}}"/>*

*<Button Content="Войти"*

*Style="{StaticResource EditButtons}"*

*Margin="0 10"*

*Command="{Binding LoginCommand}"*

*CommandParameter="{Binding ElementName=PbLogin}"*

*Cursor="Hand"/>*

*</StackPanel>*

*</Grid>*

*</Window>*

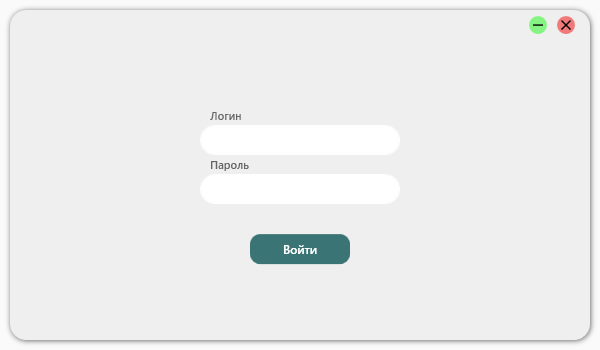


Рисунок 2 – Окно входа в систему.

Следующим окном после успешного входа в систему является главное окно, в соответствии с рисунком 3. Оно содержит все основные компоненты навигации по приложению, а так же имя, должность и изображение сотрудника, который вошёл в систему. Разделы не доступные сотруднику при выборе не будут отображаться, вместо них надпись «Выберите раздел».

*<Window x:Class="DeliveryService.View.MainWindow"*

*xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"*

*xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"*

*xmlns:d="http://schemas.microsoft.com/expression/blend/2008"*

*xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"*

*xmlns:converters="clr-namespace:DeliveryService.Converters"*

*xmlns:icons="http://metro.mahapps.com/winfx/xaml/iconpacks"*

*xmlns:pages="clr-namespace:DeliveryService.Enums"*

*mc:Ignorable="d"*

*Title="MainWindow"*

*Style="{StaticResource WinChrome}"*

*MaxHeight="{Binding MaxHight}"*

*MinHeight="650"*

*MinWidth="1200">*

*<Grid>*

*<Grid.ColumnDefinitions>*

*<ColumnDefinition Width="200"/>*

*<ColumnDefinition/>*

*</Grid.ColumnDefinitions>*

*<Grid.RowDefinitions>*

*<RowDefinition Height="30"/>*

*<RowDefinition/>*

*</Grid.RowDefinitions>*

*<!--Left Menu-->*

*<Grid Grid.Row="1"*

*Grid.Column="0">*

*<Grid.RowDefinitions>*

*<RowDefinition Height="auto"/>*

*<RowDefinition/>*

*</Grid.RowDefinitions>*

*<Border Grid.Row="1"*

*Margin="0 100 0 0 "*

*Background="{StaticResource PrimaryColorBrush}"*

*CornerRadius="0 50 0 17">*

*<DockPanel>*

*<!--Avatar-->*

*<Border Width="80"*

*Height="80"*

*Background="White"*

*CornerRadius="100"*

*DockPanel.Dock="Top"*

*HorizontalAlignment="Left"*

*Margin="20 -30 0 0"*

*Padding="3">*

*<Ellipse>*

*<Ellipse.Fill>*

*<ImageBrush ImageSource="{Binding CurrentWorker.Image.WorkerImage1}"/>*

*</Ellipse.Fill>*

*</Ellipse>*

*</Border>*

*<!--FirstName-->*

*<TextBlock Text="{Binding CurrentWorker.IdNavigation.Firstname}"*

*Foreground="White"*

*FontSize="22"*

*FontWeight="SemiBold"*

*DockPanel.Dock="Top"*

*HorizontalAlignment="Left"*

*Margin="45 15 0 5"/>*

*<!--Position-->*

*<TextBlock Text="{Binding CurrentWorker.Position.Title}"*

*Foreground="{StaticResource TextWhiteFadedColorBrush}"*

*FontSize="16"*

*FontWeight="SemiBold"*

*DockPanel.Dock="Top"*

*HorizontalAlignment="Left"*

*Margin="35 0 0 10"/>*

*<Separator DockPanel.Dock="Top"*

*Height="0.8"*

*Margin="25"*

*Background="{StaticResource TextWhiteFadedColorBrush}"/>*

*<StackPanel DockPanel.Dock="Top">*

*<RadioButton Style="{StaticResource MenuSectionButton}"*

*Command="{Binding ChangePageCommand}"*

*CommandParameter="{x:Static pages:AppPages.Worker}">*

*<StackPanel Orientation="Horizontal">*

*<icons:PackIconMaterial Kind="AccountDetails"*

*Style="{StaticResource MenuSectionButtonIcon}"/>*

*<TextBlock Text="Сотрудники"/>*

*</StackPanel>*

*</RadioButton>*

*<RadioButton Style="{StaticResource MenuSectionButton}"*

*Command="{Binding ChangePageCommand}"*

*CommandParameter="{x:Static pages:AppPages.User}">*

*<StackPanel Orientation="Horizontal">*

*<icons:PackIconMaterial Kind="Account"*

*Style="{StaticResource MenuSectionButtonIcon}"/>*

*<TextBlock Text="Пользователи"/>*

*</StackPanel>*

*</RadioButton>*

*<RadioButton Style="{StaticResource MenuSectionButton}"*

*Command="{Binding ChangePageCommand}"*

*CommandParameter="{x:Static pages:AppPages.Delivery}">*

*<StackPanel Orientation="Horizontal">*

*<icons:PackIconMaterial Kind="TruckDelivery"*

*Style="{StaticResource MenuSectionButtonIcon}"/>*

*<TextBlock Text="Доставки"/>*

*</StackPanel>*

*</RadioButton>*

*<RadioButton Style="{StaticResource MenuSectionButton}"*

*Command="{Binding ChangePageCommand}"*

*CommandParameter="{x:Static pages:AppPages.Storage}">*

*<StackPanel Orientation="Horizontal">*

*<icons:PackIconMaterial Kind="Store24Hour"*

*Style="{StaticResource MenuSectionButtonIcon}"/>*

*<TextBlock Text="Склады"/>*

*</StackPanel>*

*</RadioButton>*

*<RadioButton Style="{StaticResource MenuSectionButton}"*

*Command="{Binding ChangePageCommand}"*

*CommandParameter="{x:Static pages:AppPages.PickUpPoint}">*

*<StackPanel Orientation="Horizontal">*

*<icons:PackIconMaterial Kind="StoreMarker"*

*Style="{StaticResource MenuSectionButtonIcon}"/>*

*<TextBlock Text="Точки выдачи"/>*

*</StackPanel>*

*</RadioButton>*

*</StackPanel>*

*</DockPanel>*

*</Border>*

*</Grid>*

*<!--Control Panel-->*

*<Grid Grid.Row="0"*

*Grid.ColumnSpan="2">*

*<DockPanel x:Name="spControlPanel"*

*Background="Transparent"*

*Margin="5 0 10 0">*

*<TextBlock Text="Delivery Service"*

*Margin="5 0 0 0"*

*VerticalAlignment="Center"/>*

*<StackPanel Orientation="Horizontal"*

*FlowDirection="RightToLeft">*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlCloseColorBrush}"*

*Command="{Binding CloseWindowCommand}">*

*<icons:PackIconMaterial Kind="Close"*

*Width="10"*

*Height="10"/>*

*</Button>*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlMaxColorBrush}"*

*Command="{Binding MaximazeWindowCommand}">*

*<icons:PackIconMaterial Kind="SquareRoundedOutline"*

*Width="10"*

*Height="10"/>*

*</Button>*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlMinColorBrush}"*

*Command="{Binding MinimizeWindowCommand}">*

*<icons:PackIconMaterial Kind="Minus"*

*Width="10"*

*Height="10"/>*

*</Button>*

*</StackPanel>*

*</DockPanel>*

*</Grid>*

*<Frame Grid.Row="1"*

*Grid.Column="1"*

*Content="{Binding CurrentPage, Converter={converters:AppPageConverter}}"*

*NavigationUIVisibility="Hidden"*

*Margin="0 0 5 10"/>*

*</Grid>*

*</Window>*

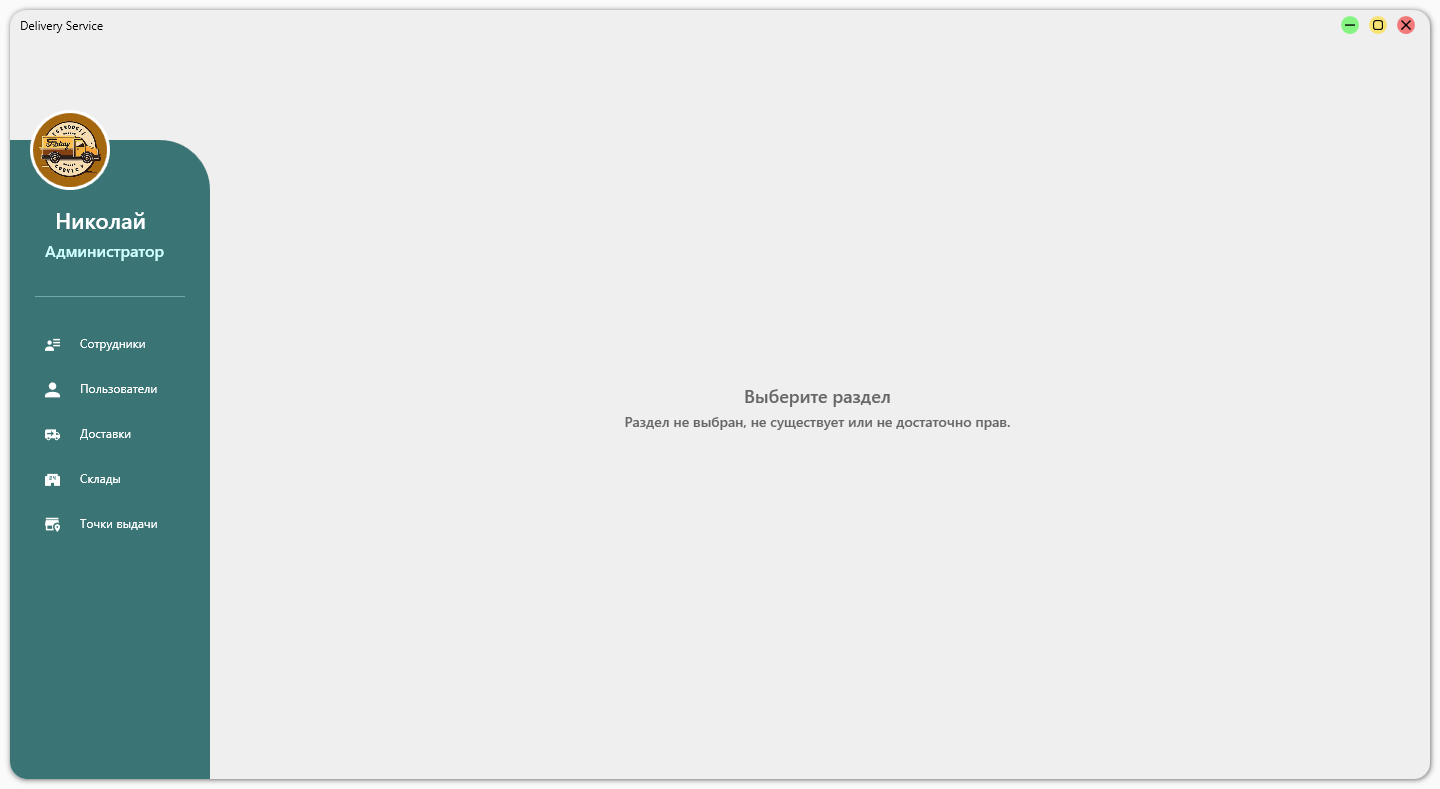


Рисунок 3 – Главное окно приложения.

Первый раздел «Сотрудники» содержит два подраздела «Основные сведения» и «Пароли», в соответствии с рисунком 4. Сам раздел предоставляет выбор одного из подразделов и не содержит в себе ничего более. По умолчанию при переходе в данный раздел выбирается подраздел «Основные сведения».

