## Аннотация

## Программа для туристического агентства «Греча» предназначена для оформления клиентов менеджерами агентства на различные туры. Программа позволяет менеджерам редактировать, удалять и добавлять клиентов, заметки, туры.

## 

**Содержание**

**Введение**……………………………………...……………………………………………………4

**1 Назначение и область применения**…………………………………………………………..5

**2 Постановка задачи**……………………………………………………………………………..6

**3 Описание программы**………………………………………………………………………….7

**3.1 Общие сведения**…………………………………………………………………………….7

**3.2 Функциональное назначение**……………………………………………………………11

**3.3 Описание логической структуры**……………………………………………………….11

**3.4 Входные и выходные данные**……………………………………………………………15

**4 Тестирование приложения**…………………………………………………………………..17

**4.1 Методы испытаний**……………………………………………………………………….17

**4.2 Тест-кейсы**…………………………………………………………………………………18

**5 Руководство пользователя**…………………………………………………………………..21

**5.1 Работа с приложением**……………………………………………………………………21

**5.2 Сообщение пользователю**………………………………………………………………..22

**6 Мероприятия по информационной безопасности**………………………………………...23

**6.1 Возможные угрозы информационной безопасности**…………………………………23

**6.2 Предотвращение угроз информационной безопасности**……………………………..23

**Приложение А**…………………………………………………………………………………...27

**Источники, использованные при разработке**……………………………………………….31

## 

**Введение**

В современном мире туристическая отрасль активно развивается, предлагая своим клиентам разнообразные туры и услуги. Важной составляющей успешной работы туристических агентств является эффективное управление данными о клиентах, турах и сопутствующих услугах. Для упрощения и автоматизации этих процессов разработано приложение «Греча».

Приложение «Греча» предназначено для регистрации клиентов на различные туры, оформления путевок и обеспечения высокой степени организации данных. Оно позволяет туристическим агентствам и компаниям быстро и эффективно обрабатывать информацию о клиентах, турах и услугах, что значительно упрощает их повседневную работу и повышает качество обслуживания клиентов.

Целью разработки данного программного обеспечения является создание удобного инструмента для управления, хранения, добавления и редактирования данных в туристической сфере. Приложение предоставляет пользователям возможность организовывать туры, оформлять страховки, а также оказывать услуги людям с ограниченными возможностями, обеспечивая тем самым комплексный подход к обслуживанию клиентов.

Приложение будет полезно различным организациям и компаниям, занимающимся туризмом, таким как туристические агентства и сервисы для путешествий, благодаря своей способности автоматизировать и оптимизировать процессы управления данными.

1. **Назначение и область применения**

Приложение "Греча" предназначено для автоматизации процессов управления турами, оформления путевок и обеспечения высокой степени организации данных для туристических агентств и компаний. Программа предоставляет удобный интерфейс для менеджеров, упрощая работу с данными о турах, клиентах и предоставляемых услугах.

Характеристики области применения:

* Организация туров, включая создание, редактирование и удаление туров;
* Оформление новых клиентов, а также редактирование и удаление данных уже имеющихся клиентов в базе;
* Оформление медицинских страховок и страховок жизни для клиентов;
* Предоставление услуг для людей с ограниченными возможностями, таких как медицинское обслуживание и помощь;
* Поддержка нескольких групп пользователей с различными уровнями доступа;
* Автоматизация процесса регистрации и учета клиентов на различные туры;
* Интерактивный интерфейс для управления базой данных клиентов и туров;
* Возможность интеграции с внешними сервисами для улучшения качества обслуживания;
* Обеспечение надежного хранения и защиты данных пользователей, включая шифрование и использование системы логинов и паролей.

1. **Постановка задачи**

## Программа "Греча" разработана для автоматизации процессов управления турами, регистрации клиентов и оформления сопутствующих услуг.

## Программа обладает следующими функциональными характеристиками:

* Регистрация и аутентификация пользователей: Менеджеры могут регистрироваться в системе, предоставляя свои персональные данные, и идентифицироваться для доступа к функционалу;
* Создание и редактирование туров: Менеджеры могут создавать новые туры, редактировать существующие туры и удалять устаревшие или неактуальные туры;
* Добавление и редактирование данных клиентов: Менеджеры могут добавлять новых клиентов в базу данных, редактировать данные уже существующих клиентов и удалять данные клиентов по мере необходимости;
* Оформление туров для клиентов: Клиенты могут выбирать туры, получать подтверждение от менеджера по поводу своих заявок;
* Оформление страховок: Программа позволяет оформлять различные виды страховок, такие как медицинская страховка, страховка жизни и др;
* Предоставление услуг для людей с ограниченными возможностями: Программа поддерживает оформление специализированных услуг для людей с ограниченными возможностями, включая медицинское обслуживание и помощь в путешествиях;
* Контроль и валидация данных: Программа обеспечивает валидацию входных данных для предотвращения ошибок ввода и повышения точности информации;
* Разграничение доступа: Поддержка нескольких групп пользователей с различными уровнями доступа, таких как администраторы и менеджеры;
* Интерактивный интерфейс: Программа предоставляет удобный и интуитивно понятный интерфейс для управления базой данных клиентов, туров, страховок и заметок;
* Защита данных: Обеспечение надежного хранения и защиты данных пользователей с использованием шифрования и системы логинов и паролей;
* Интеграция с внешними сервисами: Возможность интеграции с внешними сервисами для улучшения качества обслуживания и расширения функциональности приложения.

**3 Описание программы**

**3.1 Общие сведения**

Наименование программы: «Греча»

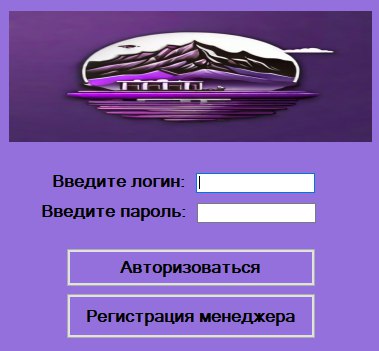
Требования:

Операционная система Windows (7 и выше);

Система управления базами данных Microsoft SQL Server;

Программа написана на языке C# в среде разработки Visual Studio.

Окна программы на рисунках (1-3).



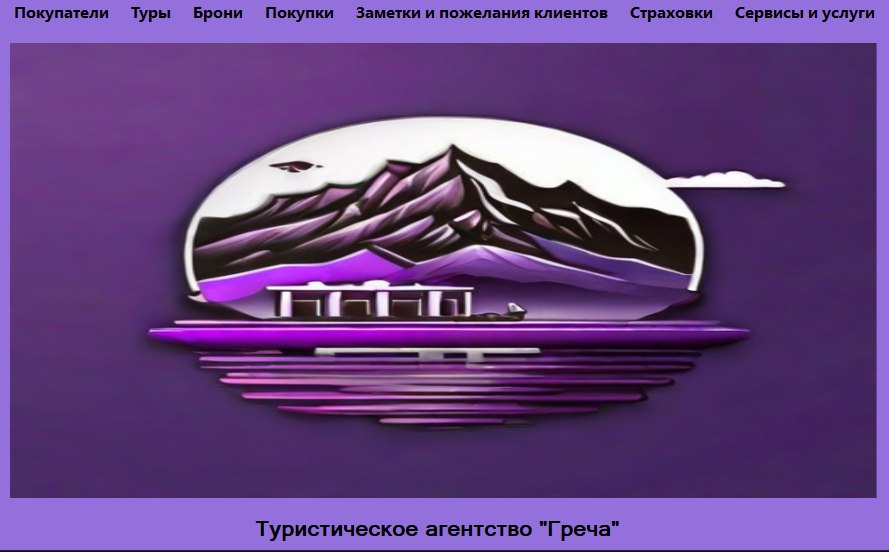
**Рис. 1** - Окно авторизации

Окно авторизации позволяет зайти в приложение уже зарегистрированных пользователей, вводя свой логин и пароль. Данный процесс необходим для обеспечения безопасности и индивидуального доступа к функционалу.



**Рис. 2** - Окно регистрации

В окне регистрации можно регистрировать новых менеджеров, указав ФИО, логин и пароль. Этот процесс позволяет новым менеджерам зарегистрироваться в системе и получить доступ ко всем возможностям приложения.



**Рис. 3 -** Главное окно пользователя

Элементы интерфейса на рисунке 3 предназначены для следующего функционала:

* Список клиентов: Менеджеры могут просматривать список клиентов и редактировать его;
* Список туров: Менеджеры могут просматривать список туров, редактировать его, удалять и добавлять записи;
* Список броней и покупок: Менеджеры могут просматривать, редактировать, удалять и отменять записи броней клиентов;
* Список заметок и пожелания клиентов: Менеджеры могут добавлять и удалять записи заметок и пожеланий клиентов для дальнейшего развития функционала приложения;
* Список страховок: Менеджеры могут добавлять и удалять записи в таблицу страховок;
* Сервисы и услуги: Менеджеры могут добавлять, удалять и редактировать записи в таблицах сервисов и услуг для людей с ограниченными возможностями.

**3.2 Функциональное назначение**

В целях обеспечения безопасности и эффективности использования нашей системы, установлено несколько функциональных ограничений.

* Уникальные имена пользователей: Имена пользователей должны быть уникальными в системе. При регистрации нового пользователя система должна проверить, не занято ли уже выбранное имя.
* Доступ к данным: Каждый пользователь имеет доступ только к собственным данным и данным, к которым ему был предоставлен доступ.

**3.3 Описание логической структуры**

**Алгоритм программы «Греча»**

**Регистрация и аутентификация пользователей:**

* Пользователь вводит свои персональные данные для регистрации;
* Система проверяет корректность введенных данных;
* При успешной регистрации данные сохраняются в базе данных;
* Менеджер авторизуется с использованием логина и пароля, указанными при регистрации.

**Страница с покупателями:**

* После авторизации менеджер может перейти на страницу с покупателями;
* Менеджер переходит к редактированию таблицы с клиентами.

**Страница с турами:**

* Менеджер переходит на страницу с турами;
* Менеджер может редактировать таблицу с турами, а все изменения сохраняются в базе данных.

**Страница с бронями и покупками:**

* Менеджер переходит на страницу с бронями и покупками;
* Менеджер может редактировать таблицу с бронями и покупками, изменения сохраняются в базе данных.

**Страница с заметками и пожеланиями клиентов:**

* Менеджер переходит на страницу с заметками и пожеланиями клиентов;
* Менеджер может редактировать таблицу, изменения сохраняются в базе данных.

**Страница страховок:**

* Менеджер переходит на страницу со страховками для дальнейшего редактирования;
* Менеджер может редактировать таблицу, изменения сохраняются в базе данных.

**Страница сервисов и услуг:**

* Менеджер переходит на страницу сервисов для клиентов для редактирования;
* Менеджер может редактировать таблицу, сохраняя изменения.

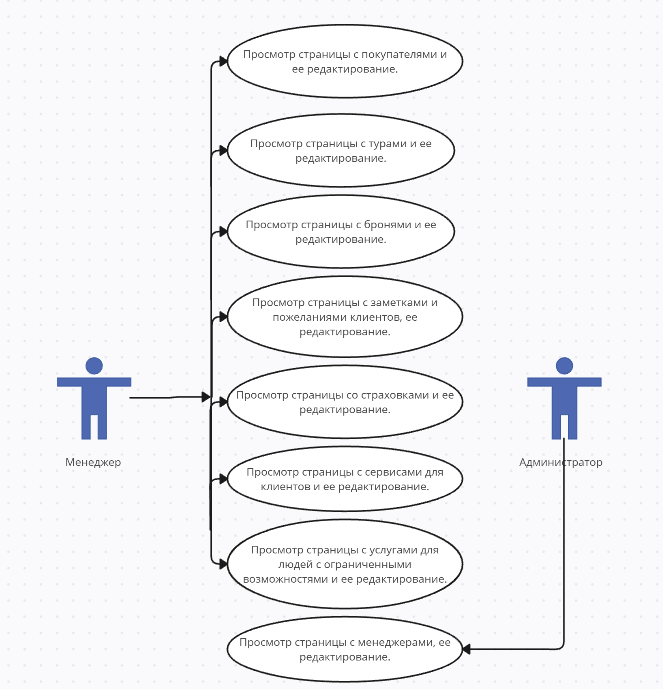
**Страница услуг для людей с ограниченными возможностями:**

* Менеджер переходит на страницу услуг, предназначенную для людей с ограниченными возможностями;
* Менеджер может редактировать данную таблицу, сохраняя изменения.

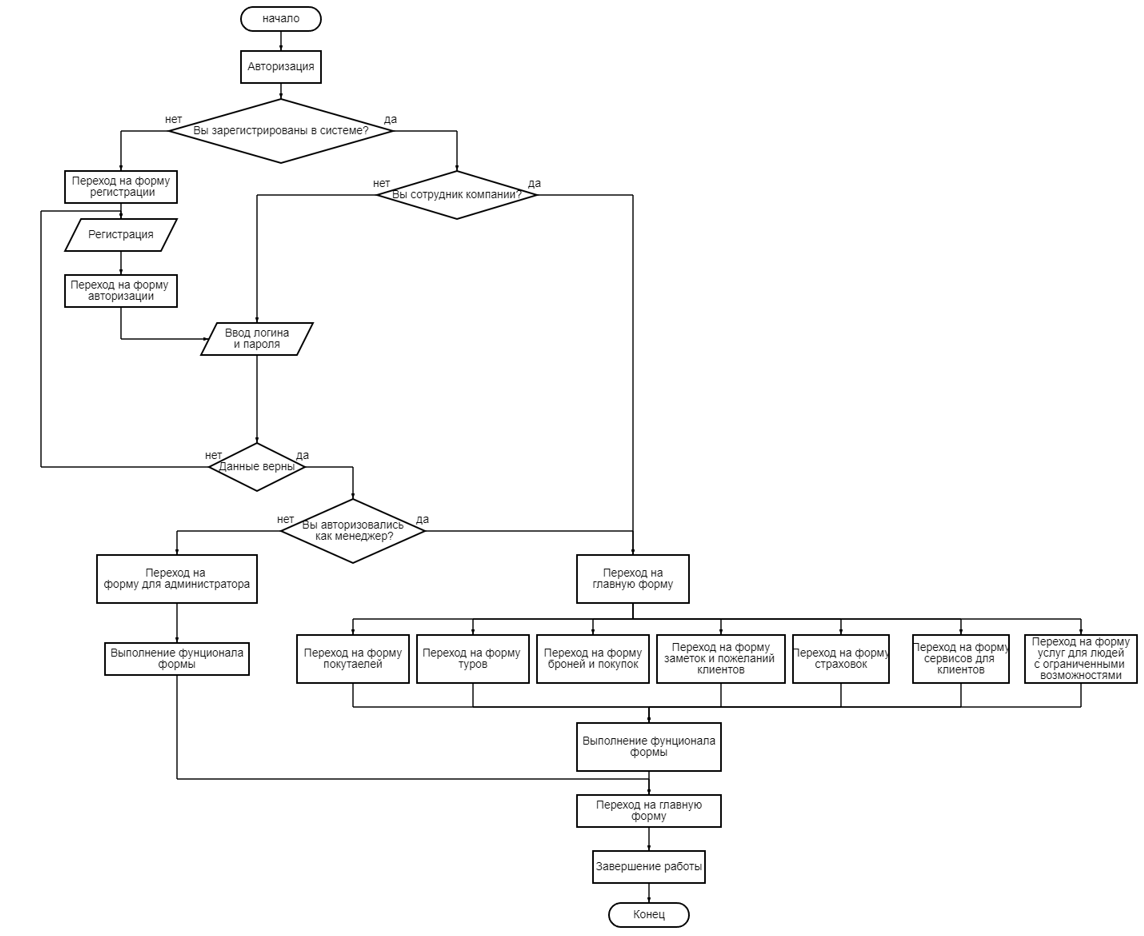
**Связи программы с другими программами**

**База данных (Database):**

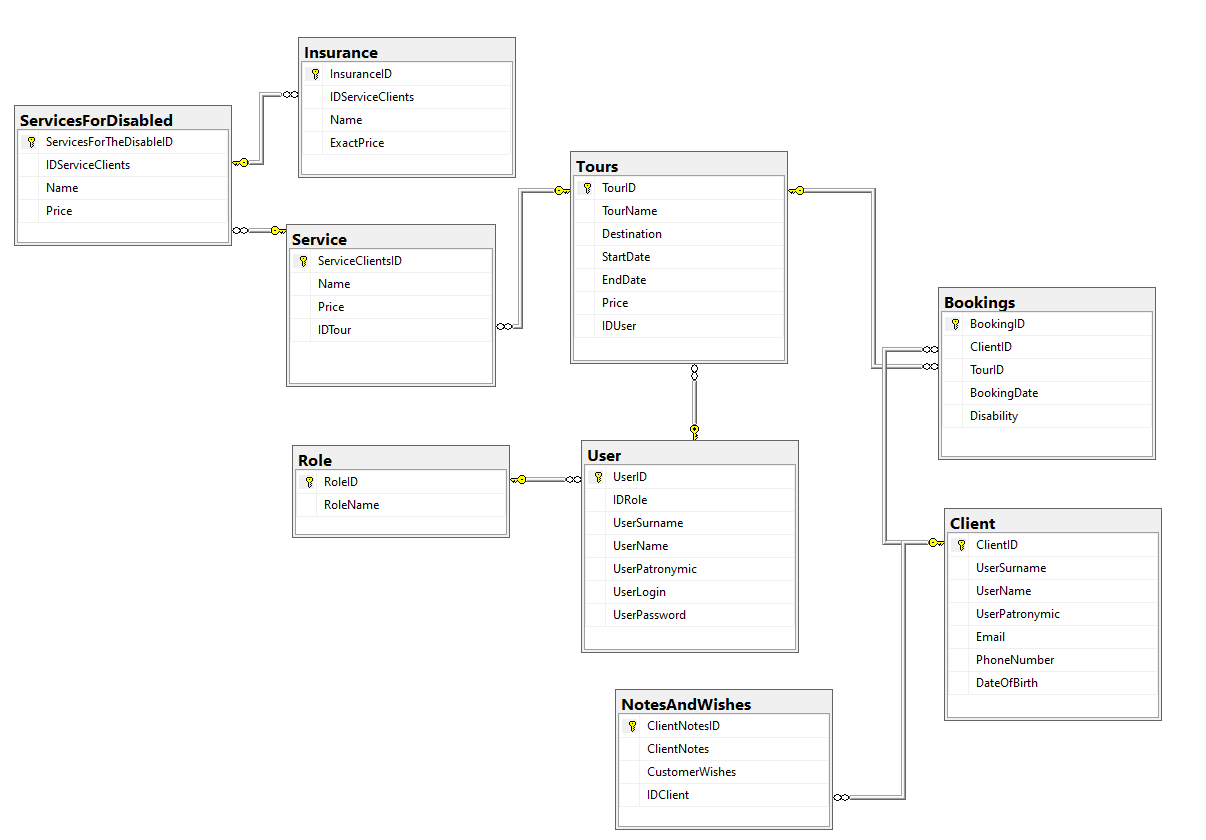
* Программа взаимодействует с базой данных для хранения и получения информации о пользователях, турах, броней, страховок, заметок и пожеланий клиентов, сервисов и услуг;
* Все данные, введенные менеджерами, сохраняются в базе данных;
* Менеджеры могут извлекать данные из базы данных для оформления туров, броней, заметок, сервисов и услуг и их редактирования.



**Рис. 4** – Диаграмма вариантов



**Рис. 5** – Диаграмма деятельности



**Рис. 6** – Схема базы данных

**3.4 Входные и выходные данные**

**Входные данные:**

**Регистрация и аутентификация пользователей:**

* Пользователь вводит свои персональные данные для регистрации;
* Система проверяет корректность введенных данных;
* При успешной регистрации данные сохраняются в базе данных;
* Менеджер авторизуется с использованием логина и пароля, указанными при регистрации.

**Страница с покупателями:**

* После авторизации менеджер может перейти на страницу с покупателями;
* Менеджер переходит к редактированию таблицы с клиентами;
* Менеджер может удалить клиента из базы данных;
* Менеджер может добавить нового клиента, введя его ФИО, email, номер телефона и дату рождения, после чего обновить данные в таблице.

**Страница с турами:**

* Менеджер переходит к редактированию туров;
* Менеджер может удалить тур из базы данных, если он уже не актуален;
* Менеджер может добавить новый тур, введя данные о нем и сохраняя его в базе данных.

**Страница с бронями и покупками:**

* Менеджер переходит к редактированию таблицы с забронированными турами клиентов;
* Менеджер может удалить бронь клиента из базы данных;
* Менеджер может добавить бронь для определенного клиента, выбрав определенного клиента по фамилии и номеру тура, выбрать дату и подтвердить инвалидность клиента, если он таковым является;
* Менеджер может экспортировать таблицу с клиентами в Excel для добавления заметок и дальнейшей печати.

**Страница с заметками и пожеланиями клиентов:**

* Менеджер переходит к редактированию страницы с заметками и пожеланиями клиентов;
* Менеджер может удалить записи из базы данных;
* Менеджер может добавить запись, введя необходимую заметки и пожелание клиента/клиентов.

**Страница страховок:**

* Менеджер переходит на страницу со страховками для дальнейшего редактирования;
* Менеджер может удалить запись;
* Менеджер может добавить запись, введя наименование страховки и ее цену.

**Страница сервисов и услуг:**

* Менеджер переходит на страницу сервисов для клиентов для редактирования;
* Менеджер может удалить запись;
* Менеджер может добавить запись в базу данных, введя необходимое название сервиса и его цену.

**Страница услуг для людей с ограниченными возможностями:**

* Менеджер переходит на страницу услуг, предназначенную для людей с ограниченными возможностями;
* Менеджер может удалить запись из базы данных;
* Менеджер может добавить запись в базу данных, введя необходимое название услуги и ее цену.

**4 Тестирование приложения**

**4.1 Методы испытаний**

**Тестирование регистрации и авторизации пользователей:**

Испытания проводятся для проверки правильности регистрации новых пользователей с различными вариантами данных, включая валидные и невалидные. Это включает в себя проверку корректности введенных данных (ФИО, адрес электронной почты, логин, пароль), а также проверку возможности авторизации существующих пользователей с использованием правильных и неправильных учетных данных.

**Тестирование добавления новых туров в систему:**

Испытания проводятся для проверки возможности добавления новых туров в систему. Это включает в себя проверку корректного сохранения введенных данных и их отображения.

**4.2 Тест-кейсы**

Таблица 1 – Тест – кейс 01.01 «Тестирование регистрации и авторизации пользователей» (страница 19-20).

|  |  |  |  |
| --- | --- | --- | --- |
| Номер тест-кейса | 01.01 | | |
| Название: | Тестирование регистрации и авторизации пользователей | | |
| Действие | | Ожидаемый результат | Результат теста:  пройден |
| Предусловие: | | | |
| Приложение запущено, сервер с базой данных доступен. | | Приложение запущено без ошибок. Сервер с базой данных доступен. | Пройден |
| Шаги теста: | | | |
| Попытка зарегистрировать нового пользователя с корректными данными (ФИО, правильный формат адреса электронной почты, логин, пароль). | | При регистрации нового пользователя система успешно создает новую учетную запись и перенаправляет пользователя на главную страницу приложения. | Пройден |
| Попытка выполнить авторизацию с использованием учетных данных вновь зарегистрированного пользователя. | | При авторизации с использованием учетных данных зарегистрированного пользователя система успешно авторизует пользователя и перенаправляет его на главную страницу приложения. | Пройден |
| Постусловие: | | | |
| Закрыть приложение | | Приложение закрыто | Пройден |

Таблица 2 – тест-кейс 02.01 «Тестирование добавления новых туров в систему» (страница 20-21).

|  |  |  |  |
| --- | --- | --- | --- |
| Номер тест-кейса | 02.01 | | |
| Название: | Тестирование добавления новых туров в систему | | |
| Действие | | Ожидаемый результат | Результат теста:  пройден |
| Предусловие: | | | |
| Приложение запущено, сервер с базой данных доступен. | | Приложение запущено без ошибок. Сервер с базой данных доступен. | Пройден |
| Шаги теста: | | | |
| Попытка перехода по кнопке «Добавить» для добавления нового тура. | | Переход на форму добавления нового тура | Пройден |
| Попытка добавления нового тура в систему, введя корректные данные (название тура, направление, дата начала, дата окончания, цена) и сохранить тур нажав на кнопку «Сохранить» | | При успешном вводе и сохранении данных пользователь получит сообщение «Сохранение успешно!» | Пройден |
| Постусловие: | | | |
| Закрыть приложение | | Приложение закрыто | Пройден |

**5 Руководство пользователя**

**5.1 Работа с приложением**

**Загрузка и запуск программы:**

* Пользователь запускает приложение на своем устройстве;
* Приложение загружается и отображает экран с авторизацией и регистрацией.

**Регистрация нового пользователя:**

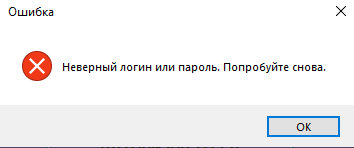
* Пользователь выбирает опцию "Регистрация";
* Пользователь вводит свои данные: ФИО, адрес электронной почты, логин, пароль;
* Приложение проверяет введенные данные на корректность и уникальность;
* При успешной регистрации пользователь получает подтверждение и может выполнить вход.

**Вход пользователя:**

* Пользователь вводит логин и пароль и нажимает на кнопку авторизоваться;
* Приложение проверяет правильность введенных данных;
* При успешной аутентификации менеджер перенаправляется на главный экран приложения, а администратор переходит на окно с таблицей для редактирования пользователей приложения (менеджеров).

**5.2 Сообщение пользователю**

**Неверный ввод логина или пароля**: «Ошибка. Неверный логин или пароль. Попробуйте снова.» (рис. 7)



**Рис. 7** – Сообщение об ошибке

**6 Мероприятия по информационной безопасности**

**6.1 Возможные угрозы информационной безопасности:**

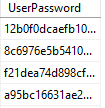
Для обеспечения безопасности информации в приложении "Греча" необходимо учитывать следующие возможные угрозы:

* Кража учетных данных;
* Вредоносные программы и вирусы;
* Несанкционированный доступ к данным;
* Подбор чужого пароля.

**6.2 Предотвращение угроз информационной безопасности:**

**Шифрование пароля:**

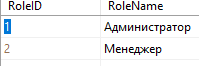
Для повышения безопасности паролей пользователей реализуется механизм шифрования. При регистрации нового пользователя его пароль хешируется по алгоритму SHA-256 перед тем, как происходит сохранение новой записи в базу данных. Это обеспечивает защиту от несанкционированного доступа к хранимым паролям (рис. 7).

****

**Рис. 8 –** Примершифрования паролей пользователей

**Разграничение прав:**

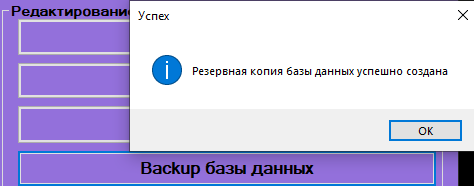
Пользователям назначаются минимально необходимые права для выполнения операций в системе. Это уменьшает риски несанкционированного доступа и снижает возможные негативные последствия в случае компрометации аккаунта (рис. 9).



**Рис. 9** – Разграничение прав

**Резервное копирование базы данных:**

Для обеспечения сохранности и доступности критически важной информации, менеджеры компании могут на регулярной основе выполнять резервное копирование базы данных. Это мероприятие направлено на защиту данных от потерь, вызванных различными угрозами, включая технические сбои, ошибки пользователей, вредоносные атаки и прочие форс-мажорные обстоятельства (рис.10).



**Рис. 10** –Резервное копирование базы данных

**Заключение**

В рамках данного дипломного проекта было разработано приложение "Греча", предназначенное для управления процессом организации туров и оформления путевок для клиентов туристических агентств. Процесс разработки включал анализ требований, проектирование функциональности, программную реализацию и тестирование.

В программе реализованы следующие функции:

* Регистрация и авторизация пользователей с использованием надежного механизма шифрования паролей;
* Создание, редактирование и удаление туров, а также оформление туров для клиентов;
* Добавление новых клиентов в базу данных, редактирование и удаление данных существующих клиентов;
* Оформление медицинских страховок и страховок жизни для клиентов;
* Предоставление услуг для людей с ограниченными возможностями, включая медицинское обслуживание и помощь в путешествиях;
* Система уведомлений для информирования клиентов и администраторов о важных событиях и изменениях в турах;
* Интерактивный интерфейс для управления базой данных клиентов и туров.

Для обеспечения информационной безопасности приложения были приняты следующие меры:

* Реализация механизма шифрования паролей и скрытие ввода пароля для защиты учетных данных пользователей;
* Применение надежного механизма аутентификации пользователей с ограничением прав доступа для предотвращения несанкционированного доступа;
* Резервное копирование базы данных.

В результате успешной реализации проекта было создано удобное и безопасное приложение, которое облегчает процесс организации туров и оформления путевок для клиентов туристических агентств. Дальнейшее развитие приложения планируется с учетом обратной связи пользователей и актуальных требований в туристической отрасли.

**Приложение А**

## Текст программы

using System;

using System.Data.SqlClient;

using System.Security.Cryptography;

using System.Text;

using System.Windows.Forms;

namespace grecha

{

public partial class RegistManager : Form

{

private const string connectionString = "Data Source=DESKTOP-R934923\\SQLEXPRESS;Initial Catalog=tyr;Integrated Security=True";

public RegistManager()

{

InitializeComponent();

this.StartPosition = FormStartPosition.CenterScreen;

this.FormClosing += RegistManager\_FormClosing;

}

private void btnRegistManager\_Click(object sender, EventArgs e)

{

string surname = txtSurname.Text;

string name = txtName.Text;

string patronymic = txtPatronymic.Text;

string login = txtLogin.Text;

string password = txtPassword.Text;

// Проверка наличия логина и пароля

if (string.IsNullOrWhiteSpace(login) || string.IsNullOrWhiteSpace(password))

{

MessageBox.Show("Логин и пароль обязательны для заполнения.", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

// Проверка наличия логина в базе данных

if (IsLoginExists(login))

{

MessageBox.Show("Данный логин уже занят, выберите другой.");

return;

}

// Хэширование пароля

string hashedPassword = HashPassword(password);

// Регистрация нового пользователя

RegisterUser(surname, name, patronymic, login, hashedPassword);

MessageBox.Show("Регистрация прошла успешно!");

Main main = new Main();

main.Show();

this.Hide();

}

private bool IsLoginExists(string login)

{

bool result = false;

using (SqlConnection connection = new SqlConnection(connectionString))

{

string query = "SELECT COUNT(\*) FROM [Usеr] WHERE UserLogin = @Login";

using (SqlCommand command = new SqlCommand(query, connection))

{

command.Parameters.AddWithValue("@Login", login);

connection.Open();

int count = (int)command.ExecuteScalar();

if (count > 0)

{

result = true;

}

}

}

return result;

}

private void RegisterUser(string surname, string name, string patronymic, string login, string password)

{

using (SqlConnection connection = new SqlConnection(connectionString))

{

string query = "INSERT INTO [Usеr] (UserSurname, UserName, UserPatronymic, UserLogin, UserPassword, IDRole) VALUES (@Surname, @Name, @Patronymic, @Login, @Password, @IDRole)";

using (SqlCommand command = new SqlCommand(query, connection))

{

command.Parameters.AddWithValue("@Surname", surname);

command.Parameters.AddWithValue("@Name", name);

command.Parameters.AddWithValue("@Patronymic", patronymic);

command.Parameters.AddWithValue("@Login", login);

command.Parameters.AddWithValue("@Password", password);

command.Parameters.AddWithValue("@IDRole", 2);

connection.Open();

command.ExecuteNonQuery();

}

}

}

private void button2\_Click(object sender, EventArgs e)

{

Form1 form1 = new Form1();

this.Hide();

form1.Show();

}

private void RegistManager\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

private string HashPassword(string password)

{

using (SHA256 sha256 = SHA256.Create())

{

byte[] bytes = sha256.ComputeHash(Encoding.UTF8.GetBytes(password));

StringBuilder builder = new StringBuilder();

for (int i = 0; i < 16; i++) // Используем первые 16 байт (32 символа)

{

builder.Append(bytes[i].ToString("x2"));

}

return builder.ToString();

}

}

}

}

**Источники, использованные при разработке**

ГОСТ Р ИСО\_МЭК 25051-2017 Требования к качеству готового к использованию программного продукта (RUSP) и инструкции по тестированию.

ГОСТ Р 2.105–2019 Общие требования к текстовым документам.

ГОСТ Р 59795–2021 Комплекс стандартов на автоматизированные системы.

ЕСПД Единая система программной документации.

**Internet – ресурсы**

* MySQL [Электронный ресурс] - <https://www.mysql.com/>
* GitHub [https://github.com/] Онлайн-платформа для хостинга и совместной разработки проектов. Позволяет отслеживать изменения в коде.
* Microsoft Documentation [https://docs.microsoft.com/] Официальная документация от Microsoft. Использовалась для изучения технических деталей, API и рекомендаций по разработке на платформе .NET.