Połatyńska Anna

**Zakład Metrologi Interdyscyplinarnej, Pracownia Wsparcia Metrologii** |

Główny Urząd Miar

Warszawa

luty 2019

Program RemaGUM

Opis rozwiązania

## Zapotrzebowanie

Oprogramowanie napisanie dla potrzeb **Warsztatu Głównego Urzędu Miar.** Program ma umożliwić zarządzanie warsztatem i magazynem materiałów i normaliów a tym samym ułatwić pracę osobie zarządzającej warsztatem i technikowi.

## Cel powstania

Program ewidencyjny maszyn i urządzeń w warsztacie GUM, oraz stanów magazynowych warsztatu. Stanowi on punkt wyjścia do stworzenia rozwiązań do zarządzania majątkiem poszczególnych laboratoriów GUM.

**Cel powstania**: ułatwienie ewidencjonowania i monitorowania stanu technicznego oraz napraw maszyn w warsztacie, oraz bieżące śledzenie stanów magazynowych .

Program składa się z dwóch modułów:

**1. spis maszyn -** poprzez bieżącą aktualizację np.: operatora maszyny, osoby zarządzającej, pomieszczenia, dodawanie nowego sprzętu i usuwania zużytego oraz monitorowania stanu technicznego i planu napraw i czasu wyłączenia z eksploatacji poszczególnych maszyn.

**2. spis magazynowy** - poprzez aktualizację, bieżącego śledzenia dostępnych ilości poszczególnych produktów, ich bieżącego zużycia, ilości odpadów itp., jak również informować o konieczności dokonania zakupów w przypadku przekroczenia stanu minimalnego. Spisy magazynowe powinny zawierać również stany magazynowe: wszelkich potrzebnych narzędzi   
i drobnych sprzętów warsztatowych jak: wiertła, klucze, frezy, gwintowniki itd. , z możliwością odpisywania jako zużyte, i w przypadku braku niezbędnej części – generowania informacji o konieczności zakupu.

1. **Opis bazy danych MS ACCESS RemaGUM**

Baza danych zosatała zaprojektowana z uwzglednieniem specyficznych potrzeb warsztatu GUM. Na dzień zamówienia programu RemaGUM nie było okreslone jakiego typu relacje mają być uwzglednione w bazie danych, z uwagi iż w przysłości może zaistnieć potrzeba zwiększenia elastyczności powiązań pomiędzy tabelami, wprowadzone zostały tabele pośrednie dające możliwość stworzenia realacji multi-multi.

### 3.1 Budowa modułu Magazyn

Dla uproszczenia w tabeli Materiały mieszczą się materiały i normalia warsztatowe (ci sami dostawcy dla obu kategorii). Rozróżnienie na poszczególne typy występuje przy użyciu danych słownikowych. Główną relacją jest połaczenie Materiałów z Dostawcami dlatego przyjęto, iż relacje multi-multi będą się odbywały poprzez tabelę posrednią



### 3.2 Budowa modułu Maszyny



## Opcje kompilacji

Program został napisany w C# w środowisku Microsoft Visual Studio Enterprise 2017.

Konfiguracja projekt rozwiązania programistycznego RemaGUM:

Aplikacja:

* Nazwa zestawu: RemaGUM
* Domyślna przestrzeń nazw: RemaGUM
* Platforma docelowa: .NET Framework 4 Client Profile
* Typ wyjściowy: Aplikacja Systemu Windows
* Obiekt uruchomieniowy: Not set

Kompilacja:

* Konfiguracja: Active (Debug)
* Platforma: Active (Any CPU)
* Symbole kompilacji warunkowej: Zdefiniuj stałą DEBUG, Zdefiniuj stałą TRACE
* Platforma docelowa: x86
* Zezwalaj na niebezpieczny kod: nie
* Optymalizuj kod: nie
* Poziom ostrzeżenia: 4
* Pomiń ostrzeżenia: brak
* Traktuj ostrzeżenia jako błędy: brak
* Ścieżka wyjściowa: bin\Debug\
* Plik dokumentacji XML: brak
* Zarejestruj w celu interakcji z modelem COM: brak
* Generuj zestaw serializacji: auto

Debug:

* Uruchom projekt: tak

1. **Użyte komponenty środowiska VS**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.IO;

using System.Xml;

using System.Diagnostics;

using System.Threading;

## Opis funkcji i klas

W niniejszym opisie przyjmuje się zasadę, że argumenty funkcji, które nie zostały opatrzone komentarzem są predefiniowane w VisualStudio.

### Klasa Main – start aplikacji.

namespace RemaGUM

{

/// <summary>

/// Class Program.

/// </summary>

static class Program

{

/// <summary>

/// Główny punkt wejścia dla aplikacji.

/// </summary>

[STAThread]

static void Main()

}

}

### Klasa Rest – hasło bazy danych MSACCESS.

namespace RemaGUM

{

/// <summary>

/// Class Rest.

/// </summary>

class Rest

{

/// <summary>

/// Połączenie z bazą danych.

/// </summary>

/// <param name="connString">The connection string.</param>

/// <returns>System.String.</returns>

public string dbConnection(string connString)

}// class Rest

}//namespace nsRest

### Obsługa działania programu – Klasa główna.

namespace RemaGUM

{

/// <summary>

/// Class SpisForm.

/// Implements the <see cref="System.Windows.Forms.Form" />

/// </summary>

/// <seealso cref="System.Windows.Forms.Form" />

public partial class SpisForm : Form

{

//ustawienia połączenie z bazą poprzez xml.

private settings \_settings;

/// <summary>

/// The connection string

/// </summary>

private string \_connString;

/// <summary>

/// The rest

/// </summary>

private Rest \_rest = new Rest();

/// <summary>

/// The help file- plik pomocy RemaGUM.

/// </summary>

private string \_helpFile = Application.StartupPath + "\\help.chm";

/// <summary>

/// zawartosc pliku-dane odczytane z pliku zdjęcia.

/// </summary>

private byte[] \_zawartoscPliku;

/// <summary>

/// dir nazwa -nazwa katalogu tymczasowego.

/// </summary>

string \_dirNazwa = "C:\\tempRemaGUM";

/// <summary>

/// dir pelna nazwa katalogu tymczasowego - pelna nazwa.

/// </summary>

string \_dirPelnaNazwa = string.Empty;

/// <summary>

/// Enum \_status- status działania formularza.

/// </summary>

enum \_status { edycja, nowy };

/// <summary>

/// status form - wartośc statusu formularza.

/// </summary>

private byte \_statusForm;

/// <summary>

/// The maszyna tag przechowuje identyfikatory maszyn dla operatora.

/// </summary>

private int[] \_maszynaTag;

/// <summary>

/// The maszyna identifier

/// </summary>

int \_maszynaId = -1; // identyfikator maszyny przy starcie programu.

/// <summary>

/// indeks szukanej maszyny.

/// </summary>

int \_maszynaSzukajIdx = 0;

/// <summary>

/// indeks szukanego dysponenta.

/// </summary>

int \_dysponentSzukajIdx = 0;

/// <summary>

/// indeks szukanego materiału.

/// </summary>

int \_materialSzukajIdx = 0;

/// <summary>

/// </summary>

private ToolTip \_tt;

/// <summary>

/// The interwal przegladow

/// </summary>

private int \_interwalPrzegladow = 365; //w dniach = 1 rok

private nsAccess2DB.OperatorBUS \_OperatorBUS;

private nsAccess2DB.MaszynyBUS \_maszynyBUS;

private nsAccess2DB.MaterialyBUS \_materialyBUS;

/// <summary>

/// Klasa umożliwiająca połączenie bazy danych za pomocą connectionStringa zapisanego w pliku xml

/// </summary>

public class settings

{

private string \_XMLfile;

private string \_connectionString; //connectionString

private string \_dBcopyFile; //pełna ścieżka kopii bazy danych

private int \_copyInterval; //interwał sporzadzania kopii bazy

private string \_dBhistoryDir; //pełna nazwa katalogu historii bazy

private int \_historyInterval; //interwał sprzadzania kopii w katalogu hsitorii bazy

/// <summary>

/// Konstruktor wymiany danych ustawień programu w XML.

/// </summary>

public settings()

/// <summary>

/// Czyta plik.

/// </summary>

public void read()

/// <summary>

/// Czyta plik.

/// </summary>

/// <param name="XMLfile">Nazwa pliku xml</param>

public void read(string XMLfile)

/// <summary>

/// Zapisuje do pliku.

/// </summary>

public void write()

/// <summary>

/// Connection string.

/// </summary>

public string connectionString

/// <summary>

/// Connection string.

/// </summary>

public string XMLfile

/// <summary>

/// Pełna ścieżka kopii bazy danych.

/// </summary>

public string dBcopyFile

/// <summary>

/// Interwał w minutach sporzadzania kopii bazy danych.

/// </summary>

public int copyInterval

/// <summary>

/// Pełna nazwa katalogu historii bazy.

/// </summary>

public string dBhistoryDir

/// <summary>

/// Interwał w dniach sporządzania kopii bazy w katalogu historii.

/// </summary>

public int historyInterval

}//class settings

/// <summary>

/// Konstruktor formularza.

/// </summary>

/// <param name="connStr">Połaczenie z bazą.</param>

public SpisForm()

{

InitializeComponent();

//połączenie z bazą danych na sztywno

//nsRest.Rest rest = new nsRest.Rest();

//\_connString += rest.dbConnection(\_connString);

//połączenie z bazą danych przez connection string z xml.

\_settings = new settings();

\_settings.XMLfile = Application.StartupPath + "\\settings.xml";

\_settings.read();

\_connString = \_rest.dbConnection(\_settings.connectionString);

\_statusForm = (int)\_status.edycja;

\_OperatorBUS = new nsAccess2DB.OperatorBUS(\_connString);

\_maszynyBUS = new nsAccess2DB.MaszynyBUS(\_connString);

\_materialyBUS = new nsAccess2DB.MaterialyBUS(\_connString);

radioButtonNazwa.Checked = true; // przy starcie programu zaznaczone sortowanie po nazwie.

WypelnijCzestotliwosc();

WypelnijKategorie();

WypelnijDzial();

WypelnijPropozycje();

WypelnijStan\_techniczny();

WypelnijOperatorow\_maszyn(checkedListBoxOperatorzy\_maszyn);// wypełniam operatorów maszyn na poczatku uruchomienia programu.

WypelnijDysponent();

}

// Indeksowanie w zakładkach

//------------------------------------ Zakładka Maszyny

//dane formularza Maszyny

//listBoxMaszyny.TabIndex = 0;

comboBoxKategoria.TabIndex = 1;

textBoxNazwa.TabIndex = 2;

textBoxTyp.TabIndex = 3;

textBoxNr\_inwentarzowy.TabIndex = 4;

textBoxNr\_fabryczny.TabIndex = 5;

richTextBoxProducent.TabIndex = 6;

textBoxRok\_produkcji.TabIndex = 7;

textBoxNr\_pom.TabIndex = 8;

dateTimePickerData\_ost\_przegl.TabIndex = 9;

comboBoxDzial.TabIndex = 10;

//richTextBoxUwaga.TabIndex = 11; //pole nieaktywne na komunikaty systemu

//dateTimePickerData\_kol\_przegl.TabIndex = 12;

comboBoxDysponent.TabIndex = 13;

checkedListBoxOperatorzy\_maszyn.TabIndex = 14;

richTextBoxUwagi.TabIndex = 15;

//wyszukiwanie Maszyny po wpisanej nazwie

textBoxWyszukiwanie.TabIndex = 16;

buttonSzukaj.TabIndex = 17;

//zdjęcie i przyciski dot obsł zdjęcia

//pictureBox1.TabIndex = 18;

buttonPokazZdj.TabIndex = 19;

buttonUsunZdj.TabIndex = 20;

//ankietka dot. Maszyn

comboBoxStan\_techniczny.TabIndex = 21;

comboBoxWykorzystanie.TabIndex = 22;

comboBoxPropozycja.TabIndex = 23;

textBoxNr\_prot\_BHP.TabIndex = 24;

//przyciski zapisz/edytuj itp

buttonNowa.TabIndex = 25;

buttonZapisz.TabIndex = 26;

buttonAnuluj.TabIndex = 27;

buttonUsun.TabIndex = 28;

//sortowanie Maszyn po radio buttonach

radioButtonNazwa.TabIndex = 29;

radioButtonTyp.TabIndex = 30;

radioButtonNr\_fabryczny.TabIndex = 31;

radioButtonNr\_inwentarzowy.TabIndex = 32;

radioButtonNr\_pomieszczenia.TabIndex = 33;

radioButtonData\_ost\_przegl.TabIndex = 34;

//------------------------------------------------ Zakładka Operatorzy maszyn

//sortowanie po comboBoxOperator

comboBoxOperator.TabIndex = 1;

// lista operatorów

//listBoxOperator.TabIndex = 2;

//wyszukiwanie

textBoxWyszukiwanieOperator.TabIndex = 3;

buttonSzukajOperator.TabIndex = 4;

//dane forlumarza

textBoxImieOperator.TabIndex = 5;

textBoxNazwiskoOperator.TabIndex = 6;

comboBoxDzialOperator.TabIndex = 7;

textBoxUprawnienieOperator.TabIndex = 8;

dateTimePickerDataKoncaUprOp.TabIndex = 9;

richTextBoxUprawnieniaOperatora.TabIndex = 10;

//listBoxMaszynyOperatora.TabIndex = 11;

//przyciski zapisz/edytuj itp

buttonNowaOperator.TabIndex = 12;

buttonZapiszOperator.TabIndex = 13;

buttonAnulujOperator.TabIndex = 14;

buttonUsunOperator.TabIndex = 15;

// ------------------------------------------- Zakładka Dysponenci maszyn.

//lista dysponentów maszyn

//listBoxDysponent.TabIndex = 1;

//wyszukiwanie

textBoxWyszukiwanieDysponent.TabIndex = 2;

buttonSzukajDysponent.TabIndex = 3;

//dane dysponenta

textBoxImieDysponent.TabIndex = 4;

textBoxNazwiskoDysponent.TabIndex = 5;

comboBoxDzialDysponent.TabIndex = 6;

richTextBoxDysponent\_dane.TabIndex = 7;

//przyciski zapisz/edytuj itp

buttonNowaDysponent.TabIndex = 8;

buttonZapiszDysponent.TabIndex = 9;

buttonAnulujDysponent.TabIndex = 10;

buttonUsunDysponent.TabIndex = 11;

//lista maszyn, którymi zarządza dysponent

//listBoxMaszynyDysponenta.TabIndex = 12;

// --------------------------------------------- Zakładka Materiały

//sortowanie Materiału po radio buttonach

radioButtonNazwa\_mat.TabIndex = 1;

radioButtonTyp\_mat.TabIndex = 2;

radioButtonStan\_min\_mat.TabIndex = 3;

radioButtonMagazyn\_ilosc\_mat.TabIndex = 4;

//lista materiałów/ normaliów

//listBoxMaterialy.TabIndex = 5;

//wyszukiwanie Materiału po wpisanej nazwie

textBoxWyszukaj\_mat.TabIndex = 6;

buttonSzukaj\_mat.TabIndex = 7;

//dane formularza Magazyn

comboBoxWyborMagazyn.TabIndex = 8;

textBoxNazwaMat.TabIndex = 9;

textBoxTypMat.TabIndex = 10;

comboBoxRodzajMat.TabIndex = 11;

comboBoxJednostkaMat.TabIndex = 12;

//richTextBoxKomunikatMaterialy.TabIndex = 13;// komunikaty dotyczące dostępności materiału.

//Gospodarka magazynowa

textBoxMagazynMat.TabIndex = 14;

textBoxZuzycieMat.TabIndex = 15;

textBoxOdpadMat.TabIndex = 16;

textBoxMinMat.TabIndex = 17;

textBoxZapotrzebowanieMat.TabIndex = 18;

//przyciski zapisz/edytuj itp

buttonNowaMat.TabIndex = 19;

buttonZapiszMat.TabIndex = 20;

buttonAnulujMat.TabIndex = 21;

buttonUsunMat.TabIndex = 22;

//dane dostawców Materiałów

checkedListBoxDostawcyMat.TabIndex = 23;

richTextBoxDaneDodatkoweDostawca.TabIndex = 24;

linkLabelDostawcaMat2.TabIndex = 25;

// -----------------------------------------------Zakładka Dostawcy

//lista dostawców

//listBoxDostawcy.TabIndex = 1;

// dane dostawców

textBoxNazwaDostawcy.TabIndex = 2;

richTextBoxDostawca.TabIndex = 3;

textBoxLink.TabIndex = 4;

linkLabelDostawcaMat.TabIndex = 5;

//przyciski zapisz/edytuj itp

buttonNowyDostawca.TabIndex = 6;

buttonZapiszDostawca.TabIndex = 7;

buttonAnulujDostawca.TabIndex = 8;

buttonUsunDostawca.TabIndex = 9;

// tool tipy

\_tt = new ToolTip();

\_tt.SetToolTip(listBoxMaszyny, "Lista maszyn, przyrządów i urządzeń itp.");

\_tt.SetToolTip(comboBoxKategoria, "Kategoria maszyn, przyrządów, urządzeń np. maszyny warsztatowe, lub przyrządy pomiarowe.");

\_tt.SetToolTip(textBoxNazwa, "Nazwa maszyny, przyrządu lub urządzenia.");

\_tt.SetToolTip(textBoxTyp, "Typ maszyny, przyrządu lub urządzenia.");

\_tt.SetToolTip(textBoxNr\_inwentarzowy, "Numer naklejki GUM (inwentarzowy) maszyny, przyrządu lub urządzenia.");

\_tt.SetToolTip(textBoxNr\_fabryczny, "Numer fabryczny maszyny, przyrządu lub urządzenia.");

\_tt.SetToolTip(richTextBoxProducent, "Producent maszyny, przyrządu lub urządzenia, dane adresowe, adres strony www i inne dodatkowe dane.");

\_tt.SetToolTip(textBoxRok\_produkcji, "Rok wyprodukowania.");

\_tt.SetToolTip(textBoxNr\_pom, "Numer pomieszczenia GUM, gdzie znajuje się maszyna, przyrząd lub urządzenie.");

\_tt.SetToolTip(dateTimePickerData\_ost\_przegl, "Data ostatniej kontroli dostosowania do minimalnych wymagań w zakresie BHP, lub innego dokumentu.");

\_tt.SetToolTip(comboBoxDzial, "Nazwa dział lista maszyn, przyrządów i urządzeń itp.");

\_tt.SetToolTip(richTextBoxUwaga, "Komunikaty programu dotyczące terminu przeglądu wybranej maszyny. Pole nieaktywne."); //pole nieaktywne na komunikaty systemu

//\_tt.SetToolTip(dateTimePickerData\_kol\_przegl, "Data kolejnej kontroli dostosowania do minimalnych wymagań w zakresie BHP, lub innego dokumentu.");

\_tt.SetToolTip(comboBoxDysponent, "Osoba zarządzająca maszynami (dysponent maszyny).");

\_tt.SetToolTip(checkedListBoxOperatorzy\_maszyn, "Główna osoba użytkująca maszynę (posiadająca odpowiednie uprawnienia).");

\_tt.SetToolTip(richTextBoxUwagi, "Opis wszelkich innych zdarzeń/statusów mających istotny wpływ na maszynę, przyrząd lub urządzenie.");

//wyszukiwanie Maszyny po wpisanej nazwie

\_tt.SetToolTip(textBoxWyszukiwanie, "Wpisz jakiej maszyny szukasz.");

\_tt.SetToolTip(buttonSzukaj, "Szukanie w bazie maszyn.");

//zdjęcie i przyciski dot obsł zdjęcia

\_tt.SetToolTip(pictureBox1, "Zdjęcie maszyny, przyrządu lub urządzenia.");

\_tt.SetToolTip(buttonPokazZdj, "Przycisk dodania zdjęcia z wybranego katalogu do bazy danych MS ACCESS.");

\_tt.SetToolTip(buttonUsunZdj, "Przycisk usunięcia zdjęcia z bazy danych.");

//ankietka dot. Maszyn

\_tt.SetToolTip(comboBoxStan\_techniczny, "Stan techniczy: złom, do naprawy, dobry.");

\_tt.SetToolTip(comboBoxWykorzystanie, "Częstotliwość wykorzystania: nieuzywana, rzadziej niż kilka razy w roku, kilka razy w roku, kilka razy w okresie pół roku, kilka razy w kwartale, kilka razy w miesiącu.");

\_tt.SetToolTip(comboBoxPropozycja, "Propozycja: do likwidacji, do remontu, zachować.");

\_tt.SetToolTip(textBoxNr\_prot\_BHP, "Numer nadany w protokole kontroli dostosowania maszyny do minimalnych wymagań w zakresie BHP z dnia 12.06.2006 r.");

//przyciski zapisz/edytuj itp

\_tt.SetToolTip(buttonNowa, "Nowa pozycja w bazie.");

\_tt.SetToolTip(buttonZapisz, "Zapis nowej maszyny, przyrządu lub urządzenia lub edycja wybranej pozycji.");

\_tt.SetToolTip(buttonAnuluj, "Anulowanie zmiany.");

\_tt.SetToolTip(buttonUsun, "Usuwa pozycję z bazy.");

//sortowanie Maszyn po radio buttonach

\_tt.SetToolTip(radioButtonNazwa, "Sortuj po nazwie maszyny, przyrządu lub urządzenia.");

\_tt.SetToolTip(radioButtonTyp, "Sortuj po typie.");

\_tt.SetToolTip(radioButtonNr\_fabryczny, "Sortuj po numerze fabrycznym.");

\_tt.SetToolTip(radioButtonNr\_inwentarzowy, "Sortuj po numerze inwentarzowym.");

\_tt.SetToolTip(radioButtonNr\_pomieszczenia, "Sortuj po numerze pomieszczenia.");

\_tt.SetToolTip(radioButtonData\_ost\_przegl, "Sortuj po dacie ostatniego przegladu.");

//------------------------------------------------ Zakładka Operatorzy maszyn

//sortowanie po comboBoxOperator

\_tt.SetToolTip(comboBoxOperator, "Sortowanie operatorow po dacie końca uprawnień.");

// lista operatorów

\_tt.SetToolTip(listBoxOperator, "Lista wszystkich operatorów maszyn.");

//wyszukiwanie

\_tt.SetToolTip(textBoxWyszukiwanieOperator, "Wpisz nazwisko operatora, którego szukasz.");

\_tt.SetToolTip(buttonSzukajOperator, "Szukanie w bazie operatora.");

//dane forlumarza

\_tt.SetToolTip(textBoxImieOperator, "Imię operatora maszyny.");

\_tt.SetToolTip(textBoxNazwiskoOperator, "Nazwisko operatora maszyny.");

\_tt.SetToolTip(comboBoxDzialOperator, "Nazwa działu, do którego należy operator maszyny.");

\_tt.SetToolTip(textBoxUprawnienieOperator, "Rodzaj uprawnienia, jakie posiada operator.");

\_tt.SetToolTip(dateTimePickerDataKoncaUprOp, "Data końca uprawnień operatora maszyny.");

\_tt.SetToolTip(richTextBoxUprawnieniaOperatora, "Uprawnienia jaki posiada wybrany operator.");

\_tt.SetToolTip(listBoxMaszynyOperatora, "Lista obsługiwanych przez operatora maszyn.");

//przyciski zapisz/edytuj itp

\_tt.SetToolTip(buttonNowaOperator, "Nowa pozycja w bazie.");

\_tt.SetToolTip(buttonZapiszOperator, "Zapis nowego operatora lub edycja wybranej pozycji.");

\_tt.SetToolTip(buttonAnulujOperator, "Anulowanie zmiany.");

\_tt.SetToolTip(buttonUsunOperator, "Usuwa pozycję z bazy.");

// ------------------------------------------- Zakładka Dysponenci maszyn.

//lista dysponentów maszyn

\_tt.SetToolTip(listBoxDysponent, "Lista dysponentów maszyn.");

//wyszukiwanie

\_tt.SetToolTip(textBoxWyszukiwanieDysponent, "Wpisz nazwisko dysponenta, którego szukasz.");

\_tt.SetToolTip(buttonSzukajDysponent, "Szukanie w bazie dysponenta.");

//dane dysponenta

\_tt.SetToolTip(textBoxImieDysponent, "Imię dysponenta maszyny.");

\_tt.SetToolTip(textBoxNazwiskoDysponent, "Nazwisko dysponenta maszyny.");

\_tt.SetToolTip(comboBoxDzialDysponent, "Nazwa działu, do którego należy dysponent maszyny.");

\_tt.SetToolTip(richTextBoxDysponent\_dane, "Dodatkowe dane dysponenta maszyny.");

//przyciski zapisz/edytuj itp

\_tt.SetToolTip(buttonNowaDysponent, "Nowa pozycja w bazie.");

\_tt.SetToolTip(buttonZapiszDysponent, "Zapis nowego dysponenta lub edycja wybranej pozycji.");

\_tt.SetToolTip(buttonAnulujDysponent, "Anulowanie zmiany.");

\_tt.SetToolTip(buttonUsunDysponent, "Usuwa pozycję z bazy.");

//lista maszyn, którymi zarządza dysponent

\_tt.SetToolTip(listBoxMaszynyDysponenta, "Maszyny, którymi dysponuje dysponent.");

// --------------------------------------------- Zakładka Materiały

//sortowanie Materiału po radio buttonach

\_tt.SetToolTip(radioButtonNazwa\_mat, "Sortuj po nazwie materiałów / normaliów.");

\_tt.SetToolTip(radioButtonTyp\_mat, "Sortuj po typie materiałów / normaliów.");

\_tt.SetToolTip(radioButtonStan\_min\_mat, "Sortuj po cenie materiałów / normaliów.");

\_tt.SetToolTip(radioButtonMagazyn\_ilosc\_mat, "Sortuj po dostępnych ilościach w magazynie.");

//lista materiałów/ normaliów

\_tt.SetToolTip(listBoxMaterialy, "Lista materiałów / normaliów.");

//wyszukiwanie Materiału po wpisanej nazwie

\_tt.SetToolTip(textBoxWyszukaj\_mat, "Wpisz czego szukasz.");

\_tt.SetToolTip(buttonSzukaj\_mat, "Szukanie w bazie materiałów / normaliów.");

//dane formularza Magazyn

\_tt.SetToolTip(comboBoxWyborMagazyn, "Wybór magazynu materiałów lub normaliów.");

\_tt.SetToolTip(textBoxNazwaMat, "Nazwa materiałów / normaliów.");

\_tt.SetToolTip(textBoxTypMat, "Typ materiałów / normaliów.");

\_tt.SetToolTip(comboBoxRodzajMat, "Rodzaj materiałów / normaliów.");

\_tt.SetToolTip(comboBoxJednostkaMat, "Jednostka miary materiałów / normaliów.");

\_tt.SetToolTip(richTextBoxKomunikatMaterialy, "Komunikaty systemowe, dotyczące dostępności materiału.");// komunikaty dotyczące dostępności materiału.

//Gospodarka magazynowa

\_tt.SetToolTip(textBoxMagazynMat, "Stan magazynowy materiałów / normaliów.");

\_tt.SetToolTip(textBoxZuzycieMat, "Wpisz ile podlega zużyciu.");

\_tt.SetToolTip(textBoxOdpadMat, "Wpisz ile jest odpadem.");

\_tt.SetToolTip(textBoxMinMat, "Wpisz stan minimalny.");

\_tt.SetToolTip(textBoxZapotrzebowanieMat, "Wpisz zapotrzebowanie materiałów / normaliów.");

//przyciski zapisz/edytuj itp

\_tt.SetToolTip(buttonNowaMat, "Nowa pozycja w bazie materiałów / normaliów.");

\_tt.SetToolTip(buttonZapiszMat, "Zapis nowej pozycji w bazie materiałów / normaliów lub edycja wybranej pozycji.");

\_tt.SetToolTip(buttonAnulujMat, "Anulowanie zmiany.");

\_tt.SetToolTip(buttonUsunMat, "Usuwa pozycję z bazy.");

//dane dostawców Materiałów

\_tt.SetToolTip(checkedListBoxDostawcyMat, "Dostawcy wybranego materiału.");

\_tt.SetToolTip(richTextBoxDaneDodatkoweDostawca, "Wszelkie dodatkowe informacje dotyczące dostawcy np. dane adresowe, telefony, informacje dotyczące otrzymanych upustów.");

\_tt.SetToolTip(linkLabelDostawcaMat2, "link do strony dostawcy materiałów / normaliów.");

// -----------------------------------------------Zakładka Dostawcy

//lista dostawców

\_tt.SetToolTip(listBoxDostawcy, "Lista dostawców.");

// dane dostawców

\_tt.SetToolTip(textBoxNazwaDostawcy, "Nazwa wybranego dostawcy.");

\_tt.SetToolTip(richTextBoxDostawca, "Wszelkie dodatkowe informacje dotyczące dostawcy np.dane adresowe, telefony, informacje dotyczące otrzymanych upustów.");

\_tt.SetToolTip(textBoxLink, "Wpisanie linku wybranego dostawcy.");

\_tt.SetToolTip(linkLabelDostawcaMat, "Wyświetlenie linku wybranego dostawcy.");

//przyciski zapisz/edytuj itp

\_tt.SetToolTip(buttonNowyDostawca, "Nowa pozycja w bazie dostawców materiałów / normaliów.");

\_tt.SetToolTip(buttonZapiszDostawca, "Zapis nowej pozycji w bazie dostawców lub edycja wybranej pozycji.");

\_tt.SetToolTip(buttonAnulujDostawca, "Anulowanie zmiany.");

\_tt.SetToolTip(buttonUsunDostawca, "Usuwa pozycję z bazy.");

}//public SpisForm()

// Wyświetla komunikaty chwilowe w programie.

private void pokazKomunikat(string tresc)

/// <summary>

///Obsługuje tiki timera.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void timer1\_Tick(object sender, EventArgs e)

/// <summary>

/// Ustawia datę i czas. 2018-02-14 11:06:21 format dat godzina indeks spacji dzieli ciąg na 2 podciagi data + godzina

/// </summary>

private void SetDataAndTime()

/// <summary>

/// Wyświetla komponenty w zależności od indeksu zakładki.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void tabControlZakladki\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Przycisk uruchamiający okienko o programie.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Przycisk uruchomiający pomoc programu RemaGUM.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Czyści dane z formularza maszyny.

/// </summary>

private void CzyscDaneMaszyny()

/// <summary>

/// zmiana indeksu w list box maszyny

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void listBoxMaszyny\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Odświeża listę maszyn w listBoxMaszyny.

/// </summary>

private void OdswiezListeMaszyn()

{

/// <summary>

/// Wypelnij operatorow maszyn - checked -boxy.

/// </summary>

/// <param name="v">The v.</param>

private void WypelnijOperatorow\_maszyn(CheckedListBox v)

/// <summary>

/// wypełnia listbox kategoriami maszyn

/// </summary>

private void WypelnijKategorie()

/// <summary>

/// Handles the SelectedIndexChanged event of the comboBoxKategoria control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBoxKategoria\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Wypełnia danymi Dysponentów w zakładce Maszyny.

/// </summary>

private void WypelnijDysponent()

/// <summary>

/// Obsługa comboBox\_Dysponent control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBox\_Dysponent\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Sortuje po nazwie maszyny.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButtonNazwa\_CheckedChanged(object sender, EventArgs e)

/// <summary>

/// Sortuje po numerze inwentarzowym maszyny.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButton\_Nr\_Inwentarzowy\_CheckedChanged(object sender, EventArgs e)

/// <summary>

/// Sortuje po typie maszyny.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButton\_Typ\_CheckedChanged(object sender, EventArgs e)

/// <summary>

/// Sortuje po numerze fabrycznym maszyny.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButtonNr\_fabrycznyCheckedChanged(object sender, EventArgs e)

/// <summary>

/// Sortuje po numerze pomieszczenia, w którym znajduje się maszyna.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButton\_Nr\_Pomieszczenia\_CheckedChanged(object sender, EventArgs e)

/// <summary>

/// Sortuje po dacie ostatniego przeglądu maszyny.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButtonData\_ost\_przegladu\_CheckedChanged(object sender, EventArgs e)

/// <summary>

/// wyszukuje maszynę po wpisaniu dowolnego ciągu wyrazów

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

///

/// <summary>

/// Wypełnia listbox Działami w których znajdują się maszyny.

/// </summary>

private void WypelnijDzial()

/// <summary>

/// Handles the SelectedIndexChanged event of the comboBoxDzial control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBoxDzial\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Wypełnia listbox pozycjami częstotliwości Wykorzystania.

/// </summary>

private void WypelnijCzestotliwosc()

// przypisuje w combo identyfikator = nazwę częstotliwości wykorzystania

/// <summary>

/// Handles the SelectedIndexChanged event of the ComboBoxWykorzystanie control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void ComboBoxWykorzystanie\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// wypełnia listbox propozycjami co zrobić z maszyną (zachować/złomować itp)

/// </summary>

private void WypelnijPropozycje()

/// <summary>

/// Handles the SelectedIndexChanged event of the comboBoxPropozycja control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBoxPropozycja\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Wypełnia stan techniczy maszyny.

/// </summary>

private void WypelnijStan\_techniczny()

/// <summary>

/// Handles the SelectedIndexChanged event of the comboBoxStan\_techniczny control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBoxStan\_techniczny\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Klawisz Nowa czyści formularz (maszyna).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Klawisz Anuluj (maszyna).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Klawisz Usuń (maszyna).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Działania po wciśnięciu klawisza Zapisz (maszyny).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Pokazuje nazwę zdjęcia w formie linku.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void linkLabelNazwaZdjecia\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

/// <summary>

/// Przycisk Wgraj/pokaż zdjęcie maszyny.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Przycisk Usuń zdjęcie.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Wypełnia listBoxMaterialy.

/// </summary>

private void WypelnijMaterialyNazwami()

/// <summary>

/// Zmiana indeksu w list box Materialy.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void listBoxMaterialy\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Wypełnia CheckedListBoxDostawcyMat (zakładka Materiały).

/// </summary>

/// <param name="v">CheckedListBoxDostawcyMat</param>

private void WypelnijCheckedBoxDostawcow(CheckedListBox v)

/// <summary>

/// Zmiana indeksu dostawcy w checkedListBoxDostawcyMat (aktualizacja pól do odczytu danych wybranego dostawcy).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void checkedListBoxDostawcyMat\_SelectedIndex(object sender, EventArgs e)

/// <summary>

/// Czyści dane w formularzu Materiały.

/// </summary>

private void CzyscDaneMaterialy()

/// <summary>

/// Odswieża dane w zakładce materiały.

/// </summary>

private void OdswiezMaterialy()

/// <summary>

/// Odświeża checked boxy dostawców w zakładce materiały.

/// </summary>

private void OdswiezDostawcowWMaterialach()

/// <summary>

/// Wybiera rodzaj magazynu (materiały lub normalia).

/// </summary>

private void WybierzMagazyn()

/// <summary>

/// Handles the SelectedIndexChanged event of the comboBoxWyborMagazyn control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBoxWyborMagazyn\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Wypełnia jednostkę miar dla Materiałów i Normaliów.

/// </summary>

private void WypelnijJednostka\_miar()

/// <summary>

/// Handles the SelectedIndexChanged event of the comboBoxJednostka\_mat control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBoxJednostka\_mat\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Wypełnia rodzaj materiału (zakładka materiały).

/// </summary>

private void WypelnijRodzaj\_mat()

/// <summary>

/// Handles the SelectedIndexChanged event of the comboBoxRodzaj control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBoxRodzaj\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Klinknięcie w link otwiera stronę z linka.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void linkLabelDostawcaMat2\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

/// <summary>

/// Sortowanie po nazwie materiału (zakładka materiał).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButtonNazwa\_mat\_CheckedChanged(object sender, EventArgs e)

/// <summary>

/// Sortowanie po typie materiału (zakładka materiał).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButtonTyp\_mat\_CheckedChanged(object sender, EventArgs e)

/// <summary>

/// Sortowanie po stanie minimalnym materiału (zakładka materiał).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButtonStan\_min\_mat\_CheckedChanged(object sender, EventArgs e)

/// <summary>

/// Sortowanie po ilości materiału w magazynie (zakładka materiał).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void radioButtonMagazyn\_ilosc\_mat\_CheckedChanged(object sender, EventArgs e)

/// <summary>

/// Działania po wciśnięciu klawisza Nowa (materiały).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void ButtonNowa\_mat\_Click(object sender, EventArgs e)

/// <summary>

/// Aktywuje pola panelu materiałów bez pola stan magazynowy

/// </summary>

private void AktywujPanelMaterial()

/// <summary>

/// Działania po wciśnięciu klawisza Anuluj (materiały).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void buttonAnuluj\_mat\_Click(object sender, EventArgs e)

/// <summary>

/// Działania po wciśnięciu klawisza Usuń (materiały).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void buttonUsun\_mat\_Click(object sender, EventArgs e)

/// <summary>

/// Działania po wciśnięciu klawisza Zapisz (materiały).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

//TODO wyszukiwarka materiałów.

/// <summary>

/// Wyszukuje materiał po dowolnym ciągu znaków w nazwie lub rodzaju materiału.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void buttonSzukaj\_mat\_Click(object sender, EventArgs e)

/// <summary>

/// Wypełnia listę dostawców wg nazwy.

/// </summary>

private void WypelnijListeDostawcowDanymi()

/// <summary>

/// Zmiana indeksu

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void listBoxDostawcy\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Tworzy nowego dostawcę (wymaga przycisku zapisz).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// otwiera link przy kliknięciu.

/// </summary>

/// <param name = "sender" ></ param >

/// < param name= "e" ></ param >

private void linkLabelDostawcaMat\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

/// <summary>

/// Usuwa link z obiektu (wymaga uzycia klawisza zapisz).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Akcje po wciśnięciu przycisku Zapisz.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// działania po wciśnięciu przycisku Anuluj.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Czyści pola dane dostawcy w zakładce Dostawca.

/// </summary>

private void CzyscDaneDostawcy()

/// <summary>

/// Usuwa wybranego dostawcę po Identyfikatorze.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// wyświetla listę operatorów maszyn po imieniu i nazwisku.

/// </summary>

private void WypelnijOperatorowDanymi()

/// <summary>

/// zmiana indeksu w list box operator maszyny

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void listBoxOperator\_maszyny\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// wyświetla listę maszyn dla danego operatora.

/// </summary>

private void WypelnijOperatorowMaszynami()

/// <summary>

/// Zmiana indeksu dla comboBoxOperator

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void comboBoxOperator\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Wypełnia dział wskazanego Operatora.

/// </summary>

private void WypelnijDzialOperatora()

/// <summary>

/// Handles the SelectedIndexChanged event of the comboBoxDzial\_operator\_maszyny control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBoxDzial\_operator\_maszyny\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Działania po wciśnięciu klawisza Nowa (zakładka operator).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Działania po wciśnięciu klawisza Zapisz (zakładka operator).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Działania po wciśnięciu klawisza Anuluj (zakładka operator).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Działania po wciśnięciu klawisza Usuń (zakładka operator).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Działania po wciśnięciu klawisza Szukaj (zakładka operator).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// wyświetla listę dysponentów maszyn - nazwisko i imię.

/// </summary>

private void WypelnijDysponentowDanymi()

/// <summary>

/// Czyści dane w zakładce dysponent maszyn.

/// </summary>

private void CzyscDaneDysponenta()

/// <summary>

/// Odświeża listBoxDysponent.

/// </summary>

private void OdswiezDysponentowMaszyn()

/// <summary>

/// Zmiana indeksu w list box dysponent.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void listBoxDysponent\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Wypełnia dane listBoxMaszynyDysponenta

/// </summary>

private void WypelnijDysponentowMaszynami()

/// <summary>

/// Wypełnia dział zakładka dysponent.

/// </summary>

private void WypelnijDzialDysponenta()

/// <summary>

/// Handles the SelectedIndexChanged event of the comboBoxDzial\_dysponent control.

/// </summary>

/// <param name="sender">The source of the event.</param>

/// <param name="e">The <see cref="EventArgs"/> instance containing the event data.</param>

private void comboBoxDzial\_dysponent\_SelectedIndexChanged(object sender, EventArgs e)

/// <summary>

/// Działania po wciśnięciu klawisza Nowa (zakładka dysponent).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Działania po wciśnięciu klawisza Zapisz (zakładka dysponent).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Działania po wciśnięciu klawisza Anuluj (zakładka dysponent).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Działania po wciśnięciu klawisza Usuń (zakładka dysponent).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

/// <summary>

/// Działania po wciśnięciu klawisza Szukaj (zakładka dysponent).

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

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

}// public partial class SpisForm : Form

}//namespace RemaGUM

### Komponenty wizualne – okienko „o RemaGUM”.

namespace RemaGUM

{

public partial class FormORemaGUM : Form

{

/// <summary>

/// Initializes a new instance of the <see cref="FormORemaGUM" /> class.

/// </summary>

public FormORemaGUM()

{

InitializeComponent();

}//FormORemaGUM()

}//FormORemaGUM : Form

}//FormORemaGUM : Form

### Komponenty wizualne- Klasa Frame

namespace RemaGUM

{

partial class Frame

{

/// <summary>

/// Required designer variable.

/// </summary>

private System.ComponentModel.IContainer components = null;

/// <summary>

/// Clean up any resources being used.

/// </summary>

/// <param name="disposing">true if managed resources should be disposed; otherwise, false.</param>

protected override void Dispose(bool disposing)

{

if (disposing && (components != null))

{

components.Dispose();

}

base.Dispose(disposing);

}

#region Windows Form Designer generated code

/// <summary>

/// Required method for Designer support - do not modify

/// the contents of this method with the code editor.

/// </summary>

private void InitializeComponent()

{

this.labelTekst = new System.Windows.Forms.Label();

this.SuspendLayout();

//

// labelTekst

//

this.labelTekst.AutoEllipsis = true;

this.labelTekst.FlatStyle = System.Windows.Forms.FlatStyle.Popup;

this.labelTekst.Font = new System.Drawing.Font("Microsoft Sans Serif", 14.25F, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((byte)(238)));

this.labelTekst.ForeColor = System.Drawing.Color.Red;

this.labelTekst.Location = new System.Drawing.Point(11, 9);

this.labelTekst.Margin = new System.Windows.Forms.Padding(2, 0, 2, 0);

this.labelTekst.Name = "labelTekst";

this.labelTekst.Size = new System.Drawing.Size(545, 96);

this.labelTekst.TabIndex = 1;

this.labelTekst.Text = "Tekst";

this.labelTekst.TextAlign = System.Drawing.ContentAlignment.MiddleCenter;

//

// Frame

//

this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);

this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;

this.BackColor = System.Drawing.Color.OldLace;

this.ClientSize = new System.Drawing.Size(567, 114);

this.Controls.Add(this.labelTekst);

this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None;

this.Name = "Frame";

this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;

this.Text = "frame";

this.ResumeLayout(false);

}

#endregion

private System.Windows.Forms.Label labelTekst;

}//Frame

}//RemaGUM

### Namespace RemaGUM.Properties – ustawienia programu.

namespace RemaGUM.Properties {

using System;

/// <summary>

/// Klasa zasobu wymagająca zdefiniowania typu do wyszukiwania zlokalizowanych ciągów itd.

/// </summary>

// Ta klasa została automatycznie wygenerowana za pomocą klasy StronglyTypedResourceBuilder

// przez narzędzie, takie jak ResGen lub Visual Studio.

// Aby dodać lub usunąć składową, edytuj plik ResX, a następnie ponownie uruchom narzędzie ResGen

// z opcją /str lub ponownie utwórz projekt VS.

global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Resources.Tools.StronglyTypedResourceBuilder", "15.0.0.0")]

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]

[global::System.Runtime.CompilerServices.CompilerGeneratedAttribute()]

internal class Resources {

/// <summary>

/// The resource man

/// </summary>

private static global::System.Resources.ResourceManager resourceMan;

/// <summary>

/// The resource culture

/// </summary>

private static global::System.Globalization.CultureInfo resourceCulture;

/// <summary>

/// Initializes a new instance of the <see cref="Resources"/> class.

/// </summary>

[global::System.Diagnostics.CodeAnalysis.SuppressMessageAttribute("Microsoft.Performance", "CA1811:AvoidUncalledPrivateCode")]

internal Resources() {

}

/// <summary>

/// Zwraca buforowane wystąpienie ResourceManager używane przez tę klasę.

/// </summary>

/// <value>The resource manager.</value>

[global::System.ComponentModel.EditorBrowsableAttribute(global::System.ComponentModel.EditorBrowsableState.Advanced)]

internal static global::System.Resources.ResourceManager ResourceManager {

get {

if (object.ReferenceEquals(resourceMan, null)) {

global::System.Resources.ResourceManager temp = new global::System.Resources.ResourceManager("RemaGUM.Properties.Resources", typeof(Resources).Assembly);

resourceMan = temp;

}

return resourceMan;

}

}

/// <summary>

/// Przesłania właściwość CurrentUICulture bieżącego wątku dla wszystkich

/// przypadków przeszukiwania zasobów za pomocą tej klasy zasobów wymagającej zdefiniowania typu.

/// </summary>

/// <value>The culture.</value>

[global::System.ComponentModel.EditorBrowsableAttribute(global::System.ComponentModel.EditorBrowsableState.Advanced)]

internal static global::System.Globalization.CultureInfo Culture {

get {

return resourceCulture;

}

set {

resourceCulture = value;

}

}

/// <summary>

/// Wyszukuje zlokalizowany zasób typu System.Drawing.Bitmap.

/// </summary>

/// <value>The ikona2.</value>

internal static System.Drawing.Bitmap ikona2 {

get {

object obj = ResourceManager.GetObject("ikona2", resourceCulture);

return ((System.Drawing.Bitmap)(obj));

}

}

/// <summary>

/// Wyszukuje zlokalizowany zasób typu System.Drawing.Bitmap.

/// </summary>

/// <value>The logo gu M2018.</value>

internal static System.Drawing.Bitmap LogoGUM2018 {

get {

object obj = ResourceManager.GetObject("LogoGUM2018", resourceCulture);

return ((System.Drawing.Bitmap)(obj));

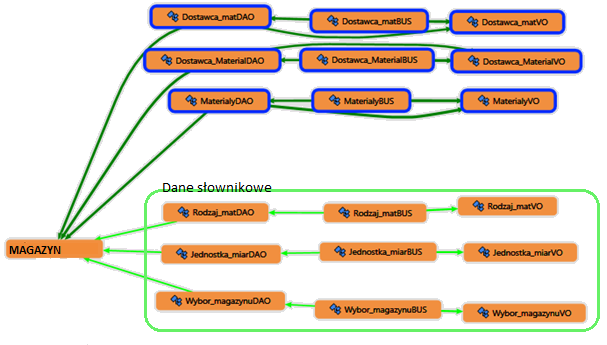
}

}

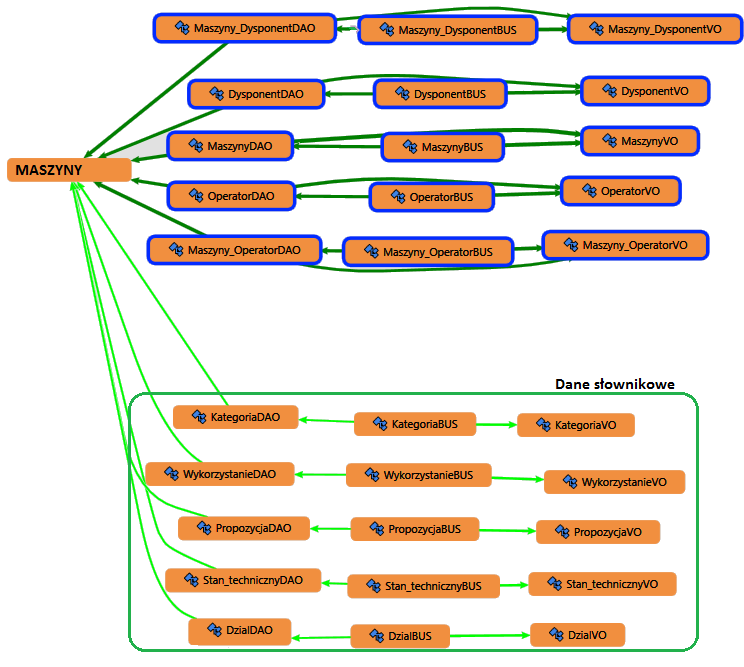
}

}

## Opis funkcji i klas połączenia z bazą danych



*Rys.1 Powiązania klas modułu magazyn.*



*Rys. Powiązania klas modułu maszyny.*

Zawartość projektu

Cały projekt zawarty jest w katalogach:

* **…\RemaGUM** – projekt rozwiązania wytworzonego w środowisku Visual Studio zawierający pełny kod źródłowy **C#** oraz wszystkie niezbędne komponenty do modyfikacji i rozwinięcia, ponownej kompilacji oraz dystrybucji programu.
* **……\RemaGUM \Docs\Help** obejmuje dokumenty stanowiące składniki pomocy bezpośredniej w programie tzw. help.
* **…\RemaGUM \bin\Debug\RemaGUM.mbd** – baza danych MS Access (hasło: zlom).

Tabela **Maszyny** powinna zawierać następujące pozycje:

Identyfikator (klucz własny)

1. **Kategoria**
2. **Nazwa**
3. **Typ**
4. **Nr\_inwentarzowy**
5. **Nr\_fabryczny**
6. **Rok\_produkcji**
7. **Producent**
8. **Zdjecie**
9. **Zawartośc pliku**
10. **Rozszerz\_zdj**
11. **Nazwa\_os\_zarzadzajaca**
12. **Nazwa\_op\_maszyny**
13. **Nr\_pom**
14. **Dzial**
15. **Nr\_prot\_BHP**
16. **Data\_ost\_przegl** (badania BHP)
17. **Data\_kol\_przegl**(badania BHP)
18. **Uwagi**
19. **Wykorzystanie** – jak często jest wykorzystywana (SŁOWNIK -> kilka razy w miesiącu
20. **Stan\_techniczny** (SŁOWNIK -> dobry/ do naprawy / złom)
21. **Propozycja** (SŁOWNIK wymiana/złom)
22. Rok\_ost\_przeg
23. Mc\_ost\_przeg
24. Dz\_ost\_przeg
25. Rok\_kol\_przeg
26. Mc\_kol\_przeg
27. Dz\_kol\_przeg
    * sortowanie tabeli maszyn po:
      + Nazwie,
      + Typie
      + Nr inwentarzowy
      + Nr fabryczny
      + Nr pomieszczenia
      + Data ostatniego przeglądu
      + Data kolejnego przeglądu

Nazwy komponentów Nazwy kolumn i ich typy w tabeli Maszyny

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identyfikator | Liczba |  |
|  | **Kategoria** | Txt 255 | 100 |
|  | **Nazwa** | Txt 255 | 255 |
|  | **Typ** | Txt 255 | 255 |
|  | **Nr\_inwentarzowy** | Txt 255 | 255 |
|  | **Nr\_fabryczny** | Txt 255 | 255 |
|  | **Rok\_produkcji** | Txt 255 | 50 |
|  | **Producent** | Txt 255 | 255 |
|  | **Zdjecie** | Txt 255 | 255 |
|  | **Zawartośc pliku** | ObiektOLE | 255 |
|  | **Rozszerz\_zdj** | Txt 255 | 255 |
|  | **Nazwa\_dysponent** | Txt 255 | 255 |
|  | **Nazwa\_op\_maszyny** | Txt 255 | 255 |
|  | **Nr\_pom** | Txt 255 | 20 |
|  | **Dzial** | Txt 255 | 20 |
|  | **Nr\_prot\_BHP** | Txt 255 | 255 |
|  | **Data\_ost\_przegl** (badania BHP) | data |  |
|  | **Data\_kol\_przegl**(badania BHP) | data |  |
|  | **Uwagi** | Txt słownik | 20 |
|  | **Wykorzystanie** | Txt słownik | 20 |
|  | **Stan\_techniczny** | Txt słownik | 20 |
|  | **Propozycja** | Txt 255 | 100 |
|  | Rok\_ost\_przeg | Int |  |
|  | Mc\_ost\_przeg | Int |  |
|  | Dz\_ost\_przeg | Int |  |
|  | Rok\_kol\_przeg | Int |  |
|  | Mc\_kol\_przeg | Int |  |
|  | Dz\_kol\_przeg | Int |  |

Nazwy komponentów tabeli maszyny

|  |  |  |
| --- | --- | --- |
| sortowanie | groupBoxSortowanie | radioButtonNazwa |
| radioButtonTyp |
| radioButtonNr\_fabryczny |
| radioButtonNr\_inwentarzowy |
| radioButtonNr\_pomieszczenia |
| radioButtonData\_ost\_przegladu |
| wyszukiwanie |  | textBoxWyszukiwanie |
|  | buttonSzukaj |
| Spis dostępnych maszyn | Lista nazw | listBoxMaszyny |
| Dane maszyny | Kategoria | comboBoxKategoria |
| Nazwa | textBoxNazwa |
| Typ | textBoxTyp |
| Rok\_produkcji | textBoxRok\_produkcji |
| Data\_ost\_przegl | dateTimePickerData\_ost\_przegl |
| Data\_kol\_przegl | dateTimePickerData\_kol\_przegl |
| Nr\_inwentarzowy | textBoxNr\_inwentarzowy |
| Nr\_fabryczny | textBoxNr\_fabryczny |
| Producent | textBoxProducent |
| Dysponent | comboBoxDysponent |
| Operator maszyny  WypelnijOperatorow\_maszyny | checkedListBoxOperator\_maszyny ~~comboBoxOperator\_maszyny~~ |
| Nr\_pom | comboBoxNr\_pom |
| Dzial | comboBoxDzial |
| Nr\_prot\_BHP | textBoxNr\_prot\_BHP |
| Uwagi | richTextBoxUwagi |
| zdjęcie | Zdjecie1 - nazwa |  |
| Rozszerz\_zdj1 |  |
| **Zawartość pliku** | pictureBox1 |
|  | buttonPokazZdj |
|  | buttonUsunZdj |
| Otwieranie po linku nazwy zdjęcia (?) | linkLabelNazwaZdjecia |
| Stan techniczny i częstotliwość wykorzystania | Stan\_techniczny | comboBoxStan\_techniczny |
| Wykorzystanie | comboBoxWykorzystanie |
| Propozycja | comboBoxPropozycja |
| Edycja spisu | nowa | buttonNowa |
| zapisz | buttonZapisz |
| anuluj | buttonAnuluj |
| usuń | buttonUsun |

Tabela **Kategoria (dane słownikowe** jaką kategorię stanowi**)** powinna zawierać następujące pozycje:

1. Identyfikator
2. **Kategoria** – (SŁOWNIK -> laboratoryjne / warsztatowe)

Tabela **Czestotliwosc (dane słownikowe** jak często jest wykorzystywana**)** powinna zawierać następujące pozycje:

1. Identyfikator
2. **Czestotliwosc** – (SŁOWNIK -> kilka razy w miesiącu / kwartale / w okresie półrocznym/w roku / rzadziej / nieużywana)

Tabela **Stan\_techniczny(dane słownikowe** jaki stan techniczny) powinna zawierać następujące pozycje:

1. Identyfikator
2. **Stan\_techniczny** (SŁOWNIK -> dobry/ do naprawy / do złomowania)

Tabela **Propozycja (dane słownikowe** jak wykorzystać**)** powinna zawierać następujące pozycje:

1. Identyfikator
2. **Propozycja** – (SŁOWNIK -> zachować, do remontu, do likwidacji)

Tabela **Operator\_maszyny** powinna zawierać następujące pozycje:

1. Identyfikator (ID\_operator) użytkownik maszyny klucz własny
2. Nazwa\_op\_maszyny (Imię i Nazwisko)
3. ID\_dzial
4. Nazwa\_dzial
5. Uprawnienie
6. Data\_konca\_upr (data po której wygasają uprawnienia np. 2 lata)
7. Rok
8. Mc
9. Dzien
10. Op\_nazwisko
11. Op\_imie

|  |  |  |
| --- | --- | --- |
| Identyfikator |  | Int |
| Nazwa\_op\_maszyny | Txt 255 | string |
| Op\_imie | Txt 255 | string |
| Op\_nazwisko | Txt 255 | string |
| ID\_dzial | liczba | Int |
| Nazwa\_dzial | Txt 255 | string |
| Uprawnienie | Txt 255 | string |
| Data\_konca\_upr | Data | string |
| Rok | liczba | Int |
| Mc | liczba | Int |
| Dzien | liczba | Int |

Nazwy komponentów tabeli Operator

|  |  |  |
| --- | --- | --- |
| sortowanie | comboBoxOperator | Nazwisko |
| Dział |
| Uprawnienia |
| Data końca uprawnień |
| wyszukiwanie |  | textBoxWyszukiwanieOperator |
|  | buttonSzukajOperator |
| Spis operatorów maszyn | Lista nazw  listBoxOperator\_maszyny\_SelectedIndexChanged | listBoxOperator |
| Dane operatora maszyny | Op\_imie | textBoxImieOperator |
| Op\_nazwisko | textBoxNazwiskoOperator |
| Dzial  comboBoxDzial\_operator\_maszyny\_SelectedIndexChanged | comboBoxDzialOperator |
| Uprawnienie | textBoxUprawnienieOperator |
| Data końca uprawnień | dateTimePickerDataKoncaUprOperator |
| Lista obsługiwanych przez wybranego operatora maszyn | listBoxMaszynyOperatora |
|  | ID operatora | toolStripStatusLabelIDOperatora |
| Edycja spisu | nowa | buttonNowaOperator |
| zapisz | buttonZapiszOperator |
| anuluj | buttonAnulujOperator |
| usuń | buttonUsunOperator |

Tabela **Operator\_maszyny\_Maszyny (multi)** określa prawa do użytkowania maszyn, powinna zawierać następujące pozycje:

1. Identyfikator (klucz własny)
2. ID\_masz (klucz obcy)
3. ID\_op\_maszyny (klucz obcy)
4. Maszyny\_nazwa

**tabela Dysponenci maszyn** (opiekunowie) określa prawa do użytkowania maszyn powinna zawierać następujące pozycje:

1. Identyfikator (ID\_dysponent) opiekuna maszyny (klucz własny)
2. ID\_dzial
3. Nazwa\_dzial
4. Dysp\_nazwisko
5. Dysp\_imie
6. Dysp\_dane (dane kontaktowe, stanowisko)
7. Dysp\_nazwa

|  |  |  |
| --- | --- | --- |
| Identyfikator |  | Int |
| Dzial | liczba | Int |
| Dysp\_imie | Txt 255 | string |
| Dysp\_nazwisko | Txt 255 | string |
| Dysp\_dane | Txt 255 | string |
| Dysp\_nazwa | Txt 255 | string |

Nazwy komponentów zakładki Dysponenci maszyn

|  |  |  |
| --- | --- | --- |
| wyszukiwanie |  | textBoxWyszukiwanieDysponent |
|  | buttonSzukajDysponent |
| Spis dysponentów maszyn | Lista nazw  listBoxDysponent\_SelectedIndexChanged | listBoxDysponent |
| Dane dysponenta maszyny | Dysp\_imie | textBoxImieDysponent |
| Dysp\_nazwisko | textBoxNazwiskoDysponent |
| Dzial comboBoxDzial\_dysponent\_SelectedIndexChanged | comboBoxDzialDysponent |
| Dysp\_dane | richTextBoxDysponent\_dane |
| lista maszyn z tabeli maszyny | listBoxMaszynyDysponenta |
|  | Identyfikator | toolStripStatusLabelIDdysponenta |
| Edycja spisu | nowa | buttonNowaDysponent |
| zapisz | buttonZapiszDysponent |
| anuluj | buttonAnulujDysponent |
| usuń | buttonUsunDysponent |

Tabela M**aszyny\_Dysponent (multi)** - nadmiarowa – do wykorzystania gdy kilku dysponentów 1 maszyny – obecnie nie ma takiej potrzeby. Określa dysponentów maszyn, powinna zawierać następujące pozycje:

1. Identyfikator (klucz własny)
2. ID\_masz (klucz obcy)
3. ID\_dysponent (klucz obcy)

Tabela **Materiały** powinna zawierać następujące pozycje:

1. Identyfikator (ID\_materiału) (klucz własny)
2. **Nazwa\_mat** - nazwa materiału
3. **Typ\_mat** - typ (np. blacha, rury)
4. **Rodzaj\_mat** - rodzaj (dane słownikowe np. mosiężne, aluminiowe itp)
5. **Jednostka\_miar\_mat** - wartość jednostkowa
6. **Stan\_mat -** Ilość na stanie magazynowym (szt/m)
7. **Zuzycie\_mat** - bieżące zużycie
8. **Odpad\_mat -** Odpad
9. **Stan\_min\_mat** - Stan minimalny materiału (szt/m)
10. **Zapotrzebowanie\_mat** – bieżące zapotrzebowanie
11. **Stan\_mag\_po\_mat** - stan magazynowy po zużyciu/zapotrzebowaniu
12. **Dostawca\_mat -** (dane dostawcy nazwa, link do strony, Dodatkowe informacje)

Tabela **Dostawca\_mat** powinna zawierać następujące pozycje:

1. Identyfikator (ID\_dostawcy) (klucz własny)
2. Nazwa\_dostawca\_mat – nazwa dostawcy materiału
3. Link\_dostawca\_mat – link do strony dostawcy materiału
4. Dod\_info\_dostawca\_mat – dodatkowe informacje np. upusty o sposobie dostawy/płatności itp.

|  |  |  |
| --- | --- | --- |
| Identyfikator |  | Int |
| **Nazwa\_mat** | Txt 255 | string |
| **Typ\_mat** | Txt 255 | string |
| **Rodzaj\_mat** | Txt 255 | string |
| **Jednostka\_miar\_mat** | Txt 255 | string |
| **Stan\_mat** | liczba | Int |
| **Zuzycie\_mat** | liczba | Int |
| **Odpad\_mat** | liczba | Int |
| **Stan\_min\_mat** | liczba | Int |
| **Zapotrzebowanie\_mat** | liczba | Int |
| **Stan\_mag\_po\_mat** | liczba | Int |
| **Dostawca\_mat** | Txt 255 | string |

Tabela multi **Dostawca\_Materiał** powinna zawierać następujące pozycje:

1. Identyfikator (ID\_dostawcy) (klucz własny)
2. ID\_material – id materiału (klucz obcy tabeli materiały)
3. ID\_dostawca\_mat – id dostawcy materiału (klucz obcy tabeli dostawcy)

Program powinien umożliwiać:

* + Informowanie o niskim stanie magazynowym,
  + dodawanie, odejmowanie, edycję pozycji spisu materiałów,
  + wyszukiwanie po ciągu znaków w nazwie materiału,
  + sortowanie tabeli magazynu po:
    - Nazwie,
    - Typie
    - Cenie
    - Ilości na stanie.

Nazwy komponentów tabeli Materiały

|  |  |  |
| --- | --- | --- |
| sortowanie | groupBoxSortowanie | radioButtonNazwa\_mat |
| radioButtonTyp\_mat |
| radioButtonCena\_mat |
| radioButtonMagazyn\_ilosc\_mat |
| wyszukiwanie |  | textBoxWyszukaj\_mat |
|  | buttonSzukaj\_mat |
| Spis dostępnych materiałów | Lista nazw | listBoxMaterialy |
|  |  |  |
| Dane materiału | Typ | textBoxTypMat |
| Rodzaj | comboBoxRodzajMat |
| Nazwa Materiału | textBoxNazwaMat |
| Wartość jednostkowa | comboBoxJednostkaMat |
| Stan magazynowy materiałów | textBoxMagazynMat |
| Bieżące zużycie | textBoxZuzycieMat |
| Odpad | textBoxOdpadMat |
| Stan minimalny | textBoxMinMat |
| Zapotrzebowanie | textBoxZapotrzebowanieMat |
| Dane dostawców materiałów | Dostawcy | checkedListBoxDostawcyMat |
| Link do strony dostawcy głównego | linkLabelDostawca |
| Dodatkowe informacje | richTextBoxDostawca |
| Edycja spisu | nowa | buttonNowaMat |
| zapisz | buttonZapiszMat |
| anuluj | buttonAnulujMat |
| usuń | buttonUsunMat |

Nazwy komponentów tabeli Normalia

Tabela **Normalia** powinna zawierać następujące pozycje:

1. Identyfikator (ID\_norm.) (klucz własny),
2. **Nazwa\_norm ,**
3. **Typ\_norm,**
4. **Rodzaj\_norm** - rodzaj (dane słownikowe np. mosiężne, aluminiowe itp.,
5. **Jednostka\_miar\_norm** - wartość jednostkowa,
6. **Stan\_norm -** Ilość na stanie magazynowym (szt/m),
7. **Zuzycie\_norm** - bieżące zużycie,
8. **Odpad\_norm –** Odpad,
9. **Stan\_min\_norm** - Stan minimalny normaliów (szt/m),
10. **Zapotrzebowanie\_norm** – bieżące zapotrzebowanie na normalia,
11. **Stan\_mag\_po\_norm** - stan magazynowy po zużyciu/zapotrzebowaniu,
12. **Dostawca\_norm -** (dane dostawcy nazwa, link do strony, dodatkowe informacje).

Tabela **Dostawca\_norm** powinna zawierać następujące pozycje:

1. Identyfikator (ID\_dostawcy) (klucz własny)
2. Nazwa\_dostawca\_norm – nazwa dostawcy normaliów
3. Link\_dostawca\_norm – link do strony dostawcy normaliów
4. Dod\_info\_dostawca\_norm – dodatkowe informacje np. upusty o sposobie dostawy/płatności itp.

|  |  |  |
| --- | --- | --- |
| Identyfikator |  | Int |
| **Nazwa\_norm** | Txt 255 | string |
| **Typ\_norm** | Txt 255 | string |
| **Rodzaj\_norm** | Txt 255 | string |
| **Jednostka\_miar\_norm** | Txt 255 | string |
| **Stan\_norm** | liczba | Int |
| **Zuzycie\_norm** | liczba | Int |
| **Odpad\_norm** | liczba | Int |
| **Stan\_min\_norm** | liczba | Int |
| **Zapotrzebowanie\_norm** | liczba | Int |
| **Stan\_mag\_po\_norm** | liczba | Int |
| **Dostawca\_norm** | Txt 255 | string |

Tabela multi **Dostawca\_Normalia** powinna zawierać następujące pozycje:

1. Identyfikator (ID\_dostawcy) (klucz własny)
2. ID\_normalia – id normaliów (klucz obcy tabeli normalia)
3. ID\_dostawca\_norm – id dostawcy materiału (klucz obcy tabeli dostawcy)

Program powinien umożliwiać:

* + Informowanie o niskim stanie magazynowym,
  + dodawanie, odejmowanie, edycję pozycji spisu materiałów,
  + wyszukiwanie po ciągu znaków w nazwie materiału,
  + sortowanie tabeli magazynu po:
    - Nazwie,
    - Typie
    - Cenie
    - Ilości na stanie.

|  |  |  |
| --- | --- | --- |
| sortowanie | groupBoxSortowanie | radioButtonNazwaNorm |
| radioButtonTypNorm |
| radioButtonCenaNorm |
| radioButtonIloscNorm |
| wyszukiwanie |  | textBoxSzukajNorm |
|  | buttonSzukajNorm |
| Spis dostępnych normaliów | Lista nazw | listBoxNormalia |
| Normalia | Typ | textBoxTypNorm |
| Rodzaj | comboBoxRodzajNorm |
| Nazwa Materiału | textBoxNazwaNorm |
| Wartość jednostkowa | comboBoxJednostkaNorm |
| Stan magazynowy normaliów | textBoxMagazynNorm |
| Bieżące zużycie | textBoxZuzycieNorm |
| Stan minimalny | textBoxStanMinNorm |
| Zapotrzebowanie | textBoxZapotrzebowanieNorm |
| Dane dostawcy głównego normaliów | Dostawcy normaliów | listViewDostawcyNorm |
| Link do strony dostawcy normaliów | linkLabelDostawcaNorm |
| Dodatkowe informacje o dostawcy | richTextBoxDostawcaNorm |
| Edycja spisu | nowa | buttonNowaNorm |
| zapisz | buttonZapiszNorm |
| anuluj | buttonAnulujNorm |
| usuń | buttonUsunNorm |

Spis treści

[1. Zapotrzebowanie 1](#_Toc535240071)

[2. Cel powstania 1](#_Toc535240072)

[3. Opcje kompilacji 1](#_Toc535240073)

[5. Opis funkcji i klas 2](#_Toc535240074)

[5.1. Klasa Main – start aplikacji. 2](#_Toc535240075)

[5.2. Klasa Rest – hasło bazy danych MSACCESS. 2](#_Toc535240076)

[5.3. Obsługa działania programu – Klasa główna. 3](#_Toc535240077)

[5.4. Komponenty wizualne – okienko „o RemaGUM”. 21](#_Toc535240078)

[5.5. Komponenty wizualne- Klasa Frame 21](#_Toc535240079)

[5.6. Namespace RemaGUM.Properties – ustawienia programu. 23](#_Toc535240080)