*<Page x:Class="DeliveryService.View.WorkerPage"*

*xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"*

*xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"*

*xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"*

*xmlns:d="http://schemas.microsoft.com/expression/blend/2008"*

*xmlns:converters="clr-namespace:DeliveryService.Converters"*

*xmlns:pages="clr-namespace:DeliveryService.Enums"*

*mc:Ignorable="d"*

*d:DesignHeight="620" d:DesignWidth="1000"*

*Title="WorkerPage">*

*<Border BorderBrush="Transparent">*

*<Grid>*

*<Grid.RowDefinitions>*

*<RowDefinition Height="150"/>*

*<RowDefinition/>*

*</Grid.RowDefinitions>*

*<!--Top Section-->*

*<Grid Margin="20 0 15 0">*

*<Grid.ColumnDefinitions>*

*<ColumnDefinition/>*

*<ColumnDefinition/>*

*</Grid.ColumnDefinitions>*

*<Grid.RowDefinitions>*

*<RowDefinition/>*

*<RowDefinition/>*

*</Grid.RowDefinitions>*

*<TextBlock Text="Сотрудники"*

*Margin="25 0 0 0"*

*Style="{StaticResource SelectionTitle}"/>*

*<StackPanel Grid.Row="1" Grid.Column="0"*

*Orientation="Horizontal"*

*<RadioButton GroupName="SubSelection"*

*Content="Общие сведения"*

*IsChecked="True"*

*Style="{StaticResource SubSectionButton}"*

*Command="{Binding ChangePageCommand}"*

*CommandParameter="{x:Static pages:WorkerPages.WorkerGeneralInfo}"/>*

*<RadioButton GroupName="SubSelection"*

*Content="Пароли"*

*Style="{StaticResource SubSectionButton}"*

*Command="{Binding ChangePageCommand}"*

*CommandParameter="{x:Static pages:WorkerPages.WorkerPassword}"/>*

*</StackPanel>*

*</Grid>*

*<Frame Grid.Row="1"*

*Content="{Binding CurrentPage, Converter={converters:WorkerPageConverter}}"*

*Margin="20 0 10 0"*

*NavigationUIVisibility="Hidden"/>*

*</Grid>*

*</Border>*

*</Page>*

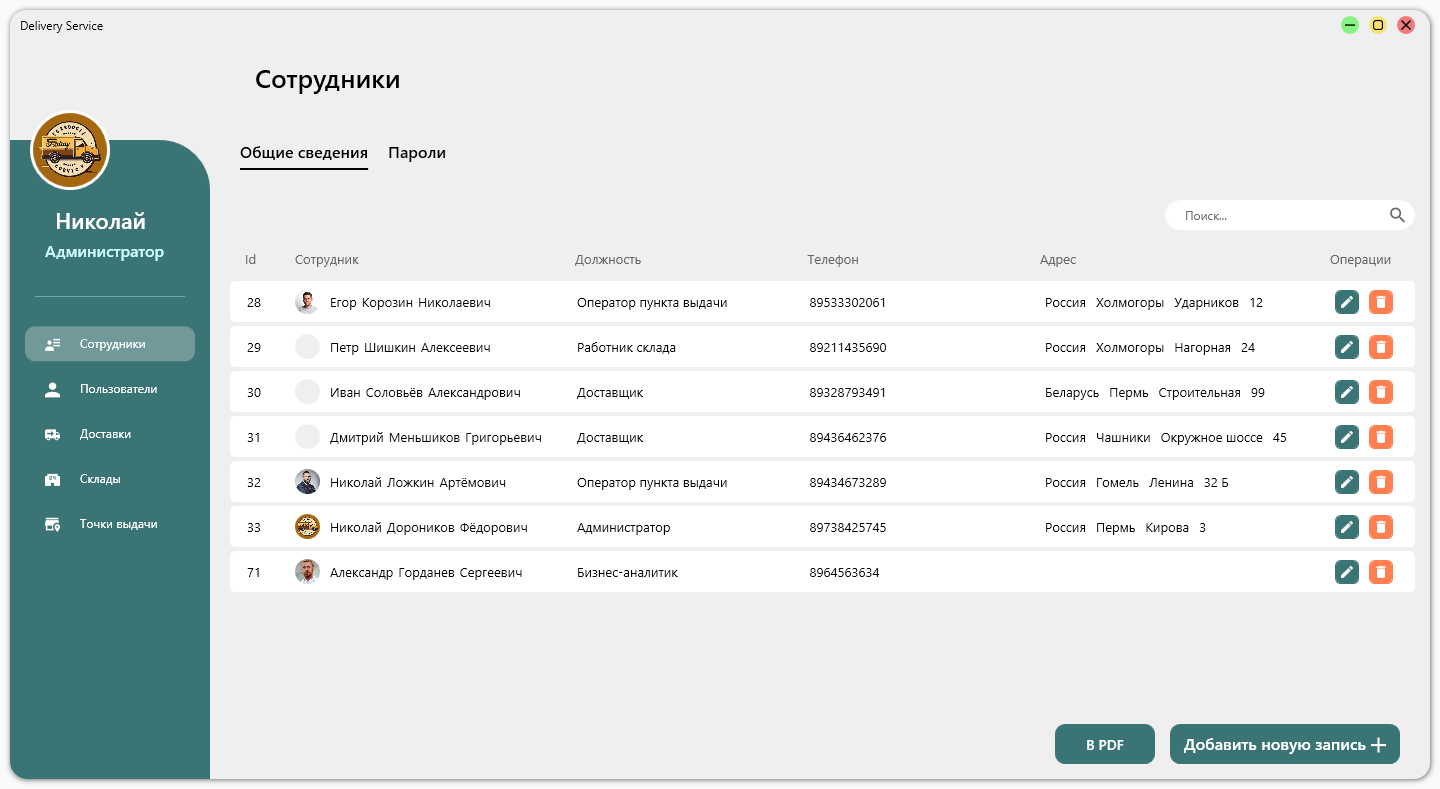


Рисунок 4 – Раздел «Сотрудники» подраздел «Общие сведения».

Подраздел «Основные сведения» радела «Сотрудники», в соответствии с рисунком 5. Он состоит из таблицы, в которой указаны id, изображение и ФИО, должность, телефон, адрес сотрудника, а также имеется возможность изменять, удалять и добавлять сотрудников соответствующими кнопками. Для облегчения поиска необходимой записи имеется поиск, расположенный слева-сверху от таблицы. Таблицу можно выгрузить в PDF файл по кнопке «В PDF».

*<Page x:Class="DeliveryService.View.Workers.WorkerGeneralInfoPage"*

*xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"*

*xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"*

*xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"*

*xmlns:d="http://schemas.microsoft.com/expression/blend/2008"*

*xmlns:icons="http://metro.mahapps.com/winfx/xaml/iconpacks"*

*xmlns:converters="clr-namespace:DeliveryService.Converters"*

*mc:Ignorable="d"*

*d:DesignHeight="450" d:DesignWidth="800"*

*Title="WorkerGeneralInfoPage">*

*<Grid>*

*<Grid.RowDefinitions>*

*<RowDefinition Height="50" />*

*<RowDefinition />*

*<RowDefinition Height="60" />*

*</Grid.RowDefinitions>*

*<!--Top Menu-->*

*<Grid>*

*<StackPanel Orientation="Horizontal" />*

*<StackPanel Orientation="Horizontal"*

*FlowDirection="RightToLeft">*

*<!--Search TextBox-->*

*<Grid Width="250"*

*FlowDirection="LeftToRight">*

*<TextBlock Margin="20 0"*

*Foreground="{StaticResource TextGrayColorBrush}"*

*VerticalAlignment="Center"*

*IsHitTestVisible="False"*

*Panel.ZIndex="1"*

*Text="Поиск..."*

*Visibility="{Binding ElementName=TbSearch, Path=Text.IsEmpty, Converter={converters:VisibilityBoolConverter}}" />*

*<TextBox x:Name="TbSearch"*

*Height="30"*

*Style="{StaticResource TextBoxSearch}"*

*Text="{Binding TextBoxSearch, UpdateSourceTrigger=PropertyChanged}" />*

*<icons:PackIconMaterial Kind="Magnify"*

*Width="15"*

*Height="15"*

*VerticalAlignment="Center"*

*HorizontalAlignment="Right"*

*Margin="10 0"*

*Foreground="{StaticResource TextGrayColorBrush}" />*

*</Grid>*

*</StackPanel>*

*</Grid>*

*<DataGrid Style="{DynamicResource DataGridModernStyle}"*

*CellStyle="{DynamicResource DataGridModernCellStyle}"*

*ColumnHeaderStyle="{DynamicResource DataGridModernColumnHeaderStyle}"*

*RowStyle="{DynamicResource DataGridModernRowStyle}"*

*Grid.Row="1"*

*ItemsSource="{Binding Workers}">*

*<DataGrid.Columns>*

*<DataGridTextColumn Header="Id"*

*IsReadOnly="True"*

*MinWidth="50"*

*Width="50"*

*Binding="{Binding Id}" />*

*<DataGridTemplateColumn Header="Сотрудник"*

*IsReadOnly="True"*

*MinWidth="280">*

*<DataGridTemplateColumn.CellTemplate>*

*<DataTemplate>*

*<StackPanel Orientation="Horizontal">*

*<Border Width="25"*

*Height="25"*

*CornerRadius="50"*

*Margin="0 0 10 0"*

*VerticalAlignment="Center"*

*Background="{StaticResource BackgroundLightBrush}">*

*<Ellipse>*

*<Ellipse.Fill>*

*<ImageBrush ImageSource="{Binding WorkerImage}" />*

*</Ellipse.Fill>*

*</Ellipse>*

*</Border>*

*<TextBlock Text="{Binding FirstName}"*

*VerticalAlignment="Center"*

*Margin="0 0 5 0" />*

*<TextBlock Text="{Binding LastName}"*

*VerticalAlignment="Center"*

*Margin="0 0 5 0" />*

*<TextBlock Text="{Binding Patronymic}"*

*VerticalAlignment="Center"*

*Margin="0 0 5 0" />*

*</StackPanel>*

*</DataTemplate>*

*</DataGridTemplateColumn.CellTemplate>*

*</DataGridTemplateColumn>*

*<DataGridTextColumn Header="Должность"*

*IsReadOnly="True"*

*MinWidth="140"*

*Binding="{Binding Title}" />*

*<DataGridTextColumn Header="Телефон"*

*IsReadOnly="True"*

*MinWidth="110"*

*Binding="{Binding TelephoneNumber}" />*

*<DataGridTemplateColumn Header="Адрес"*

*IsReadOnly="True"*

*MinWidth="290">*

*<DataGridTemplateColumn.CellTemplate>*

*<DataTemplate>*

*<StackPanel HorizontalAlignment="Left"*

*Orientation="Horizontal">*

*<TextBlock Text="{Binding Address.Country}"*

*Margin="5 0" />*

*<TextBlock Text="{Binding Address.City}"*

*Margin="5 0" />*

*<TextBlock Text="{Binding Address.Street}"*

*Margin="5 0" />*

*<TextBlock Text="{Binding Address.House}"*

*Margin="5 0" />*

*</StackPanel>*

*</DataTemplate>*

*</DataGridTemplateColumn.CellTemplate>*

*</DataGridTemplateColumn>*

*<DataGridTemplateColumn Header="Операции"*

*IsReadOnly="True"*

*MinWidth="100"*

*Width="100">*

*<DataGridTemplateColumn.CellTemplate>*

*<DataTemplate>*

*<StackPanel Orientation="Horizontal">*

*<Button Style="{StaticResource GridEditButton}"*

*Command="{Binding DataContext.EditWorkerCommand, RelativeSource={RelativeSource AncestorType={x:Type DataGrid}}}"*

*CommandParameter="{Binding Id}">*

*<icons:PackIconMaterial Kind="Pencil"*

*Style="{StaticResource GridOptionsIcon}" />*

*</Button>*

*<Button Style="{StaticResource GridDeleteButton}"*

*Command="{Binding DataContext.DeleteWorkerCommand, RelativeSource={RelativeSource AncestorType={x:Type DataGrid}}}"*

*CommandParameter="{Binding Id}">*

*<icons:PackIconMaterial Kind="Delete"*

*Style="{StaticResource GridOptionsIcon}" />*

*</Button>*

*</StackPanel>*

*</DataTemplate>*

*</DataGridTemplateColumn.CellTemplate>*

*</DataGridTemplateColumn>*

*</DataGrid.Columns>*

*</DataGrid>*

*<StackPanel Grid.Row="2"*

*Orientation="Horizontal"*

*FlowDirection="RightToLeft">*

*<Button Style="{StaticResource SelectionAddRowButton}"*

*HorizontalAlignment="Right"*

*VerticalAlignment="Bottom"*

*Command="{Binding AddWorkerCommand}" />*

*<Button Style="{StaticResource EditButtons}"*

*Height="40"*

*HorizontalAlignment="Right"*

*VerticalAlignment="Bottom"*

*Margin="0 5"*

*Command="{Binding CreatePdf}"*

*Content="В PDF"*

*FontSize="14" />*

*</StackPanel>*

*</Grid>*

*</Page>*

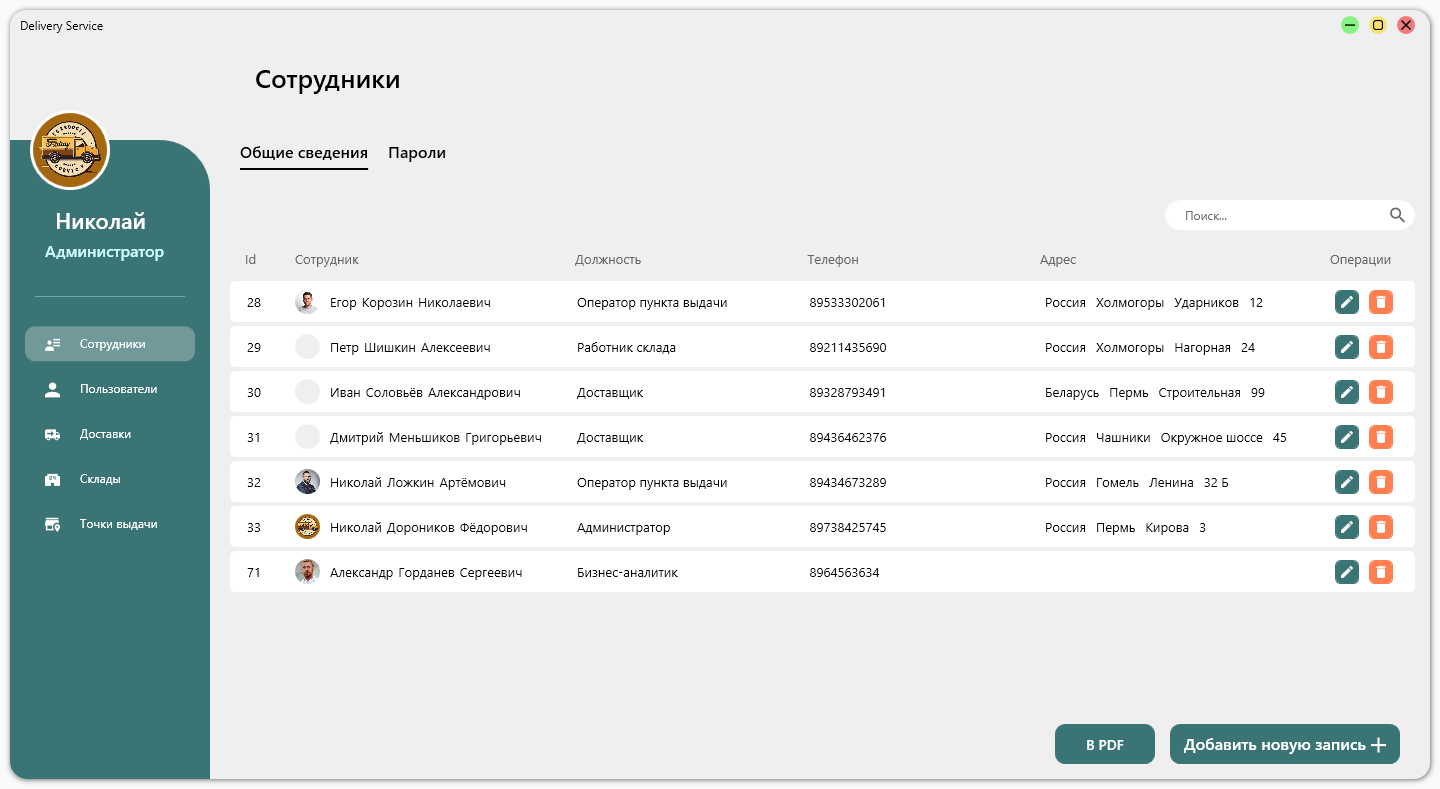


Рисунок 5 – Раздел «Сотрудники» подраздел «Общие сведения».

В подразделе «Общие сведения» раздела «Сотрудники» есть возможность добавления и редактирования записи, она реализована при помощи диалогового окна, в соответствии с рисунком 6. Окно содержит в себе поля с подписями для ввода данных, так же кнопки для изменения адреса, которые вызывают отдельные окна редактирования адреса, изменения изображения, которые вызывают диалоговое окно выбора изображения, изменения пароля, вызывающее диалоговое окно изменения пароля и выпадающий список для изменения должности сотрудника, так же внизу имеются кнопки «Сохранить» и «Отмена».

*<Window x:Class="DeliveryService.View.Workers.WorkerEdit"*

*xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"*

*xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"*

*xmlns:d="http://schemas.microsoft.com/expression/blend/2008"*

*xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"*

*xmlns:icons="http://metro.mahapps.com/winfx/xaml/iconpacks"*

*mc:Ignorable="d"*

*Title="WorkerEdit" Height="550" Width="500"*

*ResizeMode="NoResize"*

*Style="{StaticResource WinChrome}">*

*<Grid>*

*<Grid.RowDefinitions>*

*<RowDefinition Height="30"/>*

*<RowDefinition/>*

*<RowDefinition Height="60"/>*

*</Grid.RowDefinitions>*

*<!--#region Controls-->*

*<StackPanel x:Name="SpControl">*

*<StackPanel Orientation="Horizontal"*

*VerticalAlignment="Center"*

*Margin="10 8"*

*FlowDirection="RightToLeft">*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlCloseColorBrush}"*

*Command="{Binding CloseWindowCommand}">*

*<icons:PackIconMaterial Kind="Close"*

*Width="10"*

*Height="10"/>*

*</Button>*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlMinColorBrush}"*

*Command="{Binding MinimizeWindowCommand}">*

*<icons:PackIconMaterial Kind="Minus"*

*Width="10"*

*Height="10"/>*

*</Button>*

*</StackPanel>*

*</StackPanel>*

*<!--#endregion-->*

*<!--#region Fields-->*

*<StackPanel Grid.Row="1"*

*Orientation="Horizontal"*

*HorizontalAlignment="Center"*

*Margin="0 10">*

*<StackPanel HorizontalAlignment="Center"*

*Margin="15 0">*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Имя\*"*

*Style="{StaticResource EditField}"/>*

*<TextBox Width="200"*

*Style="{StaticResource TextBoxSearch}"*

*Text="{Binding Worker.FirstName}"/>*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Фамилия\*"*

*Style="{StaticResource EditField}"/>*

*<TextBox Width="200"*

*Style="{StaticResource TextBoxSearch}"*

*Text="{Binding Worker.LastName}"/>*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Отчество"*

*Style="{StaticResource EditField}"/>*

*<TextBox Width="200"*

*Style="{StaticResource TextBoxSearch}"*

*Text="{Binding Worker.Patronymic}"/>*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Должность\*"*

*Style="{StaticResource EditField}"/>*

*<ComboBox Style="{DynamicResource ComboBoxStyle1}" Width="200"*

*ItemsSource="{Binding Positions}"*

*SelectedItem="{Binding Position}">*

*<ComboBox.ItemTemplate>*

*<DataTemplate>*

*<TextBlock Text="{Binding Title}"/>*

*</DataTemplate>*

*</ComboBox.ItemTemplate>*

*</ComboBox>*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Номер телефона\*"*

*Style="{StaticResource EditField}"/>*

*<TextBox Width="200"*

*Style="{StaticResource TextBoxSearch}"*

*Text="{Binding Worker.TelephoneNumber}"/>*

*</StackPanel>*

*</StackPanel>*

*<StackPanel HorizontalAlignment="Center"*

*Margin="15 0">*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Логин\*"*

*Style="{StaticResource EditField}"/>*

*<TextBox Width="200"*

*Style="{StaticResource TextBoxSearch}"*

*Text="{Binding Worker.Login}"/>*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Номер паспорта\*"*

*Style="{StaticResource EditField}"/>*

*<TextBox Width="200"*

*Style="{StaticResource TextBoxSearch}"*

*Text="{Binding Worker.PassportNumber}"/>*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Серия паспорта\*"*

*Style="{StaticResource EditField}"/>*

*<TextBox Width="200"*

*Style="{StaticResource TextBoxSearch}"*

*Text="{Binding Worker.PassportSeries}"/>*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Адрес по паспорту"*

*Style="{StaticResource EditField}"/>*

*<Button Height="30"*

*Width="200"*

*Style="{StaticResource ButtonAddressEdit}"*

*Command="{Binding EditPassportAddressCommand}">*

*<StackPanel Orientation="Horizontal">*

*<TextBlock Text="{Binding Worker.PassportAddress.Country}"*

*Margin="5 0 0 0"/>*

*<TextBlock Text="{Binding Worker.PassportAddress.City}"*

*Margin="5 0 0 0"/>*

*<TextBlock Text="{Binding Worker.PassportAddress.Street}"*

*Margin="5 0 0 0"/>*

*<TextBlock Text="{Binding Worker.PassportAddress.House}"*

*Margin="5 0 0 0"/>*

*</StackPanel>*

*</Button>*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Адрес проживания"*

*Style="{StaticResource EditField}"/>*

*<Button Height="30"*

*Width="200"*

*Style="{StaticResource ButtonAddressEdit}"*

*Command="{Binding EditAddressCommand}">*

*<StackPanel Orientation="Horizontal">*

*<TextBlock Text="{Binding Worker.Address.Country}"*

*Margin="5 0 0 0"/>*

*<TextBlock Text="{Binding Worker.Address.City}"*

*Margin="5 0 0 0"/>*

*<TextBlock Text="{Binding Worker.Address.Street}"*

*Margin="5 0 0 0"/>*

*<TextBlock Text="{Binding Worker.Address.House}"*

*Margin="5 0 0 0"/>*

*</StackPanel>*

*</Button>*

*</StackPanel>*

*</StackPanel>*

*</StackPanel>*

*<!--#endregion-->*

*<TextBlock Text="\* - обязательные поля"*

*Grid.Row="1"*

*VerticalAlignment="Bottom"*

*Margin="40 10"*

*FontSize="10"*

*Foreground="{StaticResource TextGrayColorBrush}"/>*

*<Button Grid.Row="1"*

*Style="{StaticResource EditPasswordButtons}"*

*Content="Изменить изображение"*

*Width="200"*

*HorizontalAlignment="Right"*

*VerticalAlignment="Bottom"*

*Margin="35 60"*

*Command="{Binding ChooseImageCommand}"/>*

*<Button Grid.Row="1"*

*Style="{StaticResource EditPasswordButtons}"*

*Content="Изменить пароль\*"*

*Width="200"*

*HorizontalAlignment="Right"*

*VerticalAlignment="Bottom"*

*Margin="35 10"*

*Command="{Binding EditPasswordCommand}"/>*

*<!--#region Buttons-->*

*<Button Content="Сохранить"*

*Grid.Row="2"*

*Style="{StaticResource EditButtons}"*

*HorizontalAlignment="Left"*

*Command="{Binding SaveCommand}"/>*

*<Button Grid.Row="2"*

*Style="{StaticResource EditButtons}"*

*Content="Отмена"*

*HorizontalAlignment="Right"*

*Command="{Binding CancelCommand}"/>*

*<!--#endregion-->*

*</Grid>*

*</Window>*

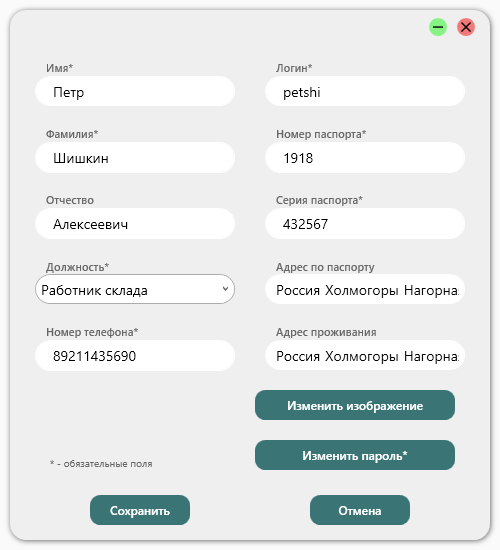


Рисунок 6 – Окно редактирования сотрудника.

Окно редактирования адреса содержит поля для заполнения с подписями, в соответствии с рисунком 7. Внизу окна имеются кнопки «Сохранить» и «Отмена».

*<Window x:Class="DeliveryService.View.AddressEdit"*

*xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"*

*xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"*

*xmlns:d="http://schemas.microsoft.com/expression/blend/2008"*

*xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"*

*xmlns:icons="http://metro.mahapps.com/winfx/xaml/iconpacks"*

*mc:Ignorable="d"*

*Title="AddressEdit" Height="450" Width="300"*

*Style="{StaticResource WinChrome}"*

*ResizeMode="NoResize">*

*<Grid>*

*<Grid.RowDefinitions>*

*<RowDefinition Height="30" />*

*<RowDefinition />*

*<RowDefinition Height="60" />*

*</Grid.RowDefinitions>*

*<!--#region Controls-->*

*<StackPanel x:Name="SpControl">*

*<StackPanel Orientation="Horizontal"*

*VerticalAlignment="Center"*

*Margin="10 8"*

*FlowDirection="RightToLeft">*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlCloseColorBrush}"*

*Command="{Binding CloseWindowCommand}">*

*<icons:PackIconMaterial Kind="Close"*

*Width="10"*

*Height="10" />*

*</Button>*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlMinColorBrush}"*

*Command="{Binding MinimizeWindowCommand}">*

*<icons:PackIconMaterial Kind="Minus"*

*Width="10"*

*Height="10" />*

*</Button>*

*</StackPanel>*

*</StackPanel>*

*<!--#endregion-->*

*<StackPanel Grid.Row="1"*

*HorizontalAlignment="Center">*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Страна"*

*Style="{StaticResource EditField}" />*

*<TextBox Text="{Binding Address.Country}"*

*Width="200"*

*Style="{StaticResource TextBoxSearch}" />*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Город"*

*Style="{StaticResource EditField}" />*

*<TextBox Text="{Binding Address.City}"*

*Width="200"*

*Style="{StaticResource TextBoxSearch}" />*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Улица"*

*Style="{StaticResource EditField}" />*

*<TextBox Text="{Binding Address.Street}"*

*Width="200"*

*Style="{StaticResource TextBoxSearch}" />*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Дом"*

*Style="{StaticResource EditField}" />*

*<TextBox Text="{Binding Address.House}"*

*Width="200"*

*Style="{StaticResource TextBoxSearch}" />*

*</StackPanel>*

*<StackPanel Margin="0 10">*

*<TextBlock Text="Почтовый индекс"*

*Style="{StaticResource EditField}" />*

*<TextBox Text="{Binding Address.Postcode}"*

*Width="200"*

*Style="{StaticResource TextBoxSearch}" />*

*</StackPanel>*

*</StackPanel>*

*<!--#region Buttons-->*

*<Button Content="Сохранить"*

*Grid.Row="2"*

*Style="{StaticResource EditButtons}"*

*HorizontalAlignment="Left"*

*Margin="20 0"*

*Command="{Binding SaveCommand}" />*

*<Button Grid.Row="2"*

*Style="{StaticResource EditButtons}"*

*Content="Отмена"*

*HorizontalAlignment="Right"*

*Margin="20 0"*

*Command="{Binding CancelCommand}" />*

*<!--#endregion-->*

*</Grid>*

*</Window>*

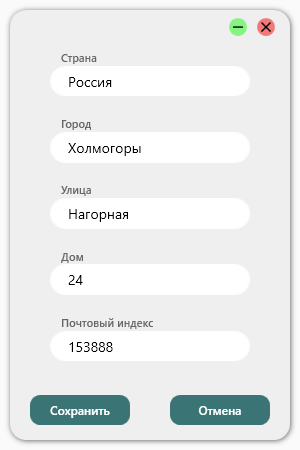


Рисунок 7 – Окно редактирования адреса.

Окно редактирования пароля содержит подписанное поле для ввода пароля и кнопки «Сохранить», «Отмена», в соответствии с рисунком 8.

*<Window x:Class="DeliveryService.View.Pages.Workers.WorkerEditPassword"*

*xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"*

*xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"*

*xmlns:d="http://schemas.microsoft.com/expression/blend/2008"*

*xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"*

*xmlns:i="http://schemas.microsoft.com/xaml/behaviors"*

*xmlns:helpers="clr-namespace:DeliveryService.Helpers"*

*xmlns:behaviors="clr-namespace:DeliveryService.Resources.Behaviors"*

*xmlns:icons="http://metro.mahapps.com/winfx/xaml/iconpacks"*

*mc:Ignorable="d"*

*Title="WorkerEditPassword" Height="200" Width="400"*

*ResizeMode="NoResize"*

*Style="{StaticResource WinChrome}">*

*<Grid>*

*<Grid.RowDefinitions>*

*<RowDefinition Height="30" />*

*<RowDefinition />*

*<RowDefinition Height="60" />*

*</Grid.RowDefinitions>*

*<!--#region Controls-->*

*<StackPanel x:Name="SpControl">*

*<StackPanel Orientation="Horizontal"*

*VerticalAlignment="Center"*

*Margin="10 8"*

*FlowDirection="RightToLeft">*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlCloseColorBrush}"*

*Command="{Binding CloseWindowCommand}">*

*<icons:PackIconMaterial Kind="Close"*

*Width="10"*

*Height="10" />*

*</Button>*

*<Button Style="{StaticResource ControlButton}"*

*Background="{StaticResource ControlMinColorBrush}"*

*Command="{Binding MinimizeWindowCommand}">*

*<icons:PackIconMaterial Kind="Minus"*

*Width="10"*

*Height="10" />*

*</Button>*

*</StackPanel>*

*</StackPanel>*

*<!--#endregion-->*

*<StackPanel Grid.Row="1"*

*HorizontalAlignment="Center"*

*VerticalAlignment="Center">*

*<TextBlock Text="Пароль"*

*Style="{StaticResource EditField}" />*

*<PasswordBox x:Name="PbPassword"*

*Width="200"*

*Style="{StaticResource PasswordBoxEdit}">*

*<i:Interaction.Behaviors>*

*<behaviors:PasswordBehavior />*

*</i:Interaction.Behaviors>*

*<helpers:PasswordHelper.Password>*

*<Binding Path="Password" Mode="TwoWay" UpdateSourceTrigger="PropertyChanged" />*

*</helpers:PasswordHelper.Password>*

*</PasswordBox>*

*<ProgressBar Width="170"*

*Margin="0 3"*

*Value="{Binding ProgressPassword}"*

*Maximum="100" />*

*</StackPanel>*

*<!--#region Buttons-->*

*<Button Content="Сохранить"*

*Grid.Row="2"*

*Style="{StaticResource EditButtons}"*

*HorizontalAlignment="Left"*

*Command="{Binding SaveCommand}" />*

*<Button Grid.Row="2"*

*Style="{StaticResource EditButtons}"*

*Content="Отмена"*

*HorizontalAlignment="Right"*

*Command="{Binding CancelCommand}" />*

*<!--#endregion-->*

*</Grid>*

*</Window>*

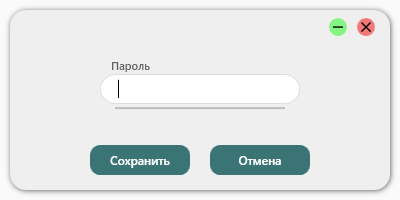


Рисунок 8 – Окно редактирования пароля.

Подраздел «Пароли» раздела «Сотрудники» содержит карточки сотрудников с идентификатором, изображением, ФИО и кнопкой «Изменить пароль», в соответствии с рисунком 9. Также имеет строку поиска для лучшей навигации по подразделу.

*<Page x:Class="DeliveryService.View.Pages.Workers.WorkerPasswordPage"*

*xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"*

*xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"*

*xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"*

*xmlns:d="http://schemas.microsoft.com/expression/blend/2008"*

*xmlns:icons="http://metro.mahapps.com/winfx/xaml/iconpacks"*

*xmlns:converters="clr-namespace:DeliveryService.Converters"*

*mc:Ignorable="d"*

*d:DesignHeight="450" d:DesignWidth="800"*

*Title="WorkerPasswordPage">*

*<Grid>*

*<Grid.RowDefinitions>*

*<RowDefinition Height="50"/>*

*<RowDefinition/>*

*</Grid.RowDefinitions>*

*<!--Top Menu-->*

*<Grid>*

*<StackPanel Orientation="Horizontal" />*

*<StackPanel Orientation="Horizontal"*

*FlowDirection="RightToLeft">*

*<!--Search TextBox-->*

*<Grid Width="250"*

*FlowDirection="LeftToRight">*

*<TextBlock Margin="20 0"*

*Foreground="{StaticResource TextGrayColorBrush}"*

*VerticalAlignment="Center"*

*IsHitTestVisible="False"*

*Panel.ZIndex="1"*

*Text="Поиск..."*

*Visibility="{Binding ElementName=TbSearch, Path=Text.IsEmpty, Converter={converters:VisibilityBoolConverter}}"/>*

*<TextBox x:Name="TbSearch"*

*Height="30"*

*Style="{StaticResource TextBoxSearch}"/>*

*<icons:PackIconMaterial Kind="Magnify"*

*Width="15"*

*Height="15"*

*VerticalAlignment="Center"*

*HorizontalAlignment="Right"*

*Margin="10 0"*

*Foreground="{StaticResource TextGrayColorBrush}"/>*

*</Grid>*

*</StackPanel>*

*</Grid>*

*<ScrollViewer Grid.Row="1"*

*VerticalScrollBarVisibility="Auto"*

*HorizontalScrollBarVisibility="Disabled">*

*<ItemsControl ItemsSource="{Binding Workers}" Width="Auto" Height="Auto">*

*<ItemsControl.ItemsPanel>*

*<ItemsPanelTemplate>*

*<WrapPanel />*

*</ItemsPanelTemplate>*

*</ItemsControl.ItemsPanel>*

*<ItemsControl.ItemTemplate>*

*<DataTemplate>*

*<Border BorderThickness="1"*

*Margin="5"*

*Background="White"*

*CornerRadius="12">*

*<Border.BorderBrush>*

*<SolidColorBrush Color="{StaticResource TextGrayColor}" Opacity="0.4"/>*

*</Border.BorderBrush>*

*<StackPanel Margin="5">*

*<TextBlock Text="{Binding Id}"*

*Style="{StaticResource EditField}"*

*FontSize="12"*

*Margin="5 0"/>*

*<Border Height="190"*

*Width="190"*

*Background="White"*

*BorderBrush="{StaticResource PrimaryColorBrush}"*

*BorderThickness="1"*

*CornerRadius="21"*

*Margin="0 0 0 5">*

*<Rectangle RadiusX="21"*

*RadiusY="21">*

*<Rectangle.Fill>*

*<ImageBrush ImageSource="{Binding WorkerImage}"/>*

*</Rectangle.Fill>*

*</Rectangle>*

*</Border>*

*<StackPanel Orientation="Horizontal">*

*<TextBlock Text="{Binding FirstName}"*

*FontSize="14"*

*FontWeight="SemiBold"*

*Margin="5 0"/>*

*<TextBlock Text="{Binding LastName}"*

*FontSize="14"*

*FontWeight="SemiBold"*

*Margin="5 0"/>*

*</StackPanel>*

*<TextBlock Text="{Binding Patronymic}"*

*FontSize="12"*

*FontWeight="SemiBold"*

*Margin="5 0"/>*

*<Button Style="{StaticResource EditPasswordButtons}"*

*Content="Изменить пароль"*

*Width="200"*

*HorizontalAlignment="Center"*

*Margin="0 5"*

*Command="{Binding DataContext.EditPasswordCommand, RelativeSource={RelativeSource AncestorType={x:Type ItemsControl}}}"*

*CommandParameter="{Binding Id}"/>*

*</StackPanel>*

*</Border>*

*</DataTemplate>*

*</ItemsControl.ItemTemplate>*

*</ItemsControl>*

*</ScrollViewer>*

*</Grid>*

*</Page>*

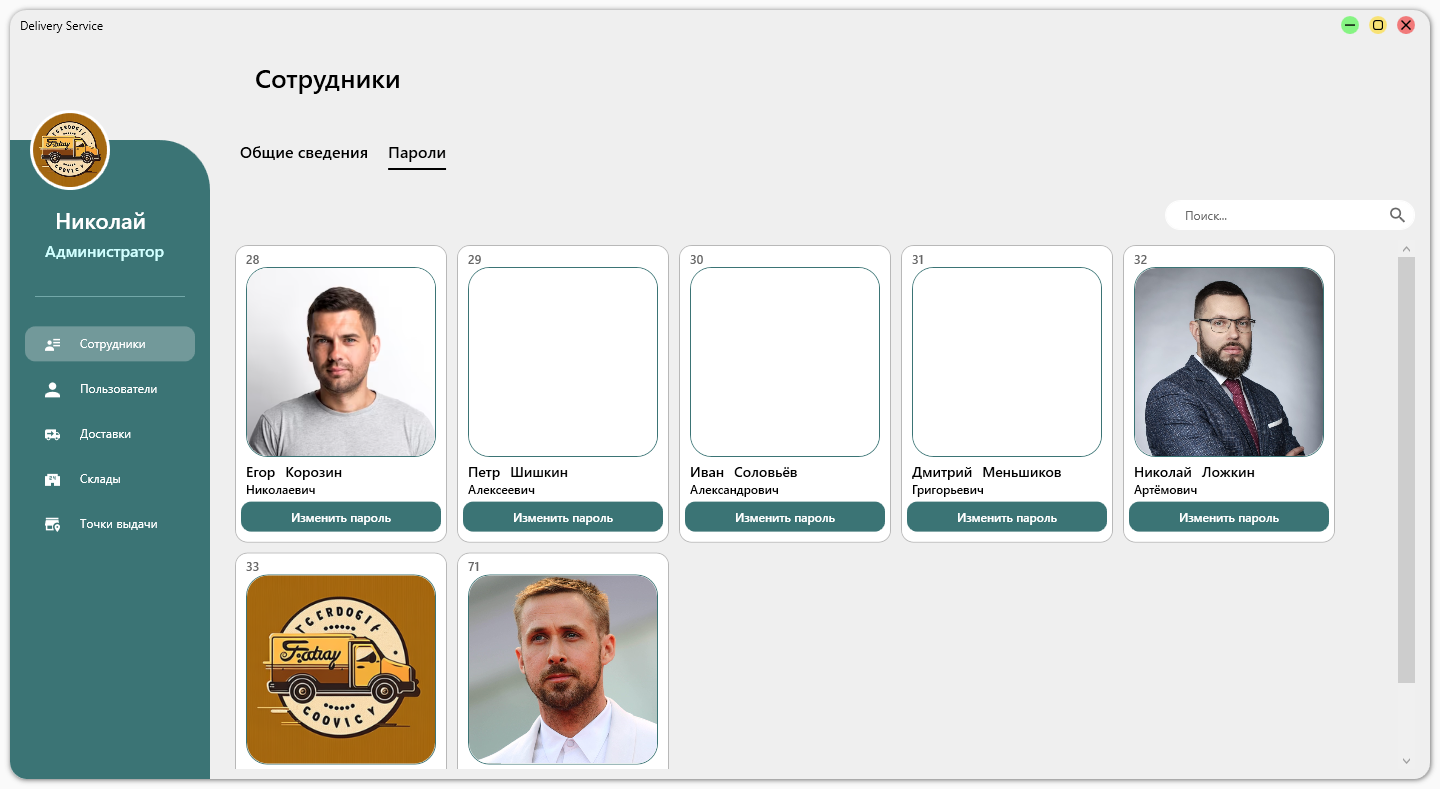


Рисунок 9 – Раздел «Сотрудники» подраздел «Пароли».

Все оставшиеся разделы и подразделы имеют схожий интерфейс с разделом «Сотрудники» и подразделом «Общие сведения».

* 1. Разработка программного функционала

Для возможности взаимодействия с интерфейсом необходимо обеспечить работу с данными. В приложении реализованы функции просмотра, удаления, изменения и добавления данных. Для работы с данными были созданы специальные классы Data Transferred Object (DTO), которые содержат необходимую информацию для подраздела. Возможность изменения, удаления и добавления DTO предоставляется благодаря сервисам, созданным для каждого DTO отдельно, там же они и заполняются данными благодаря Mapper, который создан отдельно для каждого DTO. Mapper формирует из модели DTO и обратно. Так как каждый DTO состоит из одной или нескольких моделей для каждой модели был создан репозиторий (Repository) в котором происходит добавление, удаление и изменение данных в модели. Такая структура позволяет создавать подразделы и работать с одним классом, не задумываясь об изменении моделей, которые входят в его состав.

Элементы графического интерфейса называются представлением (Views), они не имеют функциональности, этим занимаются специальные классы, называемые представлением модели (View Model). Для получения и добавления данных был использован механизм привязки (Binding). Данные хранятся в коллекции ObservableCollection, которая позволяет обновлять данные в представлении при изменении элемента. Для обработки некоторых событий созданы команды. Они используются для выполнения обработки события внутри View Model, таким образом функционал представления находится в одном месте.

Конструктор представления модели раздела «Сотрудники» не принимает ничего в качестве аргументов. Он инициализирует поля сервисов значениями, которые берутся из поставщика сервисов и вызывает метод обновления данных.

*public WorkerGeneralInfoViewModel()*

*{*

*\_service = App.ServiceProvider.GetRequiredService<IWorkerDTOService>();*

*\_authenticationService = App.ServiceProvider.GetRequiredService<IAuthenticationService>();*

*\_pdfWriterService = App.ServiceProvider.GetRequiredService<IPdfWriterService>();*

*UpdateData();*

*}*

Метод обновления данных получает данные из сервиса и присваивает их коллекции рабочих.

*private void UpdateData()*

*{*

*Workers = new ObservableCollection<WorkerDTO>(\_service.GetAll());*

*}*

Команды, выполняющиеся при событии нажатия на соответствующую кнопку.

*private RelayCommand? \_editWorkerCommand;*

*private RelayCommand? \_addWorkerCommand;*

*private RelayCommand? \_deleteWorkerCommand;*

*private RelayCommand? \_createPdf;*

*/// <summary>*

*/// Команда для открытия окна редактирования сотрудника.*

*/// </summary>*

*public RelayCommand EditWorkerCommand*

*{*

*get*

*{*

*return \_editWorkerCommand ??= new RelayCommand((obj) =>*

*{*

*if (\_authenticationService.HasPermissionToModifySubsection(WorkerPages.WorkerGeneralInfo))*

*{*

*if (obj != null)*

*{*

*Window window = new WorkerEdit((int)obj);*

*window.ShowDialog();*

*UpdateData();*

*}*

*}*

*else*

*{*

*Window window = new ErrorWindow("Не удалось выполнить действие. Недостаточно прав.");*

*window.ShowDialog();*

*}*

*});*

*}*

*}*

*/// <summary>*

*/// Команда для удаления сотрудника.*

*/// </summary>*

*public RelayCommand DeleteWorkerCommand*

*{*

*get*

*{*

*return \_deleteWorkerCommand ??= new RelayCommand((obj) =>*

*{*

*if (\_authenticationService.HasPermissionToModifySubsection(WorkerPages.WorkerGeneralInfo))*

*{*

*if (obj != null)*

*{*

*\_service.Remove((int)obj);*

*UpdateData();*

*}*

*}*

*else*

*{*

*Window window = new ErrorWindow("Не удалось выполнить действие. Недостаточно прав.");*

*window.ShowDialog();*

*}*

*});*

*}*

*}*

*/// <summary>*

*/// Команда для открытия окна добавления сотрудника.*

*/// </summary>*

*public RelayCommand AddWorkerCommand*

*{*

*get*

*{*

*return \_addWorkerCommand ??= new RelayCommand((obj) =>*

*{*

*if (\_authenticationService.HasPermissionToModifySubsection(WorkerPages.WorkerGeneralInfo))*

*{*

*Window window = new WorkerEdit(null);*

*window.ShowDialog();*

*UpdateData();*

*}*

*else*

*{*

*Window window = new ErrorWindow("Не удалось выполнить действие. Недостаточно прав.");*

*window.ShowDialog();*

*}*

*});*

*}*

*}*

*/// <summary>*

*/// Команда для создания PDF-файла.*

*/// </summary>*

*public RelayCommand CreatePdf*

*{*

*get*

*{*

*return \_createPdf ??= new RelayCommand((obj) =>*

*{*

*\_pdfWriterService.CreatePdfFile(new List<WorkerDTO>(\_service.GetAll()),*

*new Dictionary<string, string>*

*{*

*{ "Id", "Id" },*

*{ "FirstName", "Имя" },*

*{ "LastName", "Фамилия" },*

*{ "Patronymic", "Отчество" },*

*{ "Title", "Должность" },*

*{ "Login", "Логин" },*

*{ "PassportNumber", "Номер паспорта" },*

*{ "PassportSeries", "Серия паспорта" },*

*{ "TelephoneNumber", "Номер телефона" },*

*{ "Address", "Адрес проживания" },*

*{ "PassportAddress", "Адрес по паспорту" },*

*});*

*});*

*}*

*}*

Команды вызывают методы у сервиса, который позволяет работать с данными. Код сервиса:

using System.Collections.Generic;

using DeliveryService.DTO;

using DeliveryService.Mappers;

using DeliveryService.Model;

using DeliveryService.Repository;

namespace DeliveryService.Services

{

public class WorkerDTOService : IWorkerDTOService

{

private IWorkerRepository \_workerRepository;

private IUserRepository \_userRepository;

public WorkerDTOService(IWorkerRepository workerRepository, IUserRepository userRepository)

{

\_workerRepository = workerRepository;

\_userRepository = userRepository;

}

public void Add(WorkerDTO workerGeneralInfoDto)

{

Worker worker = new Worker();

User user = new User();

worker.IdNavigation = user;

WorkerMapper.Map(workerGeneralInfoDto, worker);

\_userRepository.Add(user);

worker.Id = user.Id;

\_workerRepository.Add(worker);

}

public void Edit(WorkerDTO workerGeneralInfoDto)

{

Worker worker = \_workerRepository.GetById(workerGeneralInfoDto.Id);

WorkerMapper.Map(workerGeneralInfoDto, worker);

\_workerRepository.Edit(worker);

}

public void Remove(int id)

{

\_workerRepository.Remove(id);

}

public WorkerDTO GetById(int id)

{

return WorkerMapper.Map(\_workerRepository.GetById(id));

}

public IEnumerable<WorkerDTO> GetAll()

{

return WorkerMapper.MapAll(\_workerRepository.GetAll());

}

}

}

Сервис в свою очередь использует соответствующие Mapper и репозитории для работы с данными. Код класса Mapper сотрудника:

*using System.Collections.Generic;*

*using System.Linq;*

*using DeliveryService.DTO;*

*using DeliveryService.Model;*

*using Microsoft.Extensions.DependencyInjection;*

*namespace DeliveryService.Mappers*

*{*

*/// <summary>*

*/// Класс для преобразования модели <see cref="Worker"/> в <see cref="WorkerDTO"/> и обратно.*

*/// </summary>*

*public static class WorkerMapper*

*{*

*private static DsContext \_context = App.ServiceProvider.GetRequiredService<DsContext>();*

*/// <summary>*

*/// Метод для преобразования <see cref="Worker"/> в <see cref="WorkerDTO"/>.*

*/// </summary>*

*/// <param name="worker">Модель <see cref="Worker"/> по которой нужно выполнить преобразование.</param>*

*/// <returns>DTO объект <see cref="WorkerDTO"/>, полученный в результате преобразования.</returns>*

*public static WorkerDTO Map(Worker worker)*

*{*

*return new WorkerDTO*

*{*

*Id = worker.Id,*

*Login = worker.Login,*

*Password = worker.Password,*

*FirstName = worker.IdNavigation.Firstname,*

*LastName = worker.IdNavigation.Lastname,*

*Patronymic = worker.IdNavigation.Patronymic,*

*PositionId = worker.PositionId,*

*Title = worker.Position.Title,*

*PassportNumber = worker.IdNavigation.PassportNumber,*

*PassportSeries = worker.IdNavigation.PassportSeries,*

*TelephoneNumber = worker.IdNavigation.TelephoneNumber,*

*Address = worker.IdNavigation.Address == null ? null : AddressMapper.Map(worker.IdNavigation.Address),*

*PassportAddress = worker.IdNavigation.PassportAddressNavigation == null ? null : AddressMapper.Map(worker.IdNavigation.PassportAddressNavigation),*

*WorkerImage = worker.Image?.WorkerImage1*

*};*

*}*

*/// <summary>*

*/// Метод для преобразования объекта <see cref="WorkerDTO"/> в <see cref="Worker"/>.*

*/// </summary>*

*/// <param name="workerDto">DTO объект <see cref="WorkerDTO"/> по которому нужно выполнить преобразование.</param>*

*/// <param name="worker">Модель <see cref="Worker"/>, полученная в результате преобразования.</param>*

*public static void Map(WorkerDTO workerDto, Worker worker)*

*{*

*worker.Login = workerDto.Login;*

*worker.Password = workerDto.Password;*

*worker.IdNavigation.Firstname = workerDto.FirstName;*

*worker.IdNavigation.Lastname = workerDto.LastName;*

*worker.IdNavigation.Patronymic = workerDto.Patronymic;*

*worker.PositionId = workerDto.PositionId;*

*worker.IdNavigation.PassportNumber = workerDto.PassportNumber;*

*worker.IdNavigation.PassportSeries = workerDto.PassportSeries;*

*worker.IdNavigation.TelephoneNumber = workerDto.TelephoneNumber;*

*if (workerDto.WorkerImage != null)*

*{*

*WorkerImage image = new WorkerImage() { WorkerImage1 = workerDto.WorkerImage };*

*\_context.WorkerImages.Add(image);*

*\_context.SaveChanges();*

*worker.Image = image;*

*}*

*if (workerDto.Address != null)*

*{*

*AddressMapper.Map(workerDto.Address, (worker.IdNavigation.Address = new Address()));*

*}*

*if (workerDto.PassportAddress != null)*

*{*

*AddressMapper.Map(workerDto.PassportAddress, (worker.IdNavigation.PassportAddressNavigation = new Address()));*

*}*

*}*

*/// <summary>*

*/// Метод для преобразования коллекции <see cref="Worker"/> в коллекцию <see cref="WorkerDTO"/>.*

*/// </summary>*

*/// <param name="workers">Коллекция <see cref="Worker"/>, которую нужно отобразить.</param>*

*/// <returns>Коллекция <see cref="WorkerDTO"/>, полученная в результате преобразования.</returns>*

*public static IEnumerable<WorkerDTO> MapAll(IEnumerable<Worker> workers)*

*{*

*return workers.ToList().ConvertAll(Map);*

*}*

*}*

*}*

Репозиторий позволяет добавлять, изменить и удалять записи. Так же в репозитории сделана проверка на возможность действий с данными и возвращение к предыдущим значениям, если произошла ошибка. Код класса репозитория сотрудника:

using System;

using System.Collections.Generic;

using System.Linq;

using DeliveryService.Model;

using Microsoft.EntityFrameworkCore.ChangeTracking;

namespace DeliveryService.Repository

{

/// <summary>

/// Репозиторий модели <see cref="Worker"/>, реализует соответствующий интерфейс репозитория.

/// </summary>

public class WorkerRepository : IWorkerRepository

{

#region Private Fields

private DsContext \_context;

#endregion

public WorkerRepository(DsContext context)

{

\_context = context;

}

#region Methods

public void Add(Worker worker)

{

EntityEntry<Worker> entry = \_context.Entry(worker);

try

{

\_context.Workers.Add(worker);

\_context.SaveChanges();

}

catch

{

entry.Reload();

throw new Exception("Не удалось добавить запись.");

}

}

public void Edit(Worker worker)

{

EntityEntry<Worker> entry = \_context.Entry(worker);

try

{

\_context.Workers.Update(worker);

\_context.SaveChanges();

}

catch

{

entry.Reload();

throw new Exception("Не удалось изменить запись.");

}

}

public IEnumerable<Worker> GetAll()

{

return \_context.Workers.ToList();

}

public Worker GetById(int id)

{

return \_context.Workers.Find(id) ?? throw new Exception();

}

public Worker? GetWorkerByLoginAndPassword(string login, byte[] password)

{

return \_context.Workers.FirstOrDefault(x => x.Login == login && x.Password == password);

}

public void Remove(int id)

{

Worker worker = GetById(id);

EntityEntry<Worker> entry = \_context.Entry(worker);

try

{

\_context.Workers.Remove(worker);

\_context.SaveChanges();

}

catch

{

entry.Reload();

throw new Exception("Не удалось удалить запись. Возможно присутствуют связанные записи.");

}

}

#endregion

}

}

В других разделах и подразделах реализация подобна показанной. Так же добавлена выгрузка в PDF файл, фильтрация таблиц, проверка на сложность пароля.

1. ТЕСТИРОВАНИЕ

Тестирование проводилось на устройстве с установленной операционной системой Windows 11.

* 1. Тестирование логики программы

Тестирование представлено в приложении 2. Приложение прошло тестирование успешно, обработаны многие сценарии, которые могли нарушить работу приложения. Критических ошибок не было обнаружено.

* 1. Тестирование интерфейса пользователя

Тестирование представлено в приложении 3. Тестирование пройдено успешно, отклонений в работе не было выявлено. Интерфейс приложения выполнен в соответствии требованиям.

1. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ
   1. Введение

Приложение разработано для эффективной работы сотрудников почтовой службы с заказами и простого получения доступа к информации, благодаря этому будут уменьшены затраты времени на аналитику, логистику и другие процессы работы с данными.

Данный модуль позволяет просматривать, добавлять, изменять и удалять данные связанные с пользователями, сотрудниками, доставками, адресами, пунктами выдачи и складами. Для удобного доступа к данным имеется возможность фильтрации и выгрузки данных в PDF файл для аналитики.

Пользоваться приложением могут люди, разбирающиеся в предметной области и умеющие пользоваться компьютером на базовом уровне.

* 1. Назначение и условия применения программного обеспечения

Приложение предназначено для работы сотрудников почтовой службы с данными. Для доступа сотрудников только в необходимые разделы было создано распределение по ролям. Администратору доступно изменение и просмотр любых данных в приложении. Бизнес-аналитику предоставлен доступ ко всем данным приложения, но только в режиме просмотра. Работнику склада доступно изменение и просмотр данных только о складах. Оператору точки выдачи доступна информация для изменения и просмотра о посылках в точках выдачи. Доставщик может изменять и просматривать информацию о посылках.

При необходимости в любом подразделе приложения имеется выгрузка в PDF файл для возможности аналитики данных или дальнейшей их печати.

Для работоспособности приложения необходима операционная система Windows 11 или Windows 10 и доступ к базе данных.

* 1. Подготовка к работе

Носитель данных содержит файл со скриптом базы данных, который называется DSe.sql. Требуется запустить данный файл для создания базы данных в Microsoft SQL Management Studio.Следующим шагом будет настройка приложения, для этого необходимо зайти папку Application и в ней найти и открыть файл DeliveryService.dll.config. После открытия изменить строку подключения.

Для запуска приложения необходимо запустить файл DeliveryService.exe. При запуске должно появится окно входа в систему.

* 1. Описание операций

При запуске приложения открывается окно входа в систему, в котором сотруднику необходимо ввести логин и пароль, выданные администратором, в соответствии с рисунком 10.

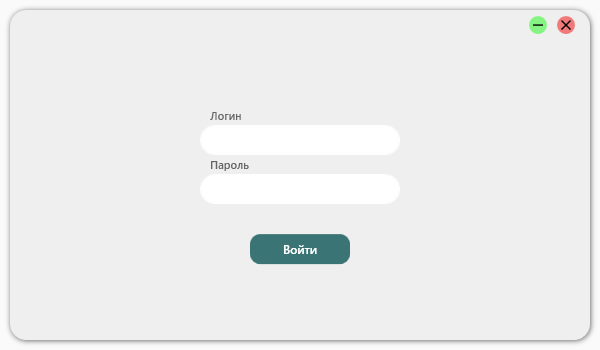


Рисунок 10 – Окно входа в систему.

После входа в систему сотруднику доступны разделы, соответствующие его правам доступа. При попытке зайти в раздел, к которому у пользователя нет доступа, он увидит сообщение, с предложением выбрать раздел.

В каждом разделе и подразделе у сотрудника доступны действия удаления, добавления и изменения данных, согласно его правам доступа. При попытке без необходимых прав взаимодействовать с данными, пользователю отображается сообщение о недостаточности прав.

В разделе «Сотрудники» представлено два подраздела «Общие сведения» и «Пароли», в соответствии с рисунком 11. В подразделе «Общие сведения» содержится информация о сотрудниках, а именно id, ФИО и изображение, телефон, адрес. Так же есть кнопки для изменения и удаления записей. Снизу находятся кнопки «Добавить новую запись» и «В PDF». Сверху-справа располагается строка поиска для фильтрации записей.

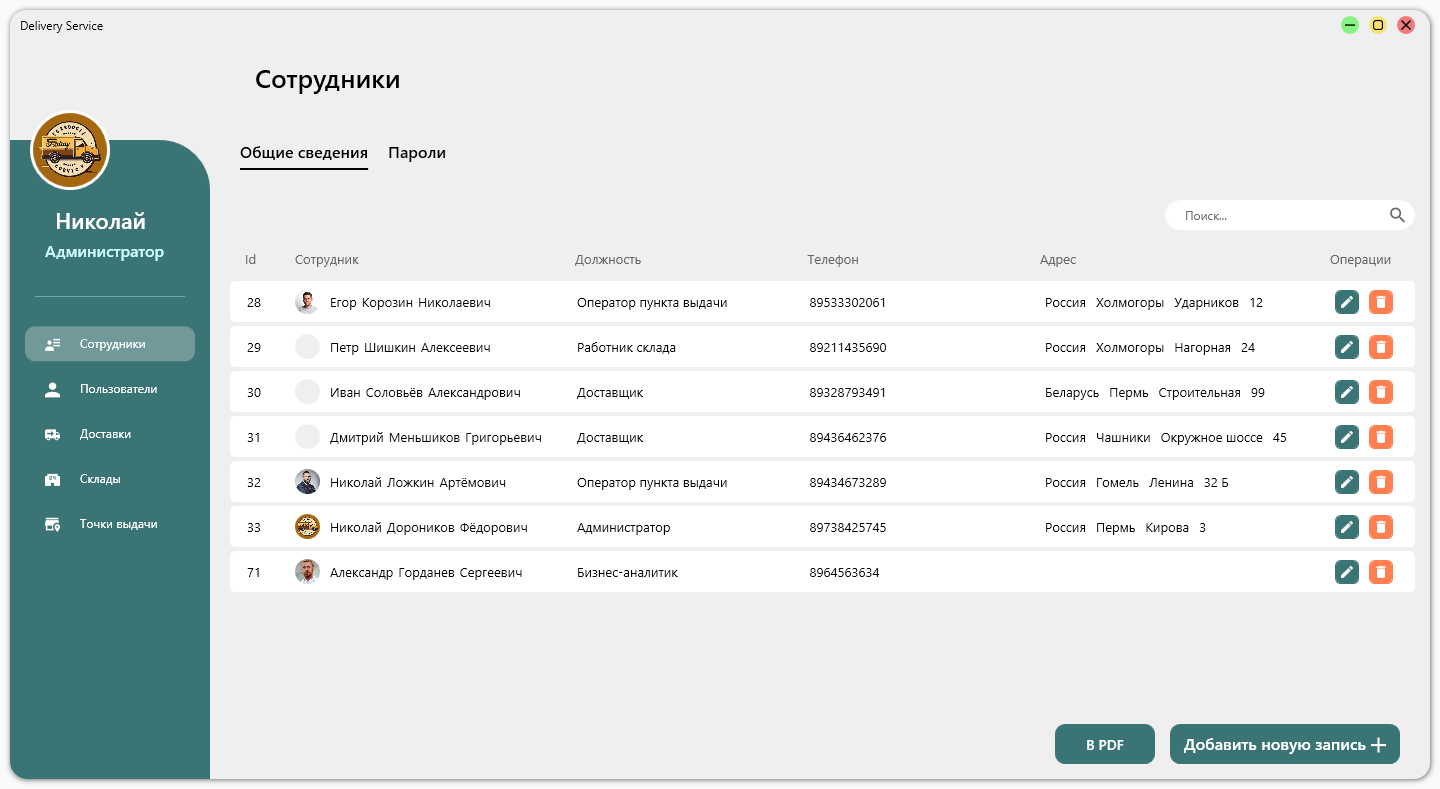


Рисунок 11 – Раздел «Сотрудники» подраздел «Общие требования».

Подраздел «Пароли» доступен только администратору и служит для быстрой смены пароля сотрудника, представлен в соответствии с рисунком 12. Записи представлены в виде карточек с кнопкой «Изменить пароль».

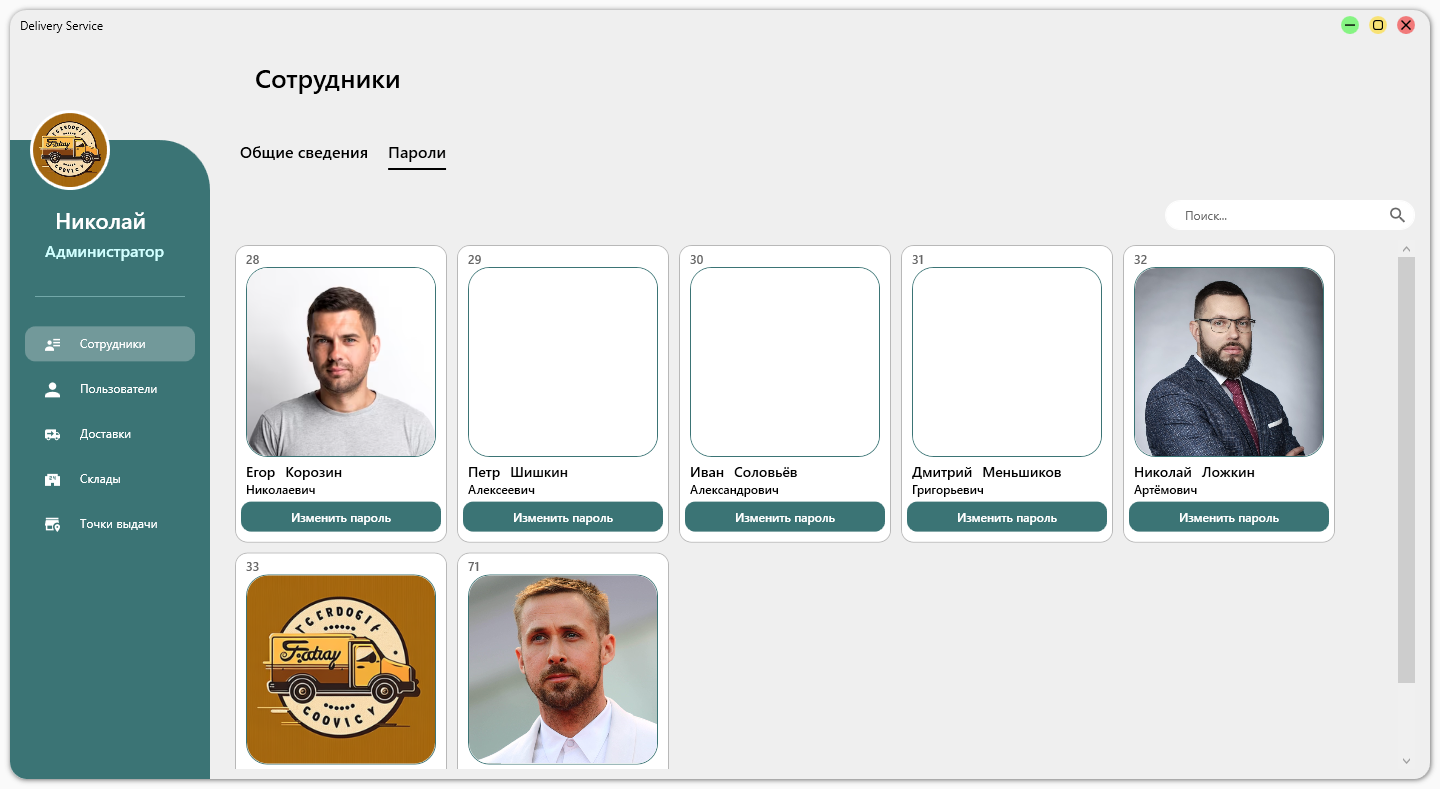


Рисунок 12 – Раздел «Сотрудники» подраздел «Пароли».

В разделе «Сотрудники» представлено несколько око для изменения данных. Окно изменения данных сотрудника содержит все данные необходимые сотрудника, в соответствии с рисунком 13. Поля обязательные к заполнению отмечены «\*».

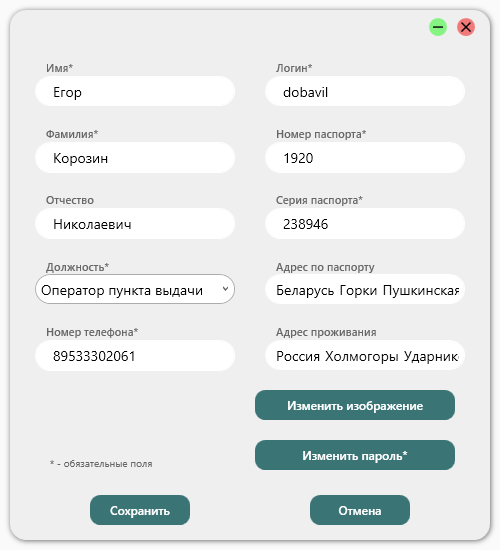


Рисунок 13 – Окно редактирования сотрудника.

Окно изменения адреса состоит из полей страны, города, улицы, номера дома и почтового индекса, в соответствии с рисунком 14.

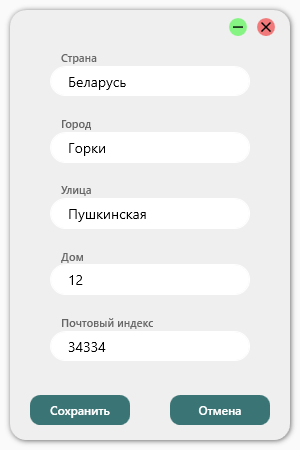


Рисунок 14 – Окно редактирования адреса.

Окно изменения пароля имеет поле ввода пароля и индикатор сложности, в соответствии с рисунком 15.

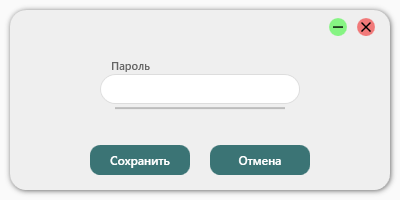


Рисунок 15 – Окно редактирования пароля.

Раздел «Пользователи» содержит подраздел «Основные сведения», в соответствии с рисунком 16. В подразделе предоставляются данные id, ФИО, телефон и адрес. Для стандартных действий над данными необходимы соответствующие права доступа.

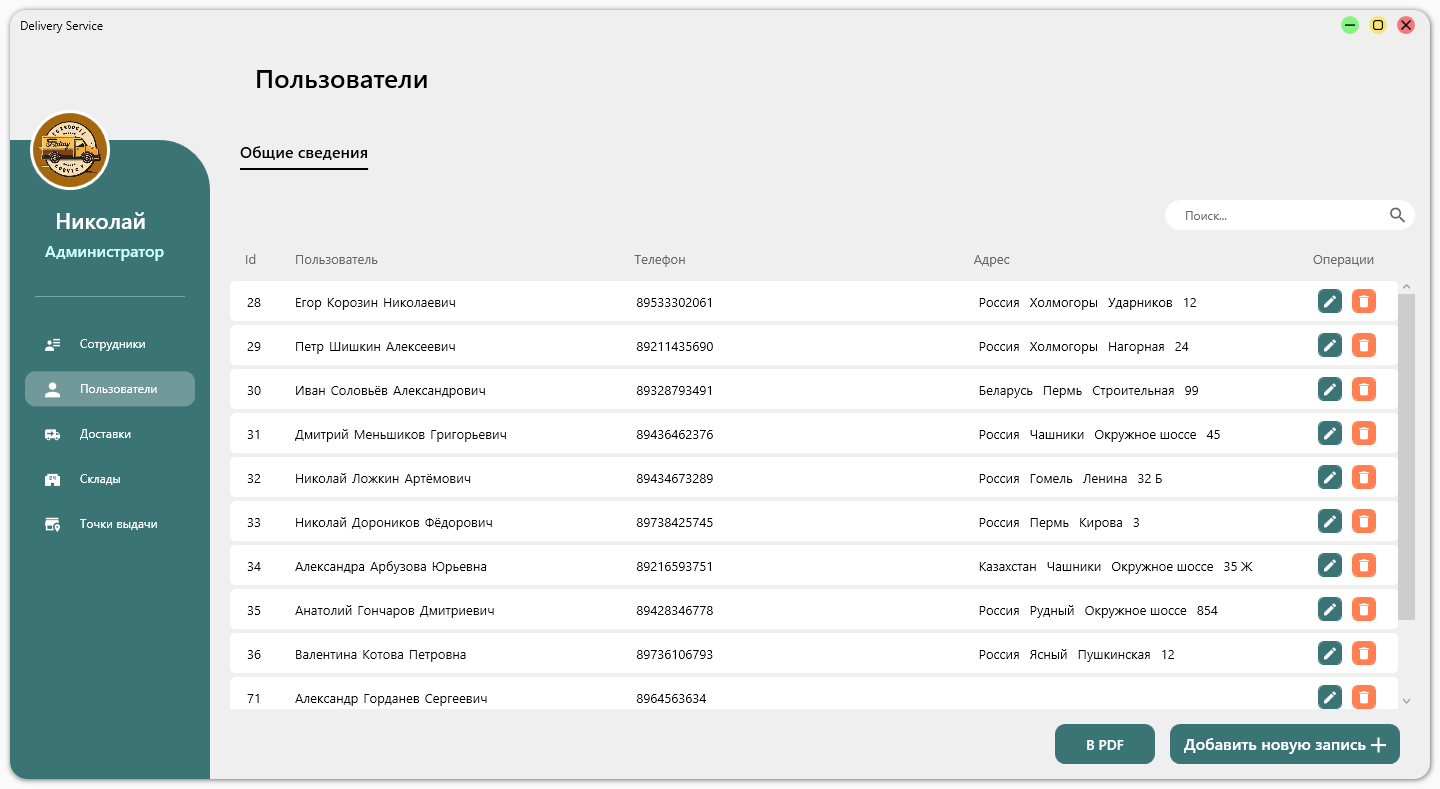


Рисунок 16 – Раздел «Пользователи» подраздел «Общие сведения».

Окно изменения пользователя содержит необходимые поля для изменения данных, часть полей совпадает с сотрудником, в соответствии с рисунком 17. Обязательные поля отмечены «\*».

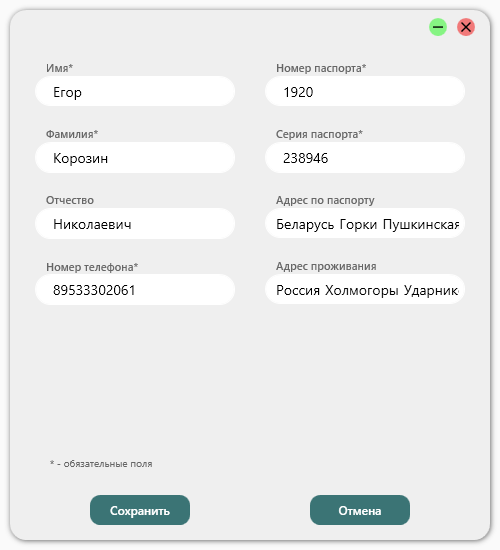


Рисунок 17 – Окно редактирования пользователя.

В разделе «Доставки» располагается единственный подраздел «Общие сведения», в соответствии с рисунком 18. Он содержит данные о доставке, а именно id, отправитель, получатель и цена.

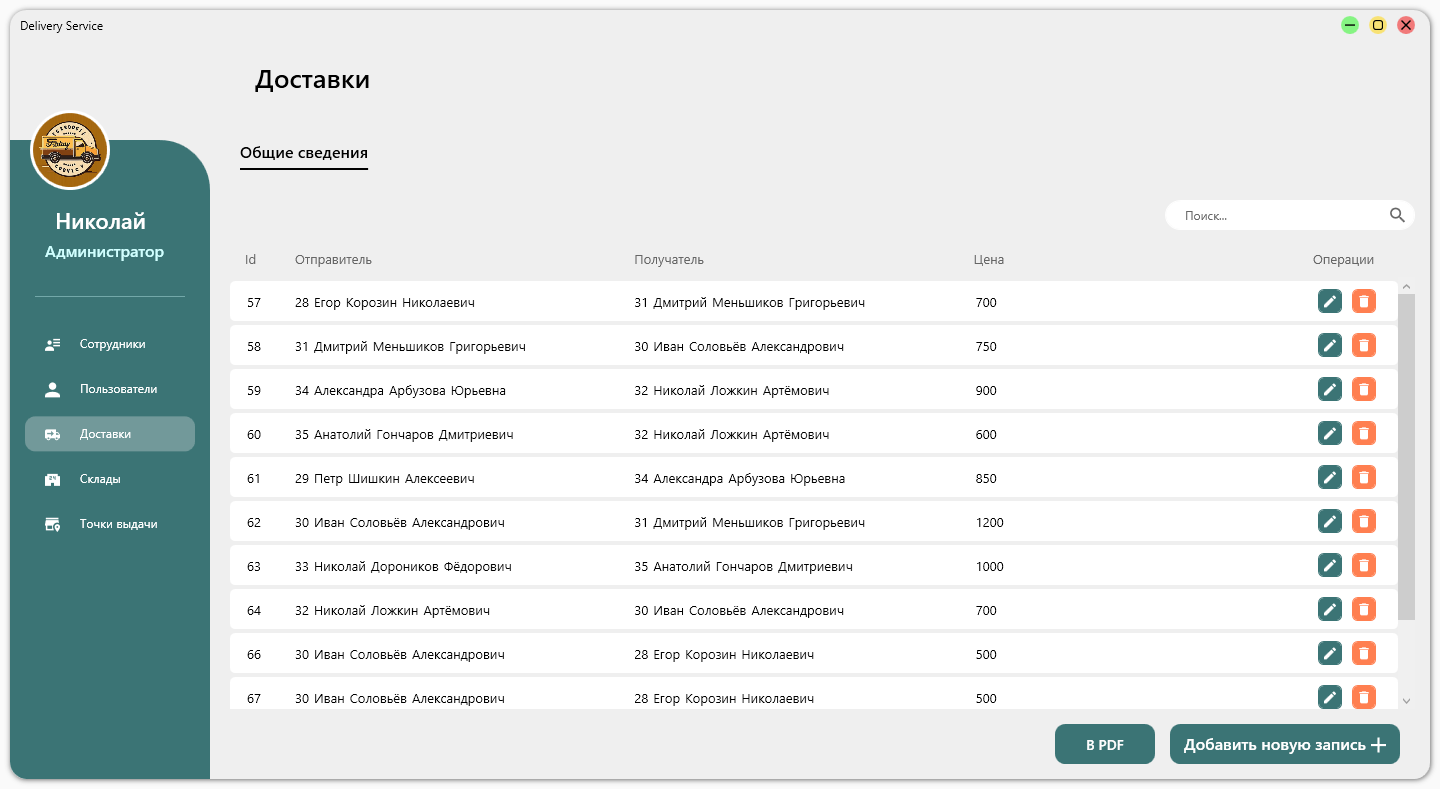


Рисунок 18 – Раздел «Доставки» подраздел «Общие сведения».

Окно изменения данных доставки содержит выбор получателя, отправителя, тарифа, адреса отправки, адреса получения и поле изменения цены, в соответствии с рисунком 19.

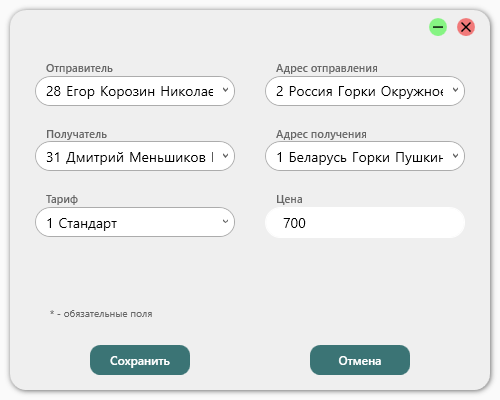


Рисунок 19 – Окно редактирования доставки.

Раздел «Склады» состоит из одного подраздела «Общие сведения», содержащий id склада, его номер и адрес, в соответствии с рисунком 20.

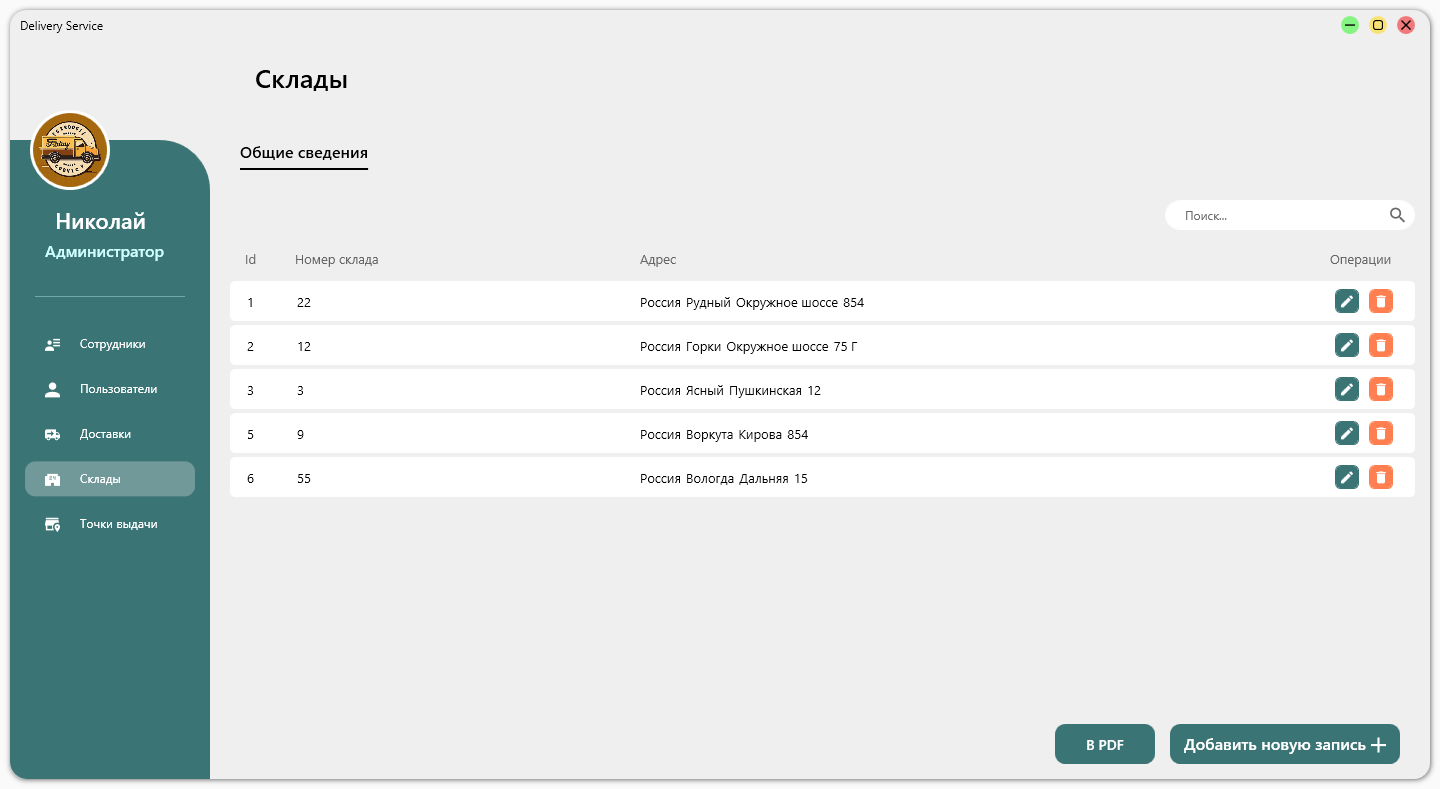


Рисунок 20 – Раздел «Склады» подраздел «Общие сведения».

Окно изменения склада предоставляет поле для ввода номера склада и кнопку, открывающую окно выбора адреса, в соответствии с рисунком 21.

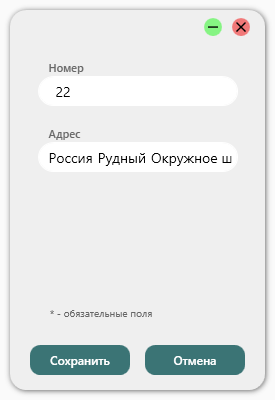


Рисунок 21– Окно редактирования склада.

Последний раздел «Точки выдачи» предоставляет доступ к подразделам «Общие сведения» и «Сотрудники в точках выдачи». В общих сведениях представлена информация о точке выдачи, а именно id и адрес, в соответствии с рисунком 22. Окно редактирования совпадает с окном редактирования адреса, но почтовый индекс является обязательным.

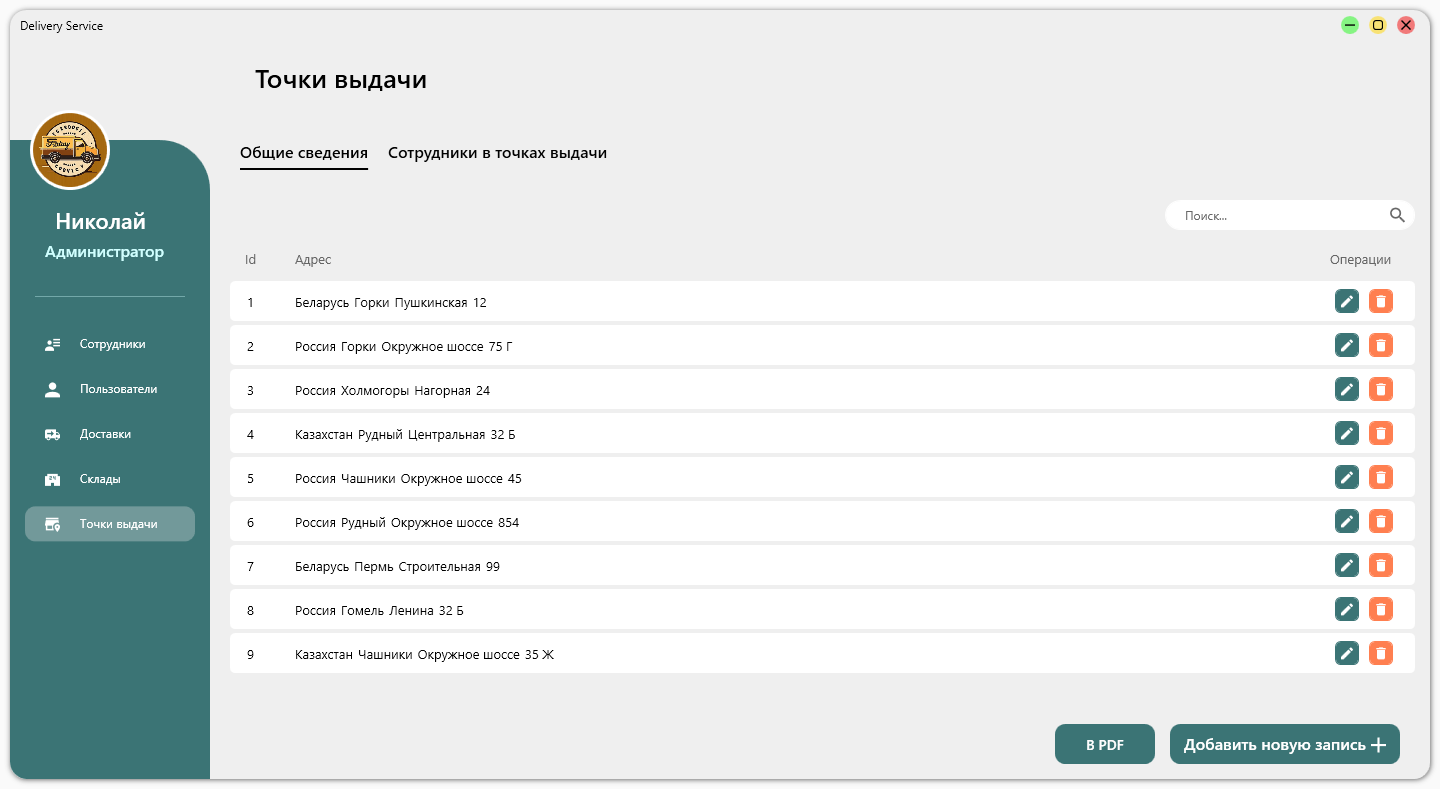


Рисунок 22 – Раздел «Точки выдачи» подраздел «Общие сведения».

Подраздел «Сотрудники в точках выдачи» состоит из id записи, сотрудника, точки выдачи и смены, в соответствии с рисунком 23.

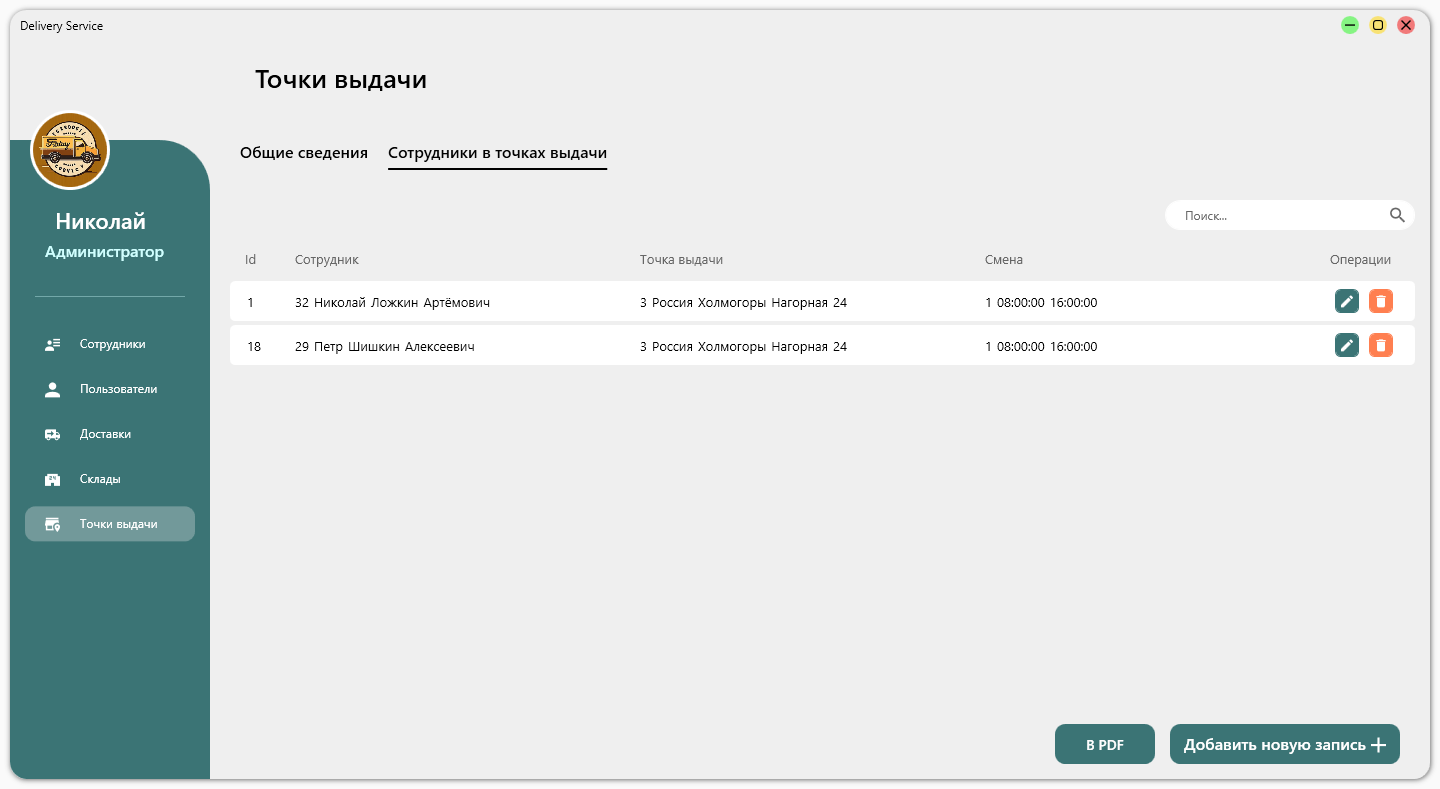


Рисунок 23 – Раздел «Точки выдачи» подраздел «Сотрудники в точках выдачи».

Окно изменения записи подраздела «Сотрудники в точках выдачи» содержит выбор сотрудника, точки выдачи и смены, в соответствии с рисунком 24.

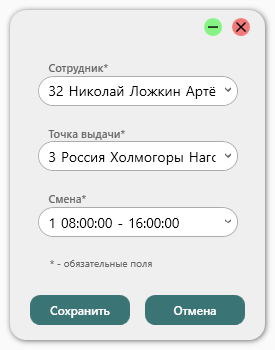


Рисунок 24 – Окно редактирования сотрудника в точке доступа.

ЗАКЛЮЧЕНИЕ

После успешного выполнения курсового проекта, направленного на разработку информационной системы для почтового сервиса, были достигнуты все поставленные задачи и цель в полном объеме. В ходе проекта были проведены комплексные анализы средств разработки, проведена разработка требований к будущему модулю и создан графический интерфейс для программы.

Основные функциональные возможности модуля заключаются в предоставлении сотрудникам возможности просмотра и изменения разнообразных данных, обеспечении их фильтрации и выгрузки в формат PDF. После завершения функционального кода, были проведены тесты всех разделов системы для подтверждения высокого качества разработки проекта.

Наконец, была создана обширная документация, которая подробно описывает различные аспекты проекта. Результаты работы не только позволили достичь поставленной цели, но и значительно повысили навыки и умения в разработке приложений.

Общий результат проекта может быть оценен как положительный, поскольку все этапы были успешно завершены, а полученный опыт может быть применен в будущих работах.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

**Электронные ресурсы**

**Ресурсы удалённого доступа**

1. Документация C# [Электронный ресурс]: официальный сайт. – Режим доступа: <https://learn.microsoft.com/ru-ru/dotnet/csharp/>
2. Документация WPF [Электронный ресурс]: официальный сайт. – Режим доступа: <https://learn.microsoft.com/ru-ru/dotnet/desktop/wpf/>
3. Документация Windows Forms [Электронный ресурс]: официальный сайт. – Режим доступа: <https://learn.microsoft.com/ru-ru/dotnet/desktop/winforms/>
4. QT [Электронный ресурс]: официальный сайт. – Режим доступа: <https://www.qt.io/>
5. JavaFX [Электронный ресурс]: официальный сайт. – Режим доступа: <https://openjfx.io/>
6. PyQT [Электронный ресурс]: официальный сайт. – Режим доступа: <https://riverbankcomputing.com/software/pyqt/>
7. Visual Studio [Электронный ресурс]: официальный сайт. – Режим доступа: <https://visualstudio.microsoft.com/ru/>
8. Rider [Электронный ресурс]: официальный сайт. – Режим доступа: <https://www.jetbrains.com/rider/>
9. Документация SQL Server [Электронный ресурс]: официальный сайт. – Режим доступа: <https://learn.microsoft.com/ru-ru/sql/sql-server/>
10. MySQL [Электронный ресурс]: официальный сайт. – Режим доступа: <https://www.mysql.com/>
11. Oracle [Электронный ресурс]: официальный сайт. – Режим доступа: <https://www.oracle.com/cis/database/>

ПРИЛОЖЕНИЕ 1

Бизнес-требования

Суть проекта:

Разработка приложения, позволяющего взаимодействовать сотрудникам с данными службы доставки.

Цели проекта:

* Возможность доступа сотрудникам к данным службы доставки, ограниченного по ролям.
* Упрощение поиска, изменения и дополнения информации в базе данных.

Системные характеристики

СХ-1 Приложение настольное.

СХ-2 Приложение разрабатывается на .Net 6.0 с применением WPF.

СХ-3 Операционная система Windows 10, 11.

Пользовательские требования

ПТ-1 Пользователи системы:

* доставщик;
* работник склада;
* оператор пункта выдачи;
* бизнес-аналитик;
* администратор информационной системы.

ПТ-2 Запуск и остановка приложения.

ПТ-2.1 Запуск приложения осуществляется двойным нажатием по исполняемому файлу приложения.

ПТ-2.2 Остановка приложения происходит при нажатии на крестик справа-сверху.

ПТ-3 Вход и выход из приложения.

ПТ-3.1 При открытии приложения появляется окно входа.

ПТ-3.2 При закрытии приложения происходит выход из системы.

ПТ-3.3 Логин и пароь сотрудникам выдаёт администратор.

ПТ-4 Взаимодействие с данными.

ПТ-4.1 Пользователь должен иметь возможность работы с данными информационной системы.

ПТ-4.2 Данные должны быть разбиты на разделы.

ПТ-5 Выгрузка данных в PDF файл.

ПТ-5.1 Должна быть возможность выгрузки даных в PDF файл.

ПТ-5 Доступ к данным сотрудников (в соответствии с рисунком 1).

ПТ-5.1 Доставщик имеет ограниченный доступ к данным, а именно к подразделу «Общие сведения» раздела «Доставки».

ПТ-5.2 Работник склада имеет ограниченный доступ к данным, а именно к подразделу «Общие сведения» раздела «Склады».

ПТ 5.3 Оператор пункта выдачи имеет ограниченный доступ к даным, а именно к подразделу «Общие сведения» раздела «Пункты выдачи».

ПТ-5.4 Бизнес-аналитик имеет доступ ко всем разделам только в режиме просмотра, кроме подраздела «Пароли» раздела «Сотрудники».

ПТ-5.5 Администратор информационной системы имеет неограниченный доступ ков всем разделам и всем операциям над данными.

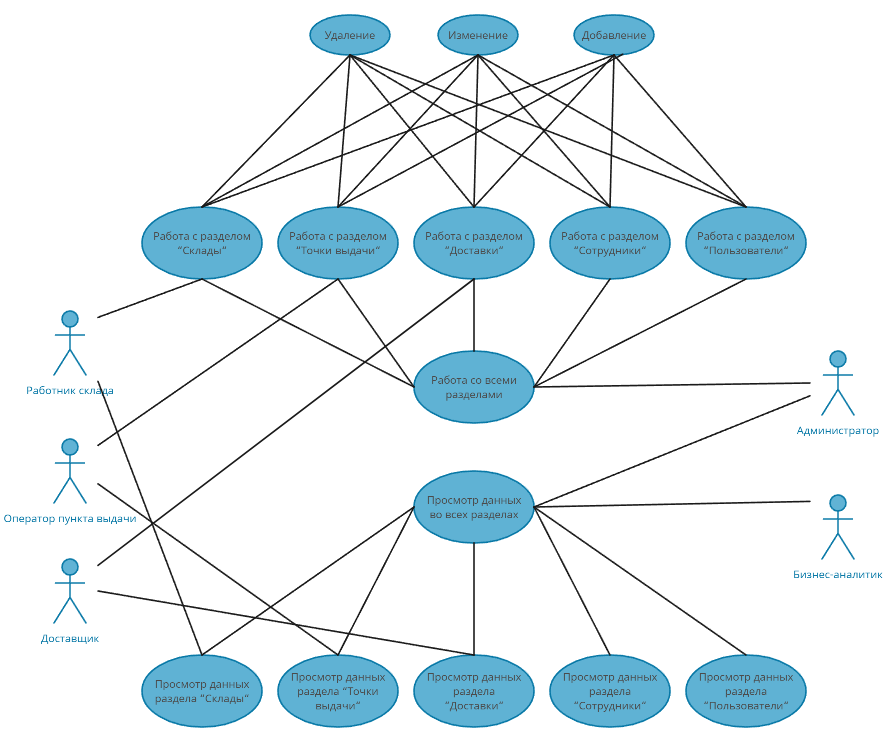


Рисунок 1 – Use Case диаграмма.

Бизнес-правила

БП-1 Доступ пользователей к информационной системе.

БП-1.1 Доступ пользователям к информационной системе даёт администратор путём создания пользователя и выдаче логина с паролем.

БП-2 Цветовая гамма.

БП-2.1 Допустимые цвета: основной цвет бирюзово-синий (#3b7475), дополнительный цвет ярко-бирюзовый (#00DEE1), цвет фона дымчато-белый (#EFEFEF), цвет текста на светлом фоне чёрный, цвет текста на тёмном фоне белый.

Детальные спецификации

ДС-1 Окно входа в систему.

ДС-1.1 Окно входа в систему содержит поля ввода «Логин» и «Пароль» и кнопку «Войти».

ДС-1.2 При неправильном вводе логина или пароля отображается сообщение «\* Неверный логин или пароль».

ДС-1.3 Если сотрудник ввёл корректные данные для входа, то окно входа в систему закрывается и открывается главное окно приложения.

ДС-2 Главное окно приложения.

ДС-2.1 В центральной части окна находится рабочая область.

ДС-2.2 Слева находится меню с разделами приложения. (см. ДС-3)

ДС-3 Меню разделов.

ДС-3.1 Меню разделов располагается слева и представляет собой список элементов.

ДС-3.2 При выборе раздела из меню, в рабочей области открывается соответствующий раздел.

ДС-3.3 Разделы к которым у сотрудника доступа нет при открытии отображают сообщение «Раздел не выбран или не доступен».

ДС-3.4 Раздел «Сотрудники» содержит в себе подразделы «Общие сведения» и «Пароли».

ДС-3.5 Раздел «Пользователи» содержит в себе подраздел «Общие сведения».

ДС-3.6 Раздел «Доставки» содержит в себе подраздел «Общие сведения».

ДС-3.7 Раздел «Склады» содержит в себе подраздел «Общие сведения».

ДС-3.8 Раздел «Пункты выдачи» содержит в себе подразделы «Общие сведения» и «Сотрудники в точках выдачи».

ДС-4 Рабочая область.

ДС-4.1 Рабочая область располагается в центральной части окна приложения.

ДС-4.2 Рабочая область содержит в себе открытый раздел или сообщение «Раздел не выбран или не доступен».

ДС-5 Раздел приложения.

ДС-5.1 В открытом разделе приложения сверху имеется название раздела, а под ним кнопки перехода в подразделы.

ДС-5.2 При открытии раздела по умолчанию выбирается первый подраздел.

ДС-6 Подраздел приложения.

ДС-6.1 В открытом подразделе приложения слева-сверху имеется строка фильтрации записи.

ДС-6.2 Табличный подраздел.

ДС-6.2.1 Табличный подраздел содержит в себе таблицу по центру.

ДС-6.2.2 В табличном подразделе есть кнопки «Добавить новую запись», «В PDF».

ДС-6.2.3 При нажатии на кнопку «Добавить новую запись» открывается окно с возможностью добавить запись.

ДС-6.2.4 При нажатии на кнопку «В PDF» открывается окно сохранения файла.

ДС-6.2.5 В таблице каждая запись содержит кнопки «Изменить» и «Удалить».

ДС-6.2.6 При нажатии на кнопку «Изменить» открывается окно с возможностью изменения записи.

ДС-6.2.7 При нажатии на кнопку «Удалить» удаляется соответствующая запись.

ДС-6.3 Карточный подраздел.

ДС-6.3.1 Карточный подраздел содержит карточки записей по центру.

ДС.6.3.2 Каждая карточка содержит кнопку для изменения данных.

ДС 7 Окно изменения и добавления записи.

ДС-7.1 Окно изменения или добавления записи содержит данные, которые расположены в текстовых полях, кнопках, выпадающих списках.

ДС-7.2 Окно изменения или добавления записи содержит кнопки «Сохранить» и «Отмена».

ДС-7.2.1 При нажатии на кнопку «Сохранить» проверяется заполненность полей и сохраняются изменения с закрытием окна.

ДС-7.2.2 Если обязательные поля не заполнены, то откроется окно с ошибкой «Обязательные поля не заполнены!».

ДС-7.2.3 При нажатии на кнопку «Отмена» отменяются все сделанные в окне изменения и оно закрывается.

ДС-8 Раздел «Пользователи».

ДС-8.1 Подраздел «Общая информация» содержит список пользователей с информацией об них.

ДС-8.1.1 Таблица содержит поля: id, ФИО, телефон, адрес.

ДС-8.1.2 В подразделе есть возможность использовать функции удаления, изменения, просмотра и добавления данных.

ДС-9 Раздел «Сотрудники».

ДС-9.1 Подраздел «Общая информация» содержит список сотрудников с информацией об них.

ДС-9.1.1 Таблица содержит: id, ФИО, должность, телефон, адрес.

ДС-9.1.2 В подразделе есть возможность использовать функции удаления, изменения, просмотра и добавления данных.

ДС-9.2 Подраздел «Пароли» содержит карточки сотрудников с возможностью изменения пароля.

ДС-9.2.1 Каждая карточка состоит из id, изображения, ФИО и кнопки «Изменить пароль».

ДС-10 Раздел «Доставки».

ДС-10.1 Подраздел «Список доставок» содержит информацию о доставках.

ДС-10.1.1 Таблица содержит: id доставки, id отправителя, ФИО отправителя, id получателя, ФИО получателя, адрес отправления, адрес получения, цена.

ДС-10.1.2 В подразделе есть возможность использовать функции удаления, изменения, просмотра и добавления данных.

ДС-11 Раздел «Склады».

ДС-11.1 Подраздел «Общая информация» содержит информацию о складах.

ДС-11.1.1 Таблица содержит: id, номер, адрес.

ДС-11.1.2 В подразделе есть возможность использовать функции удаления, изменения, просмотра и добавления данных.

ДС-12 Раздел «Точки выдачи».

ДС-12.1 Подраздел «Общая информация» содержит информацию о точках выдачи товаров.

ДС-12.1.1 Таблица содержит id, адрес.

ДС-12.1.2 В подразделе есть возможность использовать функции удаления, изменения, просмотра и добавления данных.

ДС-12.2 Подраздел «Сотрудники в точках выдачи» содержит информацию о сотрудниках, работающих в точках выдачи.

ДС-12.2.1 Таблица содержит id, сотрудника, точку выдачи, смену.

ДС-12.2.2 В подразделе есть возможность использовать функции удаления, изменения, просмотра и добавления данных.

Атрибуты качества

АК-1 В полях с определённым типом данных не должно быть возможности вводить символы другого типа данных.

АК-2 Фильтрация таблиц в подразделе (см. ДС-6.1).

АК-2.1 Должна быть возможность фильтрации таблиц.

АК-2.2 Фильтрация происходит сразу по всем полям таблицы.

Требования к данным

ТД-1 Пароли

ТД-1.1 Пароли должны шифроваться алгоритмом шифрования SHA-256. В базе данных хранение паролей только в зашифрованном виде.

Ограничения

О-1 Доступ к данным.

О-1.1 У сотрудника не должны отображаться разделы, к которым у него нет доступа (см. БП-4).

О-1.2 При попытке совершить действие на которое нет прав у сотрудника должно открываться окно с ошибкой «Не удалось выполнить действие. Не достаточно прав.».

ПРИЛОЖЕНИЕ 2

Тестирование логики программы представлено в таблице 1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Описание | Входные данные | Ожидаемый результат | Полученный результат | Статус |
| 1 | Вход в систему с верными данными под учётной записью администратора. | Логин: admin  Пароль: admin123 | Выполнение входа в систему и открытие главного окна приложения с именем, изображением и ролью вошедшего сотрудника. Доступ ко всем разделам и подразделам. | Выполнение входа в систему и открытие главного окна приложения с именем, изображением и ролью вошедшего сотрудника. Доступ ко всем разделам и подразделам. | Пройден |
| 2 | Вход в систему с верными данными под учётной записью бизнес-аналитика. | Логин: qwerty  Пароль: admin123 | Выполнение входа в систему и открытие главного окна приложения с именем, изображением и ролью вошедшего сотрудника. Доступ для просмотра ко всем разделам и подразделам, кроме паролей. | Выполнение входа в систему и открытие главного окна приложения с именем, изображением и ролью вошедшего сотрудника. Доступ для просмотра ко всем разделам и подразделам, кроме паролей. | Пройден |
| 3 | Вход в систему с верными данными под учётной записью оператора пункта выдачи. | Логин: dobavil  Пароль: admin123 | Выполнение входа в систему и открытие главного окна приложения. Доступ только к разделам и подразделам, пунктов выдачи. | Выполнение входа в систему и открытие главного окна приложения. Доступ только к разделам и подразделам, пунктов выдачи. | Пройден |
| 4 | Попытка входа в систему с не верным паролем под учётной записью администратора. | Логин: admin  Пароль: Pass123! | Под полем пароля появляется надпись «\* Не верный логин или пароль». | Под полем пароля появляется надпись «\* Не верный логин или пароль». | Пройден |
| 5 | Попытка вход в систему с не верным логином и паролем. | Логин: qwerty11  Пароль: Qwerty1! | Под полем пароля появляется надпись «\* Не верный логин или пароль». | Под полем пароля появляется надпись «\* Не верный логин или пароль». | Пройден |
| 6 | Запуск приложения и попытка входа в систему с доступом к базе данных. | Логин: admin  Пароль: admin123 | Приложение запускается и открывается окно входа в систему. При нажатии кнопки «Войти» выполняется вход в систему с открытием главного окна приложения. | Приложение запускается и открывается окно входа в систему. При нажатии кнопки «Войти» выполняется вход в систему с открытием главного окна приложения. | Пройден |
| 7 | Запуск приложения и попытка входа в систему без доступа к базе данных. | Логин: admin  Пароль: admin123 | Приложение запускается и открывается окно входа в систему. При нажатии кнопки «Войти» открывается окно с сообщением об ошибке «Возникли проблемы с подключением к базе данных. Обратитесь к администратору» | Приложение запускается и открывается окно входа в систему. При нажатии кнопки «Войти» открывается окно с сообщением об ошибке «Возникли проблемы с подключением к базе данных. Обратитесь к администратору» | Пройден |
| 8 | Создание PDF документа в разделе «Сотрудники» подразделе «Общие сведения». |  | Открывается диалоговое окно с предложением назвать файл и сохранить. При сохранении, появляется файл, содержащий таблицу с данными. | Открывается диалоговое окно с предложением назвать файл и сохранить. При сохранении, появляется файл, содержащий таблицу с данными. | Пройден |
| 9 | Попытка удаления связанных данных. |  | При попытке удалить запись, открывается окно с описанием ошибки «Не удалось удалить запись. Возможно присутствуют связанные записи.». | При попытке удалить запись, открывается окно с описанием ошибки «Не удалось удалить запись. Возможно присутствуют связанные записи.». | Пройден |
| 10 | Фильтрация записей |  | Происходит фильтрация записей и отображаются только те записи, которые содержат в себе символы из строки фильтрации. | Происходит фильтрация записей и отображаются только те записи, которые содержат в себе символы из строки фильтрации. | Пройден |
| 11 | Выход из приложения. |  | Выход из приложения происходит при его закрытии. | Выход из приложения происходит при его закрытии. | Пройден |
| 12 | Попытка добавления записи без заполнения обязательных полей. | Обязательные поля пустые | Открывается окно с ошибкой и описанием «Обязательные поля не заполнены!». | Открывается окно с ошибкой и описанием «Обязательные поля не заполнены!». | Пройден |

ПРИЛОЖЕНИЕ 3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Описание | Действие | Ожидаемый результат | Полученный результат | Статус |
| 1 | Замена символов для скрытия пароля при входе в систему. | Открыть приложение и ввести символы в поле ввода пароля. | Вместо символов отображаются точки. | Вместо символов отображаются точки. | Пройден |
| 2 | Появление полосы прокрутки при заполнении таблицы в подразделе. | Зайти в раздел с большим количеством записей, например «Пользователи» «Общие сведения» или добавить больше записей. | Появление полосы прокрутки при заполнении таблицы. | Появление полосы прокрутки при заполнении таблицы. | Пройден |
| 3 | Изменение цвета кнопки при наведении. | Открыть приложение и в окне входа в систему навести мышку на кнопку «Войти». | Смена цвета кнопки при наведению. | Смена цвета кнопки при наведению. | Пройден |
| 4 | Расширение приложения на весь экран. | В главном окне приложения нажать на жёлтую кнопку. | Расширение приложения на весь экран. | Расширение приложения на весь экран. | Пройден |
| 5 | Исчезновение надписи «Поиск...» при вводе символов в строку поиска. | Перейти в любой раздел, нажать на строку писка и начать вводить любые символы. | Исчезновение надписи «Поиск...». | Исчезновение надписи «Поиск...». | Пройден |
| 6 | Отображение фокуса на окне в виде тени. | Открыть любой раздел приложения и нажать на кнопу добавления записи. | Убирается тень с главного окна и появляется на окне добавления записи. | Убирается тень с главного окна и появляется на окне добавления записи. | Пройден |
| 7 | Возможность ввода любых символов, кроме цифр в поле ввода, предназначенное только для цифр. | Открыть раздел «Сотрудники», нажать на изменение любой записи. Попытаться ввести символы в поле ввода номера телефона. | В поле вводятся только цифры. | В поле вводятся только цифры. | Пройден |