

Министерство образования, науки и молодежной политики
Краснодарского края
Государственное бюджетное профессиональное образовательное учреждение
Краснодарского края «Ейский полипрофильный колледж»

Отделение педагогики
Специальность 09.02.07 Информационные системы и программирование

Студента И-31 группы
Сафронова Данила Дмитриевича

РАЗРАБОТКА МОДУЛЕЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ
ВОЕННОГО КОМИССАРИАТА

Научный руководитель
преподаватель специальных
дисциплин
Фомин А. Т.

г. Ейск
2024 г.

ОГЛАВЛЕНИЕ

| | |
|---|----|
| ВВЕДЕНИЕ..... | 3 |
| ГЛАВА 1 ПРОЕКТИРОВАНИЕ ПРИЛОЖЕНИЯ..... | 5 |
| 1.1 Описание предметной области..... | 5 |
| 1.2 Описание назначения программного средства (ПС)..... | 5 |
| 1.3 Анализ требований к ПС..... | 5 |
| ГЛАВА 2 РАЗРАБОТКА ПРИЛОЖЕНИЯ..... | 8 |
| 2.1 Описание модели базы данных SQLite..... | 8 |
| 2.2 Диаграммы классов..... | 9 |
| 2.3 Интерфейс программы..... | 10 |
| ЗАКЛЮЧЕНИЕ..... | 11 |
| СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ..... | 12 |
| ПРИЛОЖЕНИЕ А..... | 13 |
| ПРИЛОЖЕНИЕ Б..... | 23 |

ВВЕДЕНИЕ

Военный комиссариат создается в субъекте Российской Федерации в целях обеспечения исполнения гражданами воинской обязанности, организации и проведения мобилизационной подготовки и мобилизации, реализации права граждан, уволенных с военной службы, и членов их семей, членов семей погибших (умерших) военнослужащих на социальные гарантии, включая пенсионное обеспечение, а также в целях реализации гарантий погребения погибших (умерших) военнослужащих и иных категорий граждан в соответствии с законодательством Российской Федерации.

Так как количество мужчин, с которыми взаимодействует военкомат, велико, необходимо разработать эффективную и удобную систему их учета.

Темой моего проекта является разработка модулей информационной системы для военного комиссариата

Актуальность

Через военный комиссариат проходит большой поток людей, от призывников до людей, заключивших контракт с вооруженными силами России, поэтому данному учреждению необходима удобная система управления людьми, связанными с деятельностью военкомата, соответственно данная тема достаточно актуальна на текущий момент.

Объект – предметная область военный комиссариат.

Предмет – графическое приложение для работы с базой данных.

Цель работы – разработка модулей информационной системы базы данных предметной области «военный комиссариат».

Задачи:

Для достижения поставленной цели курсового проекта необходимо решить следующие задачи:

1. Описание предметной области;

2. Описание назначение ПС;
3. Анализ требований к ПС;
4. Выбор технологии и среды программирования;
5. Создание информационной модели данных;
6. Разработка программных модулей;
7. Интеграция программных модулей;
8. Описание интерфейса приложения;

Структура работы: курсовой проект состоит из введения, двух глав, заключения, списка литературы. В первой главе описаны данные о проектировании и представлении приложения. Вторая глава посвящена описанию процесса разработки приложения по заданной теме.

ГЛАВА 1 ПРОЕКТИРОВАНИЕ ПРИЛОЖЕНИЯ

1.1 Описание предметной области

Военный комиссариат - орган местного военного управления, ответственный за военно-мобилизационную и учётно-призывную работу в Вооружённых Силах Российской Федерации. Осуществляет организацию и воинский учет граждан, оказание содействия федеральным органам государственной власти, органам исполнительной власти субъектов Российской Федерации, органам местного самоуправления и организациям в осуществлении воинского учета.

1.2 Описание назначения программного средства (ПС)

Приложение с графическим пользовательским интерфейсом, позволяющее взаимодействовать с базой данных, предназначенной для хранения и систематизации данных военного комиссариата.

1.3 Анализ требований к ПС

Требования для приложения:

Минимальная избыточность, непротиворечивость и целостность данных, независимость, возможность ведения (добавления и удаления) и актуализации (редактирования), высокая производительность.

Условия эксплуатации:

Использовать данную программу должен только квалифицированный специалист с высшим техническим образованием по любой специальности, с высоким уровнем владения ПК и базовыми знаниями в области программирования

Требования к программному обеспечению:

◆ использование реляционной модели базы данных SQLite, которая обеспечивает хранение всей необходимой информации;

Основные требования к техническому обеспечению:

1. Двухъядерный процессор с частотой 1,6 ГГц или выше.
2. 4 ГБ оперативной памяти.
3. 4 ГБ свободного места на жестком диске.
4. Разрешение экрана 1280 x 768 или выше.
5. Операционная система Windows 10 или Windows 11.

1.4 Выбор технологии и среды программирования

При разработке было отдано предпочтение структурному подходу к программированию. При данном подходе программа разбивается на отдельные модули. У каждого модуля один вход и один выход - законченная конструкция для решения поставленной задачи. Цель создаваемой программы достигается составлением композиции из отдельных модулей, разработанных применительно к поставленной задаче. Средой разработки выбрано Microsoft Visual Studio Community 2022 при использовании шаблона Windows Forms C#.

Вывод по первой главе

В первой главе был произведен анализ для проектирования и разработки автоматизированной системы учета людей для военного комиссариата. Предметная область была изучена. Было описано назначение и анализ требований ПС, определены технология и среда разработки, которые обеспечат удобный и функциональный интерфейс.

ГЛАВА 2 РАЗРАБОТКА ПРИЛОЖЕНИЯ

2.1 Описание модели базы данных SQLite

SQLite — это быстрая и легкая встраиваемая однофайловая СУБД на языке C, которая не имеет сервера и позволяет хранить всю базу локально на одном устройстве. Для работы SQLite не нужны сторонние библиотеки или службы.

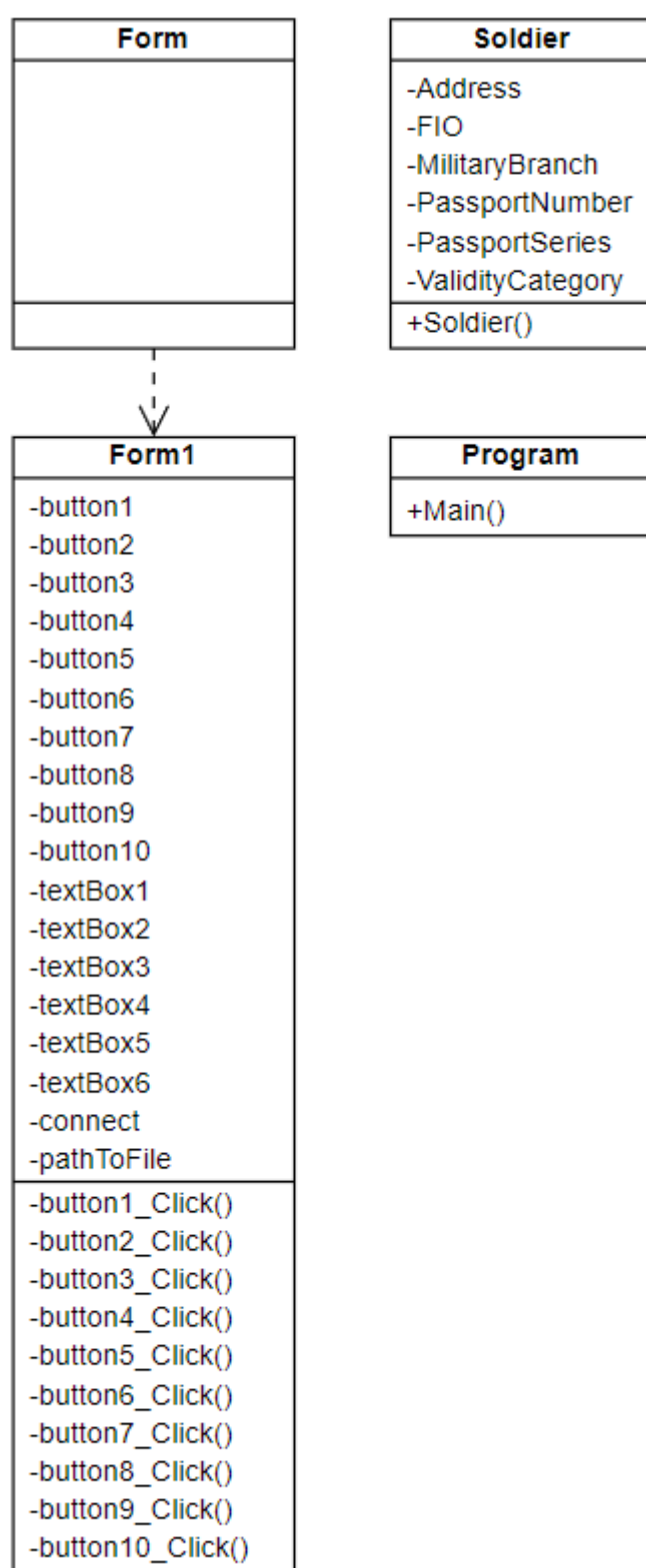
Понятие «встраиваемый» означает, что СУБД не использует парадигму клиент-сервер. Движок SQLite — не отдельно работающий процесс, с которым взаимодействует программа, а библиотека. Программа компонуется с ней, и движок служит составной частью программы. В качестве протокола обмена применяются вызовы функций (API) библиотеки SQLite.

Благодаря свойствам SQLite применяется:

- на сайтах с низким и средним трафиком;
- в локальных однопользовательских, мобильных приложениях или играх, не предназначенных для масштабирования;
- в программах, которые часто выполняют прямые операции чтения/записи на диск;
- в приложениях для тестирования бизнес-логики.

SQLite не требует администрирования и работает на мобильных устройствах, игровых приставках, телевизорах, беспилотных летательных аппаратах, камерах, автомобильных мультимедийных системах и т.д. СУБД использует множество программ: Firefox, Chrome, Safari, Skype, XnView, AIMP, Dropbox, Viber и другие.

2.2 Диаграммы классов



2.3 Интерфейс программы и инструкция пользователя

При запуске программы пользователь видит перед собой главную форму (рис. 1) информационной системы. Слева находятся поля для заполнения данных добавляемого человека, а справа — кнопки управления списком уже добавленных людей. При заполнении полей слева и нажатии кнопки «Добавить» добавляется несколько солдат (рис. 2) При выделении конкретного солдата и нажатии кнопки «Удалить выбранного», он удаляется из списка (рис. 3). Если нажать кнопку «Очистить список», весь список будет очищен. Кнопка «Загрузить в БД» произведет 1 запись в БД, где значениями столбцов будет выступать текущий текст заполненных слева полей. Кнопка «Записать в файл» запишет весь текст списка в файл.

ЗАКЛЮЧЕНИЕ

Выполнение курсового проекта было направлено на разработку графического приложения для работы с базой данