## Аннотация

Программный продукт служит для упрощенной работы с большим количеством данных. Для наглядности будут представлены структурная схема, алгоритм работы программы. А так же дана краткая характеристика аппаратных и программных средств.

**Оглавление**

Аннотация……………………………………………………………………………………………2

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

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

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

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

4.Программа и методика испытаний….…………………………………………………………..11

5.Руководство пользователя (оператор)...………………………………………………………...13

Заключение………………………………………………………………………………………….14

Приложения…………………………………………………………………………………………15

# Введение

Система базы данных необходима для управления и фильтрацией большого количество информации. Также необходимо разделить права доступа для 3-х пользователей, а именно: член экипажа самолета (может видеть других членов экипажей, а также роли каждого из членов), механик (может видеть самолеты и привязанных к ним механиков, а также другую информацию о самолетах), координатор (может видеть предстоящие рейсы и базовую информацию о них).

Данное разграничение позволит каждому пользователю видеть только необходимую для него информацию.

Система базы данных создается для достижения простоты в использовании большого количества информации с минимальными затратами времени и с отзывчивым, приятным интерфейсом.

.

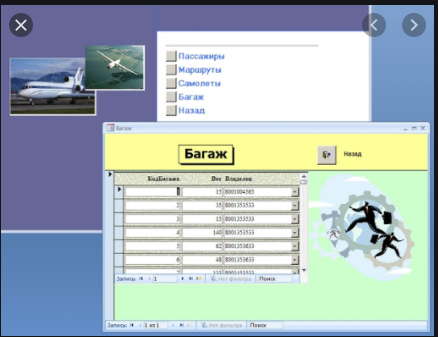
1. **Назначение и область применения**

Система базы данных аэропорта разрабатывается для удобного использования данных, включающих в себя: рейсы, команды на борту самолетов, механики, отвечающие за работоспособность самолетов и т.д. Данную разработку можно использовать в любом аэропорте. Программа не требует никаких навыков в пользовании.

Программа рассчитана для внедрения на компьютерах версии Windows 8.1 и выше.

Аналогами данного программного продукта являются:

1. База данных «Аэропорт».



Работа приложения

1. **Постановка задачи**

Основной задачей является создание исправно работающего приложения. В базе данных находятся несколько таблиц с необходимыми данными. Доступ к данным имеет ограниченное количество пользователей. В приложении можно фильтровать данные.

1. **Описание программы**
   1. **Общие сведения**

Приложение «База данных учебного процесса школы» создана для упрощения в навигации по данным школы, а также возможности использования готовых запросов для фильтрации этих данных. Также в данном приложении должно быть организовано разделение уровней прав доступа.

Программа написана с использованием языка программирования C# 2017 и базы данных SQL server.

**Необходимые программные средства:**

**–** Операционная система Windows 10;

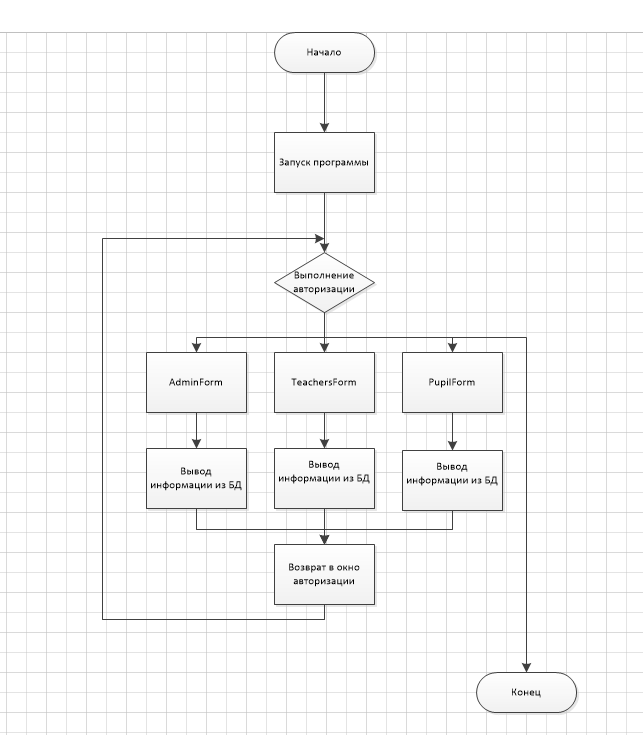
– Установленная Microsoft SQL Server Management Studio;

**–** Установленная Visual Studio 2017;

* 1. **Функциональное назначение**

В программе присутствуют 4 формы:

1. MainWindow. Запускается самой первой. В ней вводится данные для авторизации.
2. Engineer\_Window. Если данный пользователь является инженером-механиком, открывается данная форма. В ней можно выбрать предложенные варианты вывода информации из базы данных.
3. Coord\_Window. Если данный пользователь является координатором, открывается данная форма. В ней можно выбрать предложенные варианты вывода информации из базы данных.
4. Window\_Crews. Если данный пользователь является членом команды самолета, открывается данная форма. В ней можно выбрать предложенные варианты вывода информации из базы данных.
   1. **Описание логической структуры**

**

Логическая схема приложения

* 1. **Используемые технические средства**

Для корректного функционирования данной разработки необходимо наличие следующих технических средств:

* монитор;
* клавиатура;
* манипулятор типа «мышь».

Системные требования:

* тактовая частота процессора >2000 Гц;
* объем оперативной памяти 64 Мб;
* объем свободного дискового пространства 50 Мб;
* разрешение монитора 1024x768.
  1. **Вызов и загрузка**

Запуск программы производится файлом Praktika.exe. Данный файл запускает самое первое окно приложения «LoginForm». В нем происходит авторизация и дальнейшая работа программы.

* 1. **Входные и выходные данные**

Входные данные:

* Ввод пользователем логина и пароля.

Выходные данные:

* Данные таблицы базы данных «аэропорт»

**4.Программа и методика испытаний**

**4.1.Объекты испытаний**

База данных «аэропорт». Данное приложение можно использовать в аэропортах, для облегчения работы с большим объемом информации.

**4.2.Цель испытаний**

## Проверка работоспособности программы, корректного отображения информации, правильности работы разграничения доступа, а также отзывчивость интерфейса.

**4.3.Требования к программе**

* Разграничение прав доступа.
* Корректность вывода информации.
* Отзывчивость интерфейса.

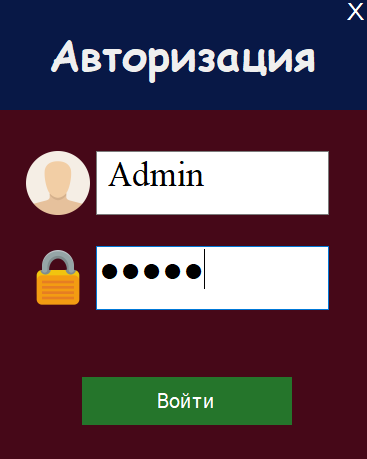
**4.4.Методы испытаний**

- Проверка разграничения прав доступа. Выполнялся вход с существующих профилей под разными паролями, чтобы удостоверится в правильности работы MainWindow.

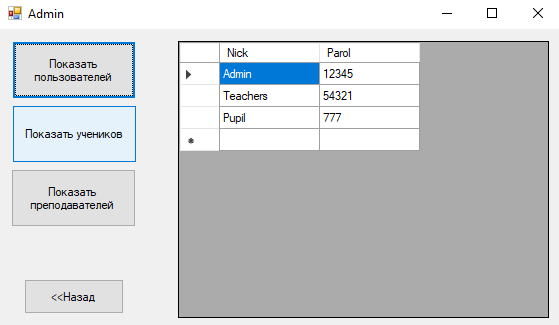
- Вывод информации по нажатию соответствующих кнопок совпадал с нужной информацией из БД.

- Интерфейс приложения был предложен на оценку 10 студентам, которые потвердели, что он: интуитивно понятный и приятный.

**4.5.Тестовый пример**

****

Проверка разграничения прав доступа



Проверка корректности вывода информации

**5.Руководство пользователя (оператора)**

**5.1.Выполнение программы**

При запуске программы пользователю необходимо ввести данные для авторизации (логин и пароль). В соответствии с введёнными данными пользователю отобразится соответствующая его уровню доступа информация. По нажатия кнопок на форме с таблицей, пользователь выполняет вызов запросов для фильтрации данных БД. По нажатию кнопки «Выйти», пользователь возвращается на первое окно. По нажатию на «крестик» выполняется завершение работы приложения.

**5.2.Сообщение оператору**

«Неверный пароль» - сообщение, которое выводится в случае ввода неправильного пароля пользователем.

«Пользователь не найден» - сообщение, которое выводится в случае ошибки в логине пользователем.

# Заключение

Все задачи, а также цель, которая стояла в начале разработки, успешно выполнены. В приложении организовано разграничение прав доступа, а также исправно работающий интерфейс, для упрощенного пользования большим количеством данных. Хотелось бы отметить, что не все задуманное было реализовано на практике. Стоило бы уделить время кросс-платформенности и добавить более обширный спектр возможностей.

# Приложения

## Текст программы

**LoginForm**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Praktika

{

public partial class LoginForm : Form

{

private readonly string sqlconnection = "Data Source = localhost; Initial Catalog = School; Integrated Security = True";

public LoginForm()

{

InitializeComponent();

this.passField.AutoSize = false;

this.passField.Size = new Size(this.passField.Size.Width, 64);

}

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

{

Application.Exit();

}

private void closeButton\_MouseEnter(object sender, EventArgs e)

{

closeButton.ForeColor = Color.Red;

}

private void closeButton\_MouseLeave(object sender, EventArgs e)

{

closeButton.ForeColor = Color.White;

}

Point lastPoint;

private void MainPanel\_MouseMove(object sender, MouseEventArgs e)

{

if(e.Button == MouseButtons.Left)

{

this.Left += e.X - lastPoint.X;

this.Top += e.Y - lastPoint.Y;

}

}

private void MainPanel\_MouseDown(object sender, MouseEventArgs e)

{

lastPoint = new Point(e.X, e.Y);

}

private void TopPanel\_MouseMove(object sender, MouseEventArgs e)

{

if (e.Button == MouseButtons.Left)

{

this.Left += e.X - lastPoint.X;

this.Top += e.Y - lastPoint.Y;

}

}

private void TopPanel\_MouseDown(object sender, MouseEventArgs e)

{

lastPoint = new Point(e.X, e.Y);

}

private void button1\_Click(object sender, EventArgs e)

{

string command = $"SELECT Parol FROM Users WHERE Nick = '{loginField.Text}'";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

SqlDataAdapter adapter;

adapter = new SqlDataAdapter(command, connection);

DataTable dt = new DataTable();

try

{

adapter.Fill(dt);

string pass = dt.Rows[0][0].ToString();

if (pass == passField.Text)

{

if (loginField.Text == "Admin")

{

Admin formAdmin = new Admin();

formAdmin.Show();

this.Hide();

}

else if (loginField.Text == "Pupil")

{

Pupil formPupil = new Pupil();

formPupil.Show();

this.Hide();

}

else if (loginField.Text == "Teachers")

{

Teacher formTeacher = new Teacher();

formTeacher.Show();

this.Hide();

}

}

else MessageBox.Show("Неверный пароль!");

}

catch

{

MessageBox.Show("Ошибка в логине!");

}

connection.Close();

}

}

}

}

**AdminForm**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace Praktika

{

public partial class Admin : Form

{

private readonly string sqlconnection = "Data Source = localhost; Initial Catalog = School; Integrated Security = True";

public Admin()

{

InitializeComponent();

}

private void Admin\_Load(object sender, EventArgs e)

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "schoolDataSet.Users". При необходимости она может быть перемещена или удалена.

//this.usersTableAdapter.Fill(this.schoolDataSet.Users);

// TODO: данная строка кода позволяет загрузить данные в таблицу "schoolDataSet.Class". При необходимости она может быть перемещена или удалена.

//this.classTableAdapter.Fill(this.schoolDataSet.Class);

}

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

{

//this.classTableAdapter.Fill(this.schoolDataSet.Class);

string command = "SELECT FIO, Letter FROM Students INNER JOIN Class ON Id\_Class = Class\_id;";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

DataTable dt = new DataTable();

SqlDataAdapter adapter = new SqlDataAdapter(command, connection);

adapter.Fill(dt);

dataGridView1.DataSource = dt.DefaultView;

connection.Close();

}

}

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

{

//this.usersTableAdapter.Fill(this.schoolDataSet.Users);

string command = "SELECT Nick, Parol FROM Users;";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

DataTable dt = new DataTable();

SqlDataAdapter adapter = new SqlDataAdapter(command, connection);

adapter.Fill(dt);

dataGridView1.DataSource = dt.DefaultView;

connection.Close();

}

}

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

{

string command = "SELECT TitleP, FIO FROM Predmet INNER JOIN Staff ON Id\_Staff = Staf\_id; ";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

DataTable dt = new DataTable();

SqlDataAdapter adapter = new SqlDataAdapter(command, connection);

adapter.Fill(dt);

dataGridView1.DataSource = dt.DefaultView;

connection.Close();

}

}

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

{

this.Close();

LoginForm formLogin = new LoginForm();

formLogin.Show();

}

}

}

**TeacherForm**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace Praktika

{

public partial class Teacher : Form

{

private readonly string sqlconnection = "Data Source = localhost; Initial Catalog = School; Integrated Security = True";

public Teacher()

{

InitializeComponent();

}

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

{

string command = "SELECT DayWeek, TitleP, Letter FROM Raspisanie INNER JOIN Class ON Id\_class = Class\_id INNER JOIN Predmet ON Id\_Predmet = Predmet\_id WHERE Letter = 'А' AND DayWeek = 'ПН';";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

DataTable dt = new DataTable();

SqlDataAdapter adapter = new SqlDataAdapter(command, connection);

adapter.Fill(dt);

dataGridView1.DataSource = dt.DefaultView;

connection.Close();

}

}

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

{

string command = "SELECT FIO, Letter FROM Students INNER JOIN Class ON Id\_Class = Class\_id;";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

DataTable dt = new DataTable();

SqlDataAdapter adapter = new SqlDataAdapter(command, connection);

adapter.Fill(dt);

dataGridView1.DataSource = dt.DefaultView;

connection.Close();

}

}

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

{

string command = "SELECT DayWeek, TitleP, Letter FROM Raspisanie INNER JOIN Class ON Id\_class = Class\_id INNER JOIN Predmet ON Id\_Predmet = Predmet\_id; ";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

DataTable dt = new DataTable();

SqlDataAdapter adapter = new SqlDataAdapter(command, connection);

adapter.Fill(dt);

dataGridView1.DataSource = dt.DefaultView;

connection.Close();

}

}

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

{

this.Close();

LoginForm formLogin = new LoginForm();

formLogin.Show();

}

}

}

**PupilForm**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace Praktika

{

public partial class Pupil : Form

{

private readonly string sqlconnection = "Data Source = localhost; Initial Catalog = School; Integrated Security = True";

public Pupil()

{

InitializeComponent();

}

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

{

string command = "SELECT DayWeek, TitleP, Letter FROM Raspisanie INNER JOIN Class ON Id\_class = Class\_id INNER JOIN Predmet ON Id\_Predmet = Predmet\_id WHERE Letter = 'А' AND DayWeek = 'ПН';";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

DataTable dt = new DataTable();

SqlDataAdapter adapter = new SqlDataAdapter(command, connection);

adapter.Fill(dt);

dataGridView1.DataSource = dt.DefaultView;

connection.Close();

}

}

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

{

string command = "SELECT TitleP, FIO FROM Predmet INNER JOIN Staff ON Id\_Staff = Staf\_id; ";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

DataTable dt = new DataTable();

SqlDataAdapter adapter = new SqlDataAdapter(command, connection);

adapter.Fill(dt);

dataGridView1.DataSource = dt.DefaultView;

connection.Close();

}

}

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

{

string command = "SELECT DayWeek, TitleP, Letter FROM Raspisanie INNER JOIN Class ON Id\_class = Class\_id INNER JOIN Predmet ON Id\_Predmet = Predmet\_id; ";

using (SqlConnection connection = new SqlConnection(sqlconnection))

{

connection.Open();

DataTable dt = new DataTable();

SqlDataAdapter adapter = new SqlDataAdapter(command, connection);

adapter.Fill(dt);

dataGridView1.DataSource = dt.DefaultView;

connection.Close();

}

}

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

{

this.Close();

LoginForm formLogin = new LoginForm();

formLogin.Show();

}

}

}

# Источники, использованные при разработке

1. ГОСТ Р ИСО\_МЭК 25051-2017 Требования к качеству готового к использованию программного продукта (RUSP) и инструкции по тестированию.
2. ЕСПД Единая система программной документации.
3. Изучаем C#. Руководство по созданию программ и приложения, Дэвид Скляр, 2019.

# Internet – ресурсы

1. C# Метанит [Электронный ресурс] – https://metanit.com/sharp/
2. Microsoft C# [Электронный ресурс] – https://docs.microsoft.com/en-us/dotnet/csharp/
3. Википедия C# [Электронный ресурс] - https://ru.wikipedia.org/wiki/C\_Sharp
4. Microsoft SQL Server [Электронный ресурс] -https://docs.microsoft.com/en-us/sql/?view=sql-server-ver15