Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Кафедра **«**Информационные технологии и автоматизированные системы**»**

**ОТЧЕТ**

**ПО ТВОРЧЕСКОЙ РАБОТЕ №1**

Дисциплина: «Основы алгоритмизации и программирования»

Семестр 2

Тема: АРМ разработчика компьютерной игры

Выполнил работу

Студент группы РИС-22-1Б

Лихачев Д.А.

Проверил

Доцент кафедры ИТАС

Полякова О.А.

г. Пермь-2023

**Введение**

Для выполнения лабораторной работы требуется сделать отчет программы. Создание программ – отличный способ практики программирования.

**Постановка задачи**

* Реализация приложения на основе Windows Forms, облегчающего работу разработчика.
* Изучить структуру библиотечной отрасли.
* Реализовать функции облегчающие задачи библиотекарей (выдача книг, прием книг, добавление читателя в базу данных, добавление книги в базу данных)

**Код программы**

**RegForm.h**

#pragma once

#include "MenuForm.h"

namespace liba {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для RegForm

/// </summary>

public ref class RegForm : public System::Windows::Forms::Form

{

private:

public:

RegForm(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~RegForm()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::TextBox^ login\_textBox;

private: System::Windows::Forms::TextBox^ password\_textBox;

protected:

protected:

private: System::Windows::Forms::Label^ label1;

private: System::Windows::Forms::Label^ label2;

private: System::Windows::Forms::Button^ enter\_button;

private: System::Windows::Forms::GroupBox^ groupBox1;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(RegForm::typeid));

this->login\_textBox = (gcnew System::Windows::Forms::TextBox());

this->password\_textBox = (gcnew System::Windows::Forms::TextBox());

this->label1 = (gcnew System::Windows::Forms::Label());

this->label2 = (gcnew System::Windows::Forms::Label());

this->enter\_button = (gcnew System::Windows::Forms::Button());

this->groupBox1 = (gcnew System::Windows::Forms::GroupBox());

this->groupBox1->SuspendLayout();

this->SuspendLayout();

//

// login\_textBox

//

this->login\_textBox->Location = System::Drawing::Point(58, 31);

this->login\_textBox->Name = L"login\_textBox";

this->login\_textBox->Size = System::Drawing::Size(122, 20);

this->login\_textBox->TabIndex = 0;

//

// password\_textBox

//

this->password\_textBox->Location = System::Drawing::Point(58, 70);

this->password\_textBox->Name = L"password\_textBox";

this->password\_textBox->PasswordChar = '\*';

this->password\_textBox->Size = System::Drawing::Size(122, 20);

this->password\_textBox->TabIndex = 1;

//

// label1

//

this->label1->AutoSize = true;

this->label1->Location = System::Drawing::Point(6, 34);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(41, 13);

this->label1->TabIndex = 2;

this->label1->Text = L"Логин:";

//

// label2

//

this->label2->AutoSize = true;

this->label2->Location = System::Drawing::Point(4, 73);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(48, 13);

this->label2->TabIndex = 3;

this->label2->Text = L"Пароль:";

//

// enter\_button

//

this->enter\_button->Location = System::Drawing::Point(5, 119);

this->enter\_button->Name = L"enter\_button";

this->enter\_button->Size = System::Drawing::Size(174, 20);

this->enter\_button->TabIndex = 4;

this->enter\_button->Text = L"Войти";

this->enter\_button->UseVisualStyleBackColor = true;

this->enter\_button->Click += gcnew System::EventHandler(this, &RegForm::enter\_button\_Click);

//

// groupBox1

//

this->groupBox1->Controls->Add(this->enter\_button);

this->groupBox1->Controls->Add(this->label2);

this->groupBox1->Controls->Add(this->label1);

this->groupBox1->Controls->Add(this->password\_textBox);

this->groupBox1->Controls->Add(this->login\_textBox);

this->groupBox1->Location = System::Drawing::Point(7, 12);

this->groupBox1->Name = L"groupBox1";

this->groupBox1->Size = System::Drawing::Size(186, 148);

this->groupBox1->TabIndex = 5;

this->groupBox1->TabStop = false;

this->groupBox1->Text = L"Авторизация";

//

// RegForm

//

this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(199, 163);

this->Controls->Add(this->groupBox1);

this->Icon = (cli::safe\_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));

this->Name = L"RegForm";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"АРМ";

this->groupBox1->ResumeLayout(false);

this->groupBox1->PerformLayout();

this->ResumeLayout(false);

}

#pragma endregion

private: System::Void enter\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

if (login\_textBox->Text == "admin" && password\_textBox->Text == "admin") {

MenuForm^ f1 = gcnew MenuForm();

f1->Show();

RegForm::Hide();

}

else {

MessageBox::Show("Неверный логин или пароль!");

}

}

};

}

**RegForm.h**

#include "RegForm.h"

#include "MenuForm.h"

#include <Windows.h>

using namespace System;

using namespace System::Windows::Forms;

[STAThreadAttribute]

int main(array<String^>^ args) {

Application::SetCompatibleTextRenderingDefault(false);

Application::EnableVisualStyles();

liba::RegForm form;

Application::Run(% form);

}

**ReaderForm.h**

#pragma once

#include "Reader.h"

#include <fstream>

#include <iostream>

#include <vector>

#include <msclr\marshal\_cppstd.h>

#include <filesystem>

#include <cstdio>

#include <vcclr.h>

namespace liba {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

using namespace System::Collections::Generic;

using namespace msclr::interop;

/// <summary>

/// Сводка для ReaderForm

/// </summary>

public ref class ReaderForm : public System::Windows::Forms::Form

{

private: std::vector<std::string>\* readers;

private: System::Windows::Forms::ListBox^ info\_list;

private: System::Windows::Forms::GroupBox^ groupBox1;

private: System::Windows::Forms::Label^ label3;

private: System::Windows::Forms::TextBox^ search\_line;

private: System::Windows::Forms::Button^ search\_button;

private: System::Windows::Forms::Label^ label4;

List<String^>^ parts;

public:

ReaderForm(void)

{

InitializeComponent();

readers = new std::vector<std::string>;

LoadList();

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~ReaderForm()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::TextBox^ fio\_textBox;

protected:

private: System::Windows::Forms::TextBox^ passNum\_textBox;

private: System::Windows::Forms::Label^ label1;

private: System::Windows::Forms::Label^ label2;

private: System::Windows::Forms::ListBox^ readerslistBox;

private: System::Windows::Forms::Button^ Registr\_button;

private: System::Windows::Forms::Button^ delete\_button;

private: System::Windows::Forms::Button^ save\_button;

private: System::Windows::Forms::Button^ info\_button;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(ReaderForm::typeid));

this->fio\_textBox = (gcnew System::Windows::Forms::TextBox());

this->passNum\_textBox = (gcnew System::Windows::Forms::TextBox());

this->label1 = (gcnew System::Windows::Forms::Label());

this->label2 = (gcnew System::Windows::Forms::Label());

this->readerslistBox = (gcnew System::Windows::Forms::ListBox());

this->Registr\_button = (gcnew System::Windows::Forms::Button());

this->delete\_button = (gcnew System::Windows::Forms::Button());

this->save\_button = (gcnew System::Windows::Forms::Button());

this->info\_button = (gcnew System::Windows::Forms::Button());

this->info\_list = (gcnew System::Windows::Forms::ListBox());

this->groupBox1 = (gcnew System::Windows::Forms::GroupBox());

this->label3 = (gcnew System::Windows::Forms::Label());

this->search\_line = (gcnew System::Windows::Forms::TextBox());

this->search\_button = (gcnew System::Windows::Forms::Button());

this->label4 = (gcnew System::Windows::Forms::Label());

this->groupBox1->SuspendLayout();

this->SuspendLayout();

//

// fio\_textBox

//

this->fio\_textBox->Location = System::Drawing::Point(63, 34);

this->fio\_textBox->Name = L"fio\_textBox";

this->fio\_textBox->Size = System::Drawing::Size(156, 20);

this->fio\_textBox->TabIndex = 0;

//

// passNum\_textBox

//

this->passNum\_textBox->Location = System::Drawing::Point(63, 60);

this->passNum\_textBox->Name = L"passNum\_textBox";

this->passNum\_textBox->Size = System::Drawing::Size(156, 20);

this->passNum\_textBox->TabIndex = 1;

//

// label1

//

this->label1->AutoSize = true;

this->label1->Location = System::Drawing::Point(6, 37);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(34, 13);

this->label1->TabIndex = 2;

this->label1->Text = L"ФИО";

//

// label2

//

this->label2->AutoSize = true;

this->label2->Location = System::Drawing::Point(4, 63);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(50, 13);

this->label2->TabIndex = 3;

this->label2->Text = L"Паспорт";

//

// readerslistBox

//

this->readerslistBox->FormattingEnabled = true;

this->readerslistBox->Location = System::Drawing::Point(8, 192);

this->readerslistBox->Name = L"readerslistBox";

this->readerslistBox->Size = System::Drawing::Size(719, 264);

this->readerslistBox->TabIndex = 4;

//

// Registr\_button

//

this->Registr\_button->Location = System::Drawing::Point(107, 95);

this->Registr\_button->Name = L"Registr\_button";

this->Registr\_button->Size = System::Drawing::Size(112, 23);

this->Registr\_button->TabIndex = 5;

this->Registr\_button->Text = L"Зарегестрировать";

this->Registr\_button->UseVisualStyleBackColor = true;

this->Registr\_button->Click += gcnew System::EventHandler(this, &ReaderForm::Registr\_button\_Click);

//

// delete\_button

//

this->delete\_button->Location = System::Drawing::Point(553, 462);

this->delete\_button->Name = L"delete\_button";

this->delete\_button->Size = System::Drawing::Size(85, 33);

this->delete\_button->TabIndex = 6;

this->delete\_button->Text = L"Удалить";

this->delete\_button->UseVisualStyleBackColor = true;

this->delete\_button->Click += gcnew System::EventHandler(this, &ReaderForm::delete\_button\_Click);

//

// save\_button

//

this->save\_button->Location = System::Drawing::Point(642, 462);

this->save\_button->Name = L"save\_button";

this->save\_button->Size = System::Drawing::Size(85, 33);

this->save\_button->TabIndex = 7;

this->save\_button->Text = L"Сохранить";

this->save\_button->UseVisualStyleBackColor = true;

this->save\_button->Click += gcnew System::EventHandler(this, &ReaderForm::save\_button\_Click);

//

// info\_button

//

this->info\_button->Location = System::Drawing::Point(8, 462);

this->info\_button->Name = L"info\_button";

this->info\_button->Size = System::Drawing::Size(85, 33);

this->info\_button->TabIndex = 9;

this->info\_button->Text = L"Информация";

this->info\_button->UseVisualStyleBackColor = true;

this->info\_button->Click += gcnew System::EventHandler(this, &ReaderForm::info\_button\_Click);

//

// info\_list

//

this->info\_list->FormattingEnabled = true;

this->info\_list->Location = System::Drawing::Point(246, 33);

this->info\_list->Name = L"info\_list";

this->info\_list->Size = System::Drawing::Size(481, 108);

this->info\_list->TabIndex = 10;

//

// groupBox1

//

this->groupBox1->Controls->Add(this->Registr\_button);

this->groupBox1->Controls->Add(this->label2);

this->groupBox1->Controls->Add(this->label1);

this->groupBox1->Controls->Add(this->passNum\_textBox);

this->groupBox1->Controls->Add(this->fio\_textBox);

this->groupBox1->Location = System::Drawing::Point(8, 12);

this->groupBox1->Name = L"groupBox1";

this->groupBox1->Size = System::Drawing::Size(232, 125);

this->groupBox1->TabIndex = 11;

this->groupBox1->TabStop = false;

this->groupBox1->Text = L"Регистрация";

//

// label3

//

this->label3->AutoSize = true;

this->label3->Location = System::Drawing::Point(433, 12);

this->label3->Name = L"label3";

this->label3->Size = System::Drawing::Size(124, 13);

this->label3->TabIndex = 12;

this->label3->Text = L"Информационное окно";

//

// search\_line

//

this->search\_line->Location = System::Drawing::Point(8, 160);

this->search\_line->Multiline = true;

this->search\_line->Name = L"search\_line";

this->search\_line->Size = System::Drawing::Size(317, 23);

this->search\_line->TabIndex = 13;

this->search\_line->TextChanged += gcnew System::EventHandler(this, &ReaderForm::search\_line\_TextChanged);

//

// search\_button

//

this->search\_button->Location = System::Drawing::Point(331, 160);

this->search\_button->Name = L"search\_button";

this->search\_button->Size = System::Drawing::Size(88, 23);

this->search\_button->TabIndex = 14;

this->search\_button->Text = L"Поиск";

this->search\_button->UseVisualStyleBackColor = true;

this->search\_button->Click += gcnew System::EventHandler(this, &ReaderForm::search\_button\_Click);

//

// label4

//

this->label4->AutoSize = true;

this->label4->Location = System::Drawing::Point(9, 144);

this->label4->Name = L"label4";

this->label4->Size = System::Drawing::Size(39, 13);

this->label4->TabIndex = 15;

this->label4->Text = L"Поиск";

//

// ReaderForm

//

this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(734, 500);

this->Controls->Add(this->label4);

this->Controls->Add(this->search\_button);

this->Controls->Add(this->search\_line);

this->Controls->Add(this->label3);

this->Controls->Add(this->groupBox1);

this->Controls->Add(this->info\_list);

this->Controls->Add(this->info\_button);

this->Controls->Add(this->save\_button);

this->Controls->Add(this->delete\_button);

this->Controls->Add(this->readerslistBox);

this->Icon = (cli::safe\_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));

this->Name = L"ReaderForm";

this->RightToLeftLayout = true;

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"АРМ";

this->groupBox1->ResumeLayout(false);

this->groupBox1->PerformLayout();

this->ResumeLayout(false);

this->PerformLayout();

}

#pragma endregion

private: System::Void Registr\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

std::string fio = msclr::interop::marshal\_as<std::string>(fio\_textBox->Text);

std::string passNum = msclr::interop::marshal\_as<std::string>(passNum\_textBox->Text);

Reader reader(fio, passNum);

std::string str = reader.toString();

(\*readers).push\_back(str);

//создание txt профиля каждого читателя

std::string filename = "";

std::wstring cyrillicDigits = L"0123456789";

for (wchar\_t c : str) {

if (cyrillicDigits.find(c) != std::wstring::npos) {

filename += c;

}

}

std::string path = "users\\" + filename + ".txt";

std::ofstream file(path);

file.close();

UpdateList();

//Сохранение данных в Читатели.txt

std::ofstream log("database\\Читатели.txt");

for (int i = 0; i < readerslistBox->Items->Count; i++)

log << msclr::interop::marshal\_as<std::string>(readerslistBox->Items[i]->ToString()) << std::endl;

log.close();

}

private: System::Void delete\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

int index = readerslistBox->SelectedIndex;

int Size = (\*readers).size();

//УДАЛЕНИЕ TXT ПРОФИЛЬ ФАЙЛА

std::string str = (\*readers)[index];

std::string filename = "";

std::wstring cyrillicDigits = L"0123456789";

for (wchar\_t c : str) {

if (cyrillicDigits.find(c) != std::wstring::npos) {

filename += c;

}

}

std::string path = "users\\" + filename + ".txt";

std::remove(path.c\_str());

//УДАЛЕНИЕ ИЗ СПИСКА

if (index >= 0 && index < Size)

{

(\*readers).erase((\*readers).begin() + index);

UpdateList();

}

//СОХРАНЕНИЕ СПИСКА

std::ofstream log("database\\Читатели.txt");

for (int i = 0; i < readerslistBox->Items->Count; i++)

log << msclr::interop::marshal\_as<std::string>(readerslistBox->Items[i]->ToString()) << std::endl;

log.close();

}

private: System::Void UpdateList() {

readerslistBox->Items->Clear();

for (int i = 0; i < (\*readers).size(); i++)

readerslistBox->Items->Add(msclr::interop::marshal\_as<String^>((\*readers)[i]));

}

private: void LoadList()

{

if (std::filesystem::exists("database\\Читатели.txt"))

{

std::ifstream file("database\\Читатели.txt");

std::string str = "";

while (getline(file, str)) {

(\*readers).push\_back(str);

readerslistBox->Items->Add(msclr::interop::marshal\_as<String^>(str));

}

file.close();

}

}

private: System::Void save\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

std::ofstream log("database\\Читатели.txt");

for (int i = 0; i < readerslistBox->Items->Count; i++)

log << msclr::interop::marshal\_as<std::string>(readerslistBox->Items[i]->ToString()) << std::endl;

log.close();

}

/\*private: System::Void check\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

for (int i = 0; i < readerslistBox->Items->Count; i++) {

Object^ selectedItemi = readerslistBox->Items[i];

String^ selectedItem = dynamic\_cast<String^>(selectedItemi);

std::string str = msclr::interop::marshal\_as<std::string>(selectedItem);

std::string filename = "";

std::wstring cyrillicDigits = L"0123456789";

for (wchar\_t c : str) {

if (cyrillicDigits.find(c) != std::wstring::npos) {

filename += c;

}

}

std::string path = "users\\" + filename + ".txt";

std::ifstream filePers(path);

std::string str1;

getline(filePers, str1);

String^ str2 = msclr::interop::marshal\_as<String^>(str1);

array<String^>^ parts = str2->Split(';');

DateTime dateTime;

for (int j = 1; j < sizeof(parts); j += 2) {

dateTime = DateTime::Parse(msclr::interop::marshal\_as<String^>(parts[j]));

//////////////////////////////////////////////////

}

}

}\*/

private: System::Void info\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

if (info\_list->Items->Count != 0) {

info\_list->Items->Clear();

}

int index = readerslistBox->SelectedIndex;

std::string str = (\*readers)[index];

std::string filename = "";

std::wstring cyrillicDigits = L"0123456789";

for (wchar\_t c : str) {

if (cyrillicDigits.find(c) != std::wstring::npos) {

filename += c;

}

}

std::string path = "users\\" + filename + ".txt";

std::ifstream filePers;

filePers.open(path, std::ios::app);

std::string books\_;

getline(filePers, books\_);

String^ books = msclr::interop::marshal\_as<String^>(books\_);

parts = gcnew List<String^>(books->Split(';'));

if (parts->Count - 1 != 0) {

for (int i = 0; i < parts->Count - 1; i += 2) {

info\_list->Items->Add("Книга: " + parts[i]);

info\_list->Items->Add("Сдать до " + parts[i + 1] + "\n");

}

}

else {

info\_list->Items->Add("Книг не зарегистрировано!");

}

}

private: System::Void search\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

String^ searchQuery = search\_line->Text; // Получение текста поискового запроса

readerslistBox->Items->Clear(); // Очистка ListBox

for (int i = 0; i < readerslistBox->Items->Count; i++) {

String^ currentItem = readerslistBox->Items[i]->ToString();

// Проверка, содержит ли текущая строка поисковый запрос

if (currentItem->Contains(searchQuery)) {

readerslistBox->Items->Add(currentItem); // Добавление строки в ListBox

}

}

if (readerslistBox->Items->Count == 0) {

MessageBox::Show("Совпадений не найдено.");

UpdateList();

}

}

private: void FilterListBoxItems(ListBox ^ listBox, String ^ filterText){

// Очистка текущего списка элементов

listBox->Items->Clear();

// Получение исходного списка элементов

array<Object^>^ items = gcnew array<Object^>(listBox->Items->Count);

listBox->Items->CopyTo(items, 0);

// Проверка каждого элемента списка

for each (Object ^ item in items)

{

// Приведение элемента списка к типу строки

String^ itemText = dynamic\_cast<String^>(item);

// Проверка наличия неполного названия в элементе списка

if (itemText->IndexOf(filterText, StringComparison::CurrentCultureIgnoreCase) >= 0)

{

// Добавление элемента в список, если неполное название найдено

listBox->Items->Add(itemText);

}

}

}

private: System::Void search\_line\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

// Приведение отправителя события к типу TextBox

TextBox^ textBox = dynamic\_cast<TextBox^>(sender);

// Получение неполного названия для фильтрации

String^ filterText = textBox->Text;

// Вызов функции для фильтрации элементов ListBox

FilterListBoxItems(readerslistBox, filterText);

}

};

}

**Reader.h**

#pragma once

#include <iostream>

#include <string>

public class Reader

{

private:

std::string fio;

std::string passNum;

public:

Reader()

{

this->fio = "";

this->passNum = "";

}

Reader(std::string fio, std::string passNum)

{

this->fio = fio;

this->passNum = passNum;

}

std::string get\_fio() { return this->fio; }

std::string get\_passNum() { return this->passNum; }

std::string toString()

{

return this->fio + " (Passport: " + this->passNum + " )";

}

};

**MenuForm.h**

#pragma once

#include "Form1.h"

#include "ReaderForm.h"

#include "GiveForm.h"

#include "AcceptForm.h"

namespace liba {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для MenuForm

/// </summary>

public ref class MenuForm : public System::Windows::Forms::Form

{

private:

ReaderForm^ rf = gcnew ReaderForm();

GiveForm^ gf = gcnew GiveForm();

Form1^ f1 = gcnew Form1();

AcceptForm^ af = gcnew AcceptForm();

private: System::Windows::Forms::MonthCalendar^ monthCalendar1;

public:

MenuForm(void)

{

InitializeComponent();

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MenuForm()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::ContextMenuStrip^ contextMenuStrip1;

private: System::Windows::Forms::MenuStrip^ menuStrip1;

private: System::Windows::Forms::ToolStripMenuItem^ складToolStripMenuItem;

private: System::Windows::Forms::ToolStripMenuItem^ читателиToolStripMenuItem;

private: System::Windows::Forms::Button^ acceptbook\_button;

private: System::Windows::Forms::Button^ givebook\_button;

private: System::Windows::Forms::GroupBox^ groupBox1;

private: System::Windows::Forms::Timer^ timer1;

protected:

private: System::ComponentModel::IContainer^ components;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

this->components = (gcnew System::ComponentModel::Container());

System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(MenuForm::typeid));

this->contextMenuStrip1 = (gcnew System::Windows::Forms::ContextMenuStrip(this->components));

this->menuStrip1 = (gcnew System::Windows::Forms::MenuStrip());

this->складToolStripMenuItem = (gcnew System::Windows::Forms::ToolStripMenuItem());

this->читателиToolStripMenuItem = (gcnew System::Windows::Forms::ToolStripMenuItem());

this->acceptbook\_button = (gcnew System::Windows::Forms::Button());

this->givebook\_button = (gcnew System::Windows::Forms::Button());

this->groupBox1 = (gcnew System::Windows::Forms::GroupBox());

this->timer1 = (gcnew System::Windows::Forms::Timer(this->components));

this->monthCalendar1 = (gcnew System::Windows::Forms::MonthCalendar());

this->menuStrip1->SuspendLayout();

this->groupBox1->SuspendLayout();

this->SuspendLayout();

//

// contextMenuStrip1

//

this->contextMenuStrip1->Name = L"contextMenuStrip1";

this->contextMenuStrip1->Size = System::Drawing::Size(61, 4);

//

// menuStrip1

//

this->menuStrip1->Items->AddRange(gcnew cli::array< System::Windows::Forms::ToolStripItem^ >(2) {

this->складToolStripMenuItem,

this->читателиToolStripMenuItem

});

this->menuStrip1->Location = System::Drawing::Point(0, 0);

this->menuStrip1->Name = L"menuStrip1";

this->menuStrip1->Size = System::Drawing::Size(220, 24);

this->menuStrip1->TabIndex = 1;

this->menuStrip1->Text = L"menuStrip1";

//

// складToolStripMenuItem

//

this->складToolStripMenuItem->Name = L"складToolStripMenuItem";

this->складToolStripMenuItem->Size = System::Drawing::Size(52, 20);

this->складToolStripMenuItem->Text = L"Склад";

this->складToolStripMenuItem->Click += gcnew System::EventHandler(this, &MenuForm::складToolStripMenuItem\_Click);

//

// читателиToolStripMenuItem

//

this->читателиToolStripMenuItem->Name = L"читателиToolStripMenuItem";

this->читателиToolStripMenuItem->Size = System::Drawing::Size(70, 20);

this->читателиToolStripMenuItem->Text = L"Читатели";

this->читателиToolStripMenuItem->Click += gcnew System::EventHandler(this, &MenuForm::читателиToolStripMenuItem\_Click);

//

// acceptbook\_button

//

this->acceptbook\_button->Location = System::Drawing::Point(16, 87);

this->acceptbook\_button->Name = L"acceptbook\_button";

this->acceptbook\_button->Size = System::Drawing::Size(161, 34);

this->acceptbook\_button->TabIndex = 2;

this->acceptbook\_button->Text = L"Принять книгу";

this->acceptbook\_button->UseVisualStyleBackColor = true;

this->acceptbook\_button->Click += gcnew System::EventHandler(this, &MenuForm::acceptbook\_button\_Click);

//

// givebook\_button

//

this->givebook\_button->Location = System::Drawing::Point(16, 31);

this->givebook\_button->Name = L"givebook\_button";

this->givebook\_button->Size = System::Drawing::Size(161, 34);

this->givebook\_button->TabIndex = 3;

this->givebook\_button->Text = L"Выдать книгу";

this->givebook\_button->UseVisualStyleBackColor = true;

this->givebook\_button->Click += gcnew System::EventHandler(this, &MenuForm::givebook\_button\_Click);

//

// groupBox1

//

this->groupBox1->Controls->Add(this->givebook\_button);

this->groupBox1->Controls->Add(this->acceptbook\_button);

this->groupBox1->Location = System::Drawing::Point(12, 42);

this->groupBox1->Name = L"groupBox1";

this->groupBox1->Size = System::Drawing::Size(196, 148);

this->groupBox1->TabIndex = 4;

this->groupBox1->TabStop = false;

//

// timer1

//

this->timer1->Enabled = true;

this->timer1->Tick += gcnew System::EventHandler(this, &MenuForm::timer1\_Tick);

//

// monthCalendar1

//

this->monthCalendar1->Location = System::Drawing::Point(22, 202);

this->monthCalendar1->Name = L"monthCalendar1";

this->monthCalendar1->TabIndex = 5;

//

// MenuForm

//

this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(220, 383);

this->Controls->Add(this->monthCalendar1);

this->Controls->Add(this->groupBox1);

this->Controls->Add(this->menuStrip1);

this->Icon = (cli::safe\_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));

this->MainMenuStrip = this->menuStrip1;

this->Name = L"MenuForm";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"АРМ";

this->FormClosed += gcnew System::Windows::Forms::FormClosedEventHandler(this, &MenuForm::MenuForm\_FormClosed);

this->Load += gcnew System::EventHandler(this, &MenuForm::MenuForm\_Load);

this->menuStrip1->ResumeLayout(false);

this->menuStrip1->PerformLayout();

this->groupBox1->ResumeLayout(false);

this->ResumeLayout(false);

this->PerformLayout();

}

#pragma endregion

private: System::Void складToolStripMenuItem\_Click(System::Object^ sender, System::EventArgs^ e) {

f1 = gcnew Form1(); // создаем экземпляр формы Form1

f1->Show(); //Открываем форму Form1

}

private: System::Void MenuForm\_Load(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void читателиToolStripMenuItem\_Click(System::Object^ sender, System::EventArgs^ e) {

rf = gcnew ReaderForm();

rf->Show();

}

private: System::Void givebook\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

gf = gcnew GiveForm();

gf->Show();

}

private: System::Void timer1\_Tick(System::Object^ sender, System::EventArgs^ e) {

if (rf->Visible || f1->Visible) {

menuStrip1->Enabled = false;

groupBox1->Enabled = false;

monthCalendar1->Enabled = false;

}

else {

menuStrip1->Enabled = true;

groupBox1->Enabled = true;

monthCalendar1->Enabled = true;

}

}

private: System::Void acceptbook\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

af = gcnew AcceptForm();

af->Show();

}

private: System::Void MenuForm\_FormClosed(System::Object^ sender, System::Windows::Forms::FormClosedEventArgs^ e) {

Application::Exit();

}

};

}

**GiveForm.h**

#pragma once

#include "Book.h"

#include "dateTimePicker.h"

#include <fstream>

#include <iostream>

#include <vector>

#include <msclr\marshal\_cppstd.h>

#include <filesystem>

namespace liba {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для GiveForm

/// </summary>

public ref class GiveForm : public System::Windows::Forms::Form

{

private: std::vector<std::string>\* books;

dateTimePicker^ dtp = gcnew dateTimePicker();

std::vector<std::string>\* readers;

public:

GiveForm(void)

{

InitializeComponent();

books = new std::vector<std::string>;

readers = new std::vector<std::string>;

LoadListBook();

LoadListPerson();

give\_groupBox2->Hide();

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~GiveForm()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::Label^ label1;

protected:

private: System::Windows::Forms::ListBox^ give\_listBox;

private: System::Windows::Forms::Button^ choose\_button;

private: System::Windows::Forms::GroupBox^ give\_groupBox;

private: System::Windows::Forms::GroupBox^ give\_groupBox2;

private: System::Windows::Forms::ListBox^ person\_listBox;

private: System::Windows::Forms::Label^ label2;

private: System::Windows::Forms::Button^ choose\_button2;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container^ components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(GiveForm::typeid));

this->label1 = (gcnew System::Windows::Forms::Label());

this->give\_listBox = (gcnew System::Windows::Forms::ListBox());

this->choose\_button = (gcnew System::Windows::Forms::Button());

this->give\_groupBox = (gcnew System::Windows::Forms::GroupBox());

this->give\_groupBox2 = (gcnew System::Windows::Forms::GroupBox());

this->person\_listBox = (gcnew System::Windows::Forms::ListBox());

this->label2 = (gcnew System::Windows::Forms::Label());

this->choose\_button2 = (gcnew System::Windows::Forms::Button());

this->give\_groupBox->SuspendLayout();

this->give\_groupBox2->SuspendLayout();

this->SuspendLayout();

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9.75F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->Location = System::Drawing::Point(114, 24);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(215, 16);

this->label1->TabIndex = 0;

this->label1->Text = L"Выберите книгу для выдачи";

//

// give\_listBox

//

this->give\_listBox->FormattingEnabled = true;

this->give\_listBox->Location = System::Drawing::Point(13, 43);

this->give\_listBox->Name = L"give\_listBox";

this->give\_listBox->Size = System::Drawing::Size(429, 225);

this->give\_listBox->TabIndex = 1;

//

// choose\_button

//

this->choose\_button->Location = System::Drawing::Point(13, 297);

this->choose\_button->Name = L"choose\_button";

this->choose\_button->Size = System::Drawing::Size(429, 23);

this->choose\_button->TabIndex = 2;

this->choose\_button->Text = L"Выбрать";

this->choose\_button->UseVisualStyleBackColor = true;

this->choose\_button->Click += gcnew System::EventHandler(this, &GiveForm::choose\_button\_Click);

//

// give\_groupBox

//

this->give\_groupBox->Controls->Add(this->give\_listBox);

this->give\_groupBox->Controls->Add(this->label1);

this->give\_groupBox->Controls->Add(this->choose\_button);

this->give\_groupBox->Location = System::Drawing::Point(12, 12);

this->give\_groupBox->Name = L"give\_groupBox";

this->give\_groupBox->Size = System::Drawing::Size(457, 326);

this->give\_groupBox->TabIndex = 4;

this->give\_groupBox->TabStop = false;

this->give\_groupBox->Text = L"Выдача книги";

//

// give\_groupBox2

//

this->give\_groupBox2->Controls->Add(this->person\_listBox);

this->give\_groupBox2->Controls->Add(this->label2);

this->give\_groupBox2->Controls->Add(this->choose\_button2);

this->give\_groupBox2->Location = System::Drawing::Point(12, 12);

this->give\_groupBox2->Name = L"give\_groupBox2";

this->give\_groupBox2->Size = System::Drawing::Size(457, 326);

this->give\_groupBox2->TabIndex = 5;

this->give\_groupBox2->TabStop = false;

this->give\_groupBox2->Text = L"Выдача книги";

//

// person\_listBox

//

this->person\_listBox->BackColor = System::Drawing::SystemColors::Window;

this->person\_listBox->FormattingEnabled = true;

this->person\_listBox->Location = System::Drawing::Point(13, 43);

this->person\_listBox->Name = L"person\_listBox";

this->person\_listBox->Size = System::Drawing::Size(429, 225);

this->person\_listBox->TabIndex = 1;

//

// label2

//

this->label2->AutoSize = true;

this->label2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9.75F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label2->Location = System::Drawing::Point(147, 24);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(153, 16);

this->label2->TabIndex = 0;

this->label2->Text = L"Выберите читателя";

//

// choose\_button2

//

this->choose\_button2->Location = System::Drawing::Point(13, 291);

this->choose\_button2->Name = L"choose\_button2";

this->choose\_button2->Size = System::Drawing::Size(429, 23);

this->choose\_button2->TabIndex = 2;

this->choose\_button2->Text = L"Выбрать";

this->choose\_button2->UseVisualStyleBackColor = true;

this->choose\_button2->Click += gcnew System::EventHandler(this, &GiveForm::choose\_button2\_Click);

//

// GiveForm

//

this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(487, 357);

this->Controls->Add(this->give\_groupBox2);

this->Controls->Add(this->give\_groupBox);

this->Icon = (cli::safe\_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));

this->Name = L"GiveForm";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"АРМ";

this->give\_groupBox->ResumeLayout(false);

this->give\_groupBox->PerformLayout();

this->give\_groupBox2->ResumeLayout(false);

this->give\_groupBox2->PerformLayout();

this->ResumeLayout(false);

}

#pragma endregion

private: void LoadListBook()

{

if (std::filesystem::exists("database\\библиотека.txt"))

{

std::ifstream file("database\\библиотека.txt");

std::string str = "";

while (getline(file, str)) {

(\*books).push\_back(str);

give\_listBox->Items->Add(msclr::interop::marshal\_as<String^>(str));

}

file.close();

}

}

private: System::Void choose\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

int index = give\_listBox->SelectedIndex;

std::ofstream file("database\\buffer.txt");

std::string str = (\*books)[index];

file << str;

file.close();

if (index >= 0 && index < (\*books).size())

{

(\*books).erase((\*books).begin() + index);

UpdateListBook();

}

std::ofstream log("database\\библиотека.txt");

for (int i = 0; i < give\_listBox->Items->Count; i++)

log << msclr::interop::marshal\_as<std::string>(give\_listBox->Items[i]->ToString()) << std::endl;

log.close();

give\_groupBox->Hide();

give\_groupBox2->Show();

}

private: System::Void UpdateListBook() {

give\_listBox->Items->Clear();

for (int i = 0; i < (\*books).size(); i++)

give\_listBox->Items->Add(msclr::interop::marshal\_as<String^>((\*books)[i]));

}

private: System::Void UpdateListPerson() {

person\_listBox->Items->Clear();

for (int i = 0; i < (\*readers).size(); i++)

person\_listBox->Items->Add(msclr::interop::marshal\_as<String^>((\*readers)[i]));

}

private: void LoadListPerson()

{

if (std::filesystem::exists("database\\Читатели.txt"))

{

std::ifstream file("database\\Читатели.txt");

std::string str = "";

while (getline(file, str)) {

(\*readers).push\_back(str);

person\_listBox->Items->Add(msclr::interop::marshal\_as<String^>(str));

}

file.close();

}

}

private: System::Void choose\_button2\_Click(System::Object^ sender, System::EventArgs^ e) {

int index = person\_listBox->SelectedIndex;

std::string str = (\*readers)[index];

std::string filename = "";

std::wstring cyrillicDigits = L"0123456789";

for (wchar\_t c : str) {

if (cyrillicDigits.find(c) != std::wstring::npos) {

filename += c;

}

}

std::string path = "users\\" + filename + ".txt";

std::ifstream fileBuff("database\\buffer.txt");

std::ofstream filePers;

filePers.open(path, std::ios::app);

std::string str1;

getline(fileBuff, str1);

filePers << str1 + ";";

fileBuff.close();

filePers.close();

dtp->Show();

dtp->Pathtxt(path);

this->Hide();

}

};

}

**Form1.h**

**#pragma once**

**#include "Book.h"**

**#include <fstream>**

**#include <iostream>**

**#include <vector>**

**#include <msclr\marshal\_cppstd.h>**

**#include <filesystem>**

**namespace liba**

**{**

**using namespace System;**

**using namespace System::ComponentModel;**

**using namespace System::Collections;**

**using namespace System::Windows::Forms;**

**using namespace System::Data;**

**using namespace System::Drawing;**

**/// <summary>**

**/// Сводка для Form1**

**/// </summary>**

**public ref class Form1 : public System::Windows::Forms::Form**

**{**

**private: std::vector<std::string>\* books;**

**public:**

**Form1(void)**

**{**

**InitializeComponent();**

**books = new std::vector<std::string>;**

**LoadList();**

**}**

**protected:**

**/// <summary>**

**/// Освободить все используемые ресурсы.**

**/// </summary>**

**~Form1()**

**{**

**if (components)**

**{**

**delete components;**

**}**

**}**

**private: System::Windows::Forms::TextBox^ titleTextBox;**

**private: System::Windows::Forms::TextBox^ authorTextBox;**

**private: System::Windows::Forms::TextBox^ publisherTextBox;**

**private: System::Windows::Forms::TextBox^ yearTextBox;**

**private: System::Windows::Forms::Label^ label1;**

**private: System::Windows::Forms::Label^ label2;**

**private: System::Windows::Forms::Label^ label3;**

**private: System::Windows::Forms::Label^ label4;**

**private: System::Windows::Forms::Button^ addButton;**

**private: System::Windows::Forms::Button^ removeButton;**

**private: System::Windows::Forms::ListBox^ booksListBox;**

**protected:**

**protected:**

**private:**

**/// <summary>**

**/// Обязательная переменная конструктора.**

**/// </summary>**

**System::ComponentModel::Container^ components;**

**private: System::Windows::Forms::Button^ save\_button;**

**private: System::Windows::Forms::TextBox^ searchTextBox;**

**private: System::Windows::Forms::Label^ label6;**

**private: System::Windows::Forms::GroupBox^ groupBox1;**

**private: System::Windows::Forms::Button^ button1;**

**private: System::Windows::Forms::Label^ label5;**

**#pragma region Windows Form Designer generated code**

**/// <summary>**

**/// Требуемый метод для поддержки конструктора — не изменяйте**

**/// содержимое этого метода с помощью редактора кода.**

**/// </summary>**

**void InitializeComponent(void)**

**{**

**System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(Form1::typeid));**

**this->titleTextBox = (gcnew System::Windows::Forms::TextBox());**

**this->authorTextBox = (gcnew System::Windows::Forms::TextBox());**

**this->publisherTextBox = (gcnew System::Windows::Forms::TextBox());**

**this->yearTextBox = (gcnew System::Windows::Forms::TextBox());**

**this->label1 = (gcnew System::Windows::Forms::Label());**

**this->label2 = (gcnew System::Windows::Forms::Label());**

**this->label3 = (gcnew System::Windows::Forms::Label());**

**this->label4 = (gcnew System::Windows::Forms::Label());**

**this->addButton = (gcnew System::Windows::Forms::Button());**

**this->removeButton = (gcnew System::Windows::Forms::Button());**

**this->booksListBox = (gcnew System::Windows::Forms::ListBox());**

**this->label5 = (gcnew System::Windows::Forms::Label());**

**this->save\_button = (gcnew System::Windows::Forms::Button());**

**this->searchTextBox = (gcnew System::Windows::Forms::TextBox());**

**this->label6 = (gcnew System::Windows::Forms::Label());**

**this->groupBox1 = (gcnew System::Windows::Forms::GroupBox());**

**this->button1 = (gcnew System::Windows::Forms::Button());**

**this->groupBox1->SuspendLayout();**

**this->SuspendLayout();**

**//**

**// titleTextBox**

**//**

**this->titleTextBox->Location = System::Drawing::Point(116, 31);**

**this->titleTextBox->Name = L"titleTextBox";**

**this->titleTextBox->Size = System::Drawing::Size(100, 20);**

**this->titleTextBox->TabIndex = 0;**

**//**

**// authorTextBox**

**//**

**this->authorTextBox->Location = System::Drawing::Point(116, 68);**

**this->authorTextBox->Name = L"authorTextBox";**

**this->authorTextBox->Size = System::Drawing::Size(100, 20);**

**this->authorTextBox->TabIndex = 1;**

**//**

**// publisherTextBox**

**//**

**this->publisherTextBox->Location = System::Drawing::Point(116, 110);**

**this->publisherTextBox->Name = L"publisherTextBox";**

**this->publisherTextBox->Size = System::Drawing::Size(100, 20);**

**this->publisherTextBox->TabIndex = 2;**

**//**

**// yearTextBox**

**//**

**this->yearTextBox->Location = System::Drawing::Point(116, 149);**

**this->yearTextBox->Name = L"yearTextBox";**

**this->yearTextBox->Size = System::Drawing::Size(100, 20);**

**this->yearTextBox->TabIndex = 3;**

**//**

**// label1**

**//**

**this->label1->AutoSize = true;**

**this->label1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,**

**static\_cast<System::Byte>(204)));**

**this->label1->ForeColor = System::Drawing::SystemColors::ControlText;**

**this->label1->Location = System::Drawing::Point(6, 32);**

**this->label1->Name = L"label1";**

**this->label1->Size = System::Drawing::Size(67, 15);**

**this->label1->TabIndex = 4;**

**this->label1->Text = L"Название:";**

**//**

**// label2**

**//**

**this->label2->AutoSize = true;**

**this->label2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,**

**static\_cast<System::Byte>(204)));**

**this->label2->ForeColor = System::Drawing::SystemColors::ControlText;**

**this->label2->Location = System::Drawing::Point(6, 73);**

**this->label2->Name = L"label2";**

**this->label2->Size = System::Drawing::Size(45, 15);**

**this->label2->TabIndex = 5;**

**this->label2->Text = L"Автор:";**

**//**

**// label3**

**//**

**this->label3->AutoSize = true;**

**this->label3->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,**

**static\_cast<System::Byte>(204)));**

**this->label3->ForeColor = System::Drawing::SystemColors::ControlText;**

**this->label3->Location = System::Drawing::Point(6, 111);**

**this->label3->Name = L"label3";**

**this->label3->Size = System::Drawing::Size(94, 15);**

**this->label3->TabIndex = 6;**

**this->label3->Text = L"Издательство:";**

**//**

**// label4**

**//**

**this->label4->AutoSize = true;**

**this->label4->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,**

**static\_cast<System::Byte>(204)));**

**this->label4->ForeColor = System::Drawing::SystemColors::ControlText;**

**this->label4->Location = System::Drawing::Point(6, 149);**

**this->label4->Name = L"label4";**

**this->label4->Size = System::Drawing::Size(81, 15);**

**this->label4->TabIndex = 7;**

**this->label4->Text = L"Год выпуска:";**

**//**

**// addButton**

**//**

**this->addButton->Location = System::Drawing::Point(149, 190);**

**this->addButton->Name = L"addButton";**

**this->addButton->Size = System::Drawing::Size(67, 29);**

**this->addButton->TabIndex = 8;**

**this->addButton->Text = L"Добавить";**

**this->addButton->UseVisualStyleBackColor = true;**

**this->addButton->Click += gcnew System::EventHandler(this, &Form1::addButton\_Click);**

**//**

**// removeButton**

**//**

**this->removeButton->Location = System::Drawing::Point(671, 341);**

**this->removeButton->Name = L"removeButton";**

**this->removeButton->Size = System::Drawing::Size(65, 29);**

**this->removeButton->TabIndex = 9;**

**this->removeButton->Text = L"Удалить";**

**this->removeButton->UseVisualStyleBackColor = true;**

**this->removeButton->Click += gcnew System::EventHandler(this, &Form1::removeButton\_Click);**

**//**

**// booksListBox**

**//**

**this->booksListBox->Cursor = System::Windows::Forms::Cursors::Default;**

**this->booksListBox->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 8.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,**

**static\_cast<System::Byte>(0)));**

**this->booksListBox->FormattingEnabled = true;**

**this->booksListBox->Location = System::Drawing::Point(250, 32);**

**this->booksListBox->Name = L"booksListBox";**

**this->booksListBox->Size = System::Drawing::Size(584, 303);**

**this->booksListBox->TabIndex = 10;**

**this->booksListBox->DoubleClick += gcnew System::EventHandler(this, &Form1::booksListBox\_DoubleClick);**

**//**

**// label5**

**//**

**this->label5->AutoSize = true;**

**this->label5->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,**

**static\_cast<System::Byte>(204)));**

**this->label5->ForeColor = System::Drawing::SystemColors::ControlText;**

**this->label5->Location = System::Drawing::Point(246, 9);**

**this->label5->Name = L"label5";**

**this->label5->Size = System::Drawing::Size(76, 15);**

**this->label5->TabIndex = 11;**

**this->label5->Text = L"Список книг";**

**//**

**// save\_button**

**//**

**this->save\_button->Location = System::Drawing::Point(742, 341);**

**this->save\_button->Name = L"save\_button";**

**this->save\_button->Size = System::Drawing::Size(80, 29);**

**this->save\_button->TabIndex = 12;**

**this->save\_button->Text = L"Сохранить";**

**this->save\_button->UseVisualStyleBackColor = true;**

**this->save\_button->Click += gcnew System::EventHandler(this, &Form1::save\_button\_Click);**

**//**

**// searchTextBox**

**//**

**this->searchTextBox->Location = System::Drawing::Point(6, 27);**

**this->searchTextBox->Name = L"searchTextBox";**

**this->searchTextBox->Size = System::Drawing::Size(237, 20);**

**this->searchTextBox->TabIndex = 13;**

**this->searchTextBox->TextChanged += gcnew System::EventHandler(this, &Form1::searchTextBox\_TextChanged);**

**//**

**// label6**

**//**

**this->label6->AutoSize = true;**

**this->label6->Location = System::Drawing::Point(12, 11);**

**this->label6->Name = L"label6";**

**this->label6->Size = System::Drawing::Size(39, 13);**

**this->label6->TabIndex = 14;**

**this->label6->Text = L"Поиск";**

**//**

**// groupBox1**

**//**

**this->groupBox1->Controls->Add(this->yearTextBox);**

**this->groupBox1->Controls->Add(this->titleTextBox);**

**this->groupBox1->Controls->Add(this->authorTextBox);**

**this->groupBox1->Controls->Add(this->publisherTextBox);**

**this->groupBox1->Controls->Add(this->label1);**

**this->groupBox1->Controls->Add(this->label2);**

**this->groupBox1->Controls->Add(this->addButton);**

**this->groupBox1->Controls->Add(this->label3);**

**this->groupBox1->Controls->Add(this->label4);**

**this->groupBox1->Location = System::Drawing::Point(6, 97);**

**this->groupBox1->Name = L"groupBox1";**

**this->groupBox1->Size = System::Drawing::Size(237, 271);**

**this->groupBox1->TabIndex = 15;**

**this->groupBox1->TabStop = false;**

**this->groupBox1->Text = L"Добавление";**

**//**

**// button1**

**//**

**this->button1->Location = System::Drawing::Point(173, 53);**

**this->button1->Name = L"button1";**

**this->button1->Size = System::Drawing::Size(70, 23);**

**this->button1->TabIndex = 16;**

**this->button1->Text = L"Поиск";**

**this->button1->UseVisualStyleBackColor = true;**

**//**

**// Form1**

**//**

**this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);**

**this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;**

**this->BackColor = System::Drawing::Color::White;**

**this->ClientSize = System::Drawing::Size(846, 380);**

**this->Controls->Add(this->button1);**

**this->Controls->Add(this->groupBox1);**

**this->Controls->Add(this->label6);**

**this->Controls->Add(this->searchTextBox);**

**this->Controls->Add(this->save\_button);**

**this->Controls->Add(this->label5);**

**this->Controls->Add(this->booksListBox);**

**this->Controls->Add(this->removeButton);**

**this->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 8.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,**

**static\_cast<System::Byte>(204)));**

**this->Icon = (cli::safe\_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));**

**this->Name = L"Form1";**

**this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;**

**this->Text = L"АРМ";**

**this->groupBox1->ResumeLayout(false);**

**this->groupBox1->PerformLayout();**

**this->ResumeLayout(false);**

**this->PerformLayout();**

**}**

**#pragma endregion**

**private: System::Void addButton\_Click(System::Object^ sender, System::EventArgs^ e) {**

**std::string title = msclr::interop::marshal\_as<std::string>(titleTextBox->Text);**

**std::string author = msclr::interop::marshal\_as<std::string>(authorTextBox->Text);**

**std::string publisher = msclr::interop::marshal\_as<std::string>(publisherTextBox->Text);**

**int year = Int32::Parse(yearTextBox->Text);**

**Book book(title, author, publisher, year);**

**std::string str = book.toString();**

**(\*books).push\_back(str);**

**UpdateList();**

**}**

**private: System::Void removeButton\_Click(System::Object^ sender, System::EventArgs^ e) {**

**int index = booksListBox->SelectedIndex;**

**if (index >= 0 && index < (\*books).size())**

**{**

**(\*books).erase((\*books).begin() + index);**

**UpdateList();**

**}**

**}**

**private: System::Void UpdateList() {**

**booksListBox->Items->Clear();**

**for (int i = 0; i < (\*books).size(); i++)**

**booksListBox->Items->Add(msclr::interop::marshal\_as<String^>((\*books)[i]));**

**}**

**private: void LoadList()**

**{**

**if (std::filesystem::exists("database\\библиотека.txt"))**

**{**

**std::ifstream file("database\\библиотека.txt");**

**std::string str = "";**

**while (getline(file, str)) {**

**(\*books).push\_back(str);**

**booksListBox->Items->Add(msclr::interop::marshal\_as<String^>(str));**

**}**

**file.close();**

**}**

**}**

**private: System::Void save\_button\_Click(System::Object^ sender, System::EventArgs^ e) {**

**std::ofstream log("database\\библиотека.txt");**

**for (int i = 0; i < booksListBox->Items->Count; i++)**

**log << msclr::interop::marshal\_as<std::string>(booksListBox->Items[i]->ToString()) << std::endl;**

**log.close();**

**}**

**private: System::Void searchTextBox\_TextChanged(System::Object^ sender, System::EventArgs^ e) {**

**// Получение текущего текста из TextBox**

**String^ searchText = searchTextBox->Text;**

**// Очистка ListBox от предыдущих результатов поиска**

**booksListBox->Items->Clear();**

**// Проход по элементам ListBox и проверка наличия введенного текста поиска**

**for (int i = 0; i < booksListBox->Items->Count; i++) {**

**String^ itemText = booksListBox->Items[i]->ToString();**

**// Если элемент содержит введенный текст, добавляем его в ListBox**

**if (itemText->Contains(searchText)) {**

**booksListBox->Items->Add(itemText);**

**}**

**}**

**}**

**private: System::Void booksListBox\_DoubleClick(System::Object^ sender, System::EventArgs^ e) {**

**std::string randomSector;**

**std::string randomPolka;**

**///СЕКТОР**

**std::ifstream file("place\\sector.txt");**

**if (file.is\_open()) {**

**std::vector<std::string> lines;**

**std::string line;**

**while (getline(file, line))**

**{**

**lines.push\_back(line);**

**}**

**srand(static\_cast<unsigned int>(time(0)));**

**int randomIndex = rand() % lines.size();**

**randomSector = lines[randomIndex];**

**file.close();**

**}**

**else**

**{**

**// файл не удалось открыть**

**std::cout << "Не удалось открыть файл." << std::endl;**

**}**

**///ПОЛКА**

**std::ifstream file2("place\\polka.txt");**

**if (file2.is\_open()) {**

**std::vector<std::string> lines2;**

**std::string line2;**

**while (getline(file2, line2))**

**{**

**lines2.push\_back(line2);**

**}**

**int randomIndex2 = rand() % lines2.size();**

**randomPolka = lines2[randomIndex2];**

**file2.close();**

**}**

**else {**

**// файл не удалось открыть**

**std::cout << "Не удалось открыть файл." << std::endl;**

**}**

**System::String^ randomLineString1 = msclr::interop::marshal\_as<System::String^>(randomSector);**

**System::String^ randomLineString2 = msclr::interop::marshal\_as<System::String^>(randomPolka);**

**MessageBox::Show("Сектор: " + randomLineString1 + "\n" + "\n" + "Полка: " + randomLineString2, "Местоположение");**

**}**

**};**

**}**

**dateTimePicker.h**

#pragma once

#include <Windows.h>

#include <msclr\marshal\_cppstd.h>

#include <fstream>

namespace liba {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для dateTimePicker

/// </summary>

///

static std::string Path;

public ref class dateTimePicker : public System::Windows::Forms::Form

{

public:

dateTimePicker(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~dateTimePicker()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::DateTimePicker^ dateTimePicker1;

private: System::Windows::Forms::Button^ vybrat\_button;

private: System::Windows::Forms::Label^ label1;

protected:

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(dateTimePicker::typeid));

this->dateTimePicker1 = (gcnew System::Windows::Forms::DateTimePicker());

this->vybrat\_button = (gcnew System::Windows::Forms::Button());

this->label1 = (gcnew System::Windows::Forms::Label());

this->SuspendLayout();

//

// dateTimePicker1

//

this->dateTimePicker1->Location = System::Drawing::Point(12, 28);

this->dateTimePicker1->Name = L"dateTimePicker1";

this->dateTimePicker1->Size = System::Drawing::Size(230, 20);

this->dateTimePicker1->TabIndex = 6;

//

// vybrat\_button

//

this->vybrat\_button->Location = System::Drawing::Point(25, 85);

this->vybrat\_button->Name = L"vybrat\_button";

this->vybrat\_button->Size = System::Drawing::Size(200, 25);

this->vybrat\_button->TabIndex = 7;

this->vybrat\_button->Text = L"Выбрать";

this->vybrat\_button->UseVisualStyleBackColor = true;

this->vybrat\_button->Click += gcnew System::EventHandler(this, &dateTimePicker::vybrat\_button\_Click);

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->Location = System::Drawing::Point(67, 9);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(108, 15);

this->label1->TabIndex = 8;

this->label1->Text = L"Выберите дату";

//

// dateTimePicker

//

this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(254, 122);

this->Controls->Add(this->label1);

this->Controls->Add(this->vybrat\_button);

this->Controls->Add(this->dateTimePicker1);

this->Icon = (cli::safe\_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));

this->Name = L"dateTimePicker";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"АРМ";

this->ResumeLayout(false);

this->PerformLayout();

}

#pragma endregion

public: void Pathtxt(std::string put) {

Path = put;

}

private: System::Void vybrat\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

DateTime selectedDate = dateTimePicker1->Value;

String^ dateString = selectedDate.ToString();

DateTime currentTime = DateTime::Now;

TimeSpan difference = selectedDate - currentTime;

MessageBox::Show("Книга выдана до " + dateString + "\n" + "Сдать через " + difference.Days + " дн." + difference.Hours + "ч." + difference.Minutes + "мин.", "Срок сдачи");

String^ selectedString = selectedDate.ToString(); // Преобразование разницы в строку

std::string differenceStdString = msclr::interop::marshal\_as<std::string>(selectedString); // Преобразование String^ в std::string

std::ofstream filedate;

filedate.open(Path, std::ios::app);

std::string str = differenceStdString + ";";

filedate << str;

filedate.close();

this->Close();

}

};

}

**Book.h**

#pragma once

#include <iostream>

#include <string>

public class Book

{

private:

std::string title;

std::string author;

std::string publisher;

int year;

public:

Book()

{

this->title = "";

this->publisher = "";

this->author = "";

this->year = 0;

}

Book(std::string title, std::string author, std::string publisher, int year)

{

this->title = title;

this->publisher = publisher;

this->author = author;

this->year = year;

}

std::string get\_title() { return this->title; }

std::string get\_author() { return this->author; }

std::string get\_publisher() { return this->publisher; }

int get\_year() { return this->year; }

std::string toString()

{

return this->title + " by " + this->author + " (" + this->publisher + ", " + std::to\_string(this->year) + ")";

}

};

**AcceptForm.h**

#pragma once

#include <fstream>

#include <vector>

#include <msclr\marshal\_cppstd.h>

#include <filesystem>

namespace liba {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

using namespace System::Collections::Generic;

using namespace msclr::interop;

/// <summary>

/// Сводка для AcceptForm

/// </summary>

static std::string Path2;

public ref class AcceptForm : public System::Windows::Forms::Form

{

private: std::vector<std::string>\* readers;

std::vector<std::string>\* booki;

private: System::Windows::Forms::Label^ label1;

private: System::Windows::Forms::Label^ label2;

List<String^>^ parts;

//System::Collections::Generic::List<String^>^ parts;

//array<String^>^ parts;

public:

AcceptForm(void)

{

InitializeComponent();

readers = new std::vector<std::string>;

booki = new std::vector<std::string>;

//parts = gcnew System::Collections::Generic::List<String^>();

LoadListPerson();

groupBox1->Enabled = false;

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~AcceptForm()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::ListBox^ reader\_listBox7;

protected:

private: System::Windows::Forms::GroupBox^ accepr\_groupBox;

private: System::Windows::Forms::Button^ choosePers\_button;

private: System::Windows::Forms::ListBox^ book\_list;

private: System::Windows::Forms::GroupBox^ groupBox1;

private: System::Windows::Forms::Button^ accept\_button;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(AcceptForm::typeid));

this->reader\_listBox7 = (gcnew System::Windows::Forms::ListBox());

this->accepr\_groupBox = (gcnew System::Windows::Forms::GroupBox());

this->choosePers\_button = (gcnew System::Windows::Forms::Button());

this->book\_list = (gcnew System::Windows::Forms::ListBox());

this->groupBox1 = (gcnew System::Windows::Forms::GroupBox());

this->accept\_button = (gcnew System::Windows::Forms::Button());

this->label1 = (gcnew System::Windows::Forms::Label());

this->label2 = (gcnew System::Windows::Forms::Label());

this->accepr\_groupBox->SuspendLayout();

this->groupBox1->SuspendLayout();

this->SuspendLayout();

//

// reader\_listBox7

//

this->reader\_listBox7->FormattingEnabled = true;

this->reader\_listBox7->Location = System::Drawing::Point(15, 37);

this->reader\_listBox7->Name = L"reader\_listBox7";

this->reader\_listBox7->Size = System::Drawing::Size(209, 238);

this->reader\_listBox7->TabIndex = 0;

//

// accepr\_groupBox

//

this->accepr\_groupBox->Controls->Add(this->label1);

this->accepr\_groupBox->Controls->Add(this->choosePers\_button);

this->accepr\_groupBox->Controls->Add(this->reader\_listBox7);

this->accepr\_groupBox->Location = System::Drawing::Point(12, 12);

this->accepr\_groupBox->Name = L"accepr\_groupBox";

this->accepr\_groupBox->Size = System::Drawing::Size(244, 318);

this->accepr\_groupBox->TabIndex = 1;

this->accepr\_groupBox->TabStop = false;

this->accepr\_groupBox->Text = L"Читатели";

//

// choosePers\_button

//

this->choosePers\_button->Location = System::Drawing::Point(15, 281);

this->choosePers\_button->Name = L"choosePers\_button";

this->choosePers\_button->Size = System::Drawing::Size(209, 26);

this->choosePers\_button->TabIndex = 1;

this->choosePers\_button->Text = L"Выбрать";

this->choosePers\_button->UseVisualStyleBackColor = true;

this->choosePers\_button->Click += gcnew System::EventHandler(this, &AcceptForm::choosePers\_button\_Click);

//

// book\_list

//

this->book\_list->FormattingEnabled = true;

this->book\_list->Location = System::Drawing::Point(6, 37);

this->book\_list->Name = L"book\_list";

this->book\_list->Size = System::Drawing::Size(558, 238);

this->book\_list->TabIndex = 2;

//

// groupBox1

//

this->groupBox1->Controls->Add(this->label2);

this->groupBox1->Controls->Add(this->accept\_button);

this->groupBox1->Controls->Add(this->book\_list);

this->groupBox1->Location = System::Drawing::Point(262, 12);

this->groupBox1->Name = L"groupBox1";

this->groupBox1->Size = System::Drawing::Size(573, 318);

this->groupBox1->TabIndex = 3;

this->groupBox1->TabStop = false;

this->groupBox1->Text = L"Книги";

//

// accept\_button

//

this->accept\_button->Location = System::Drawing::Point(6, 281);

this->accept\_button->Name = L"accept\_button";

this->accept\_button->Size = System::Drawing::Size(558, 26);

this->accept\_button->TabIndex = 3;

this->accept\_button->Text = L"Принять";

this->accept\_button->UseVisualStyleBackColor = true;

this->accept\_button->Click += gcnew System::EventHandler(this, &AcceptForm::accept\_button\_Click);

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9, System::Drawing::FontStyle::Italic, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->Location = System::Drawing::Point(59, 16);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(124, 15);

this->label1->TabIndex = 2;

this->label1->Text = L"Выберите читателя";

//

// label2

//

this->label2->AutoSize = true;

this->label2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9, System::Drawing::FontStyle::Italic, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label2->Location = System::Drawing::Point(217, 19);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(99, 15);

this->label2->TabIndex = 3;

this->label2->Text = L"Выберите книгу";

//

// AcceptForm

//

this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(844, 337);

this->Controls->Add(this->groupBox1);

this->Controls->Add(this->accepr\_groupBox);

this->Icon = (cli::safe\_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));

this->Name = L"AcceptForm";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"АРМ";

this->accepr\_groupBox->ResumeLayout(false);

this->accepr\_groupBox->PerformLayout();

this->groupBox1->ResumeLayout(false);

this->groupBox1->PerformLayout();

this->ResumeLayout(false);

}

#pragma endregion

private: void LoadListPerson()

{

if (std::filesystem::exists("database\\Читатели.txt"))

{

std::ifstream file("database\\Читатели.txt");

std::string str = "";

while (getline(file, str)) {

(\*readers).push\_back(str);

reader\_listBox7->Items->Add(msclr::interop::marshal\_as<String^>(str));

}

file.close();

}

}

public: void Pathtxt(std::string path) {

Path2 = path;

}

private: System::Void choosePers\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

int index = reader\_listBox7->SelectedIndex;

std::string str = (\*readers)[index];

std::string filename = "";

std::wstring cyrillicDigits = L"0123456789";

for (wchar\_t c : str) {

if (cyrillicDigits.find(c) != std::wstring::npos) {

filename += c;

}

}

std::string path = "users\\" + filename + ".txt";

Pathtxt(path);

std::ifstream filePers;

filePers.open(path, std::ios::app);

std::string books\_;

getline(filePers, books\_);

String^ books = msclr::interop::marshal\_as<String^>(books\_);

parts = gcnew List<String^>(books->Split(';'));

for (int i = 0; i < parts->Count-1; i += 2) {

book\_list->Items->Add(parts[i]);

}

if (book\_list->Items->Count == 0) {

MessageBox::Show("Книг не зарегестрировано");

}

else {

accepr\_groupBox->Enabled = false;

groupBox1->Enabled = true;

}

}

private: System::Void accept\_button\_Click(System::Object^ sender, System::EventArgs^ e) {

String^ selectedItem = book\_list->SelectedItem->ToString();

book\_list->Items->RemoveAt(book\_list->SelectedIndex);

std::ofstream fileReader("database\\библиотека.txt", std::ios::app);

std::string strSelItem = marshal\_as<std::string>(selectedItem);

fileReader << strSelItem;

fileReader.close();

for (int i = 0; i < parts->Count; i++) {

if (parts[i] == selectedItem){

parts->RemoveAt(i + 1);

parts->RemoveAt(i);

}

}

std::ofstream file(Path2);

std::string strPart;

for (int i = 0; i < parts->Count; i++) {

strPart = marshal\_as<std::string>(parts[i]);

file << strPart + ";";

}

file.close();

if (book\_list->Items->Count == 0) {

accepr\_groupBox->Enabled = true;

groupBox1->Enabled = false;

MessageBox::Show("Книг больше не зарегестрировано");

}

else {

accepr\_groupBox->Enabled = false;

groupBox1->Enabled = true;

}

}

};

}

**Вывод программы**



***Рисунок 1 – вывод программы***



***Рисунок 2 – вывод программы***



***Рисунок 3 – вывод программы***



***Рисунок 4 – вывод программы***



***Рисунок 5 – вывод программы***



***Рисунок 6 – вывод программы***



***Рисунок 7 – вывод программы***



**UML диаграмма**



***Рисунок 2 – UML диаграмма***

**Вывод**

Программа выполняет свою задачу.

**Заключение**

Для решения задачи потребовались знания языка программирования, а конкретнее C++. Программа выполняет те условия, что были указаны в постановке задачи и работает без проблем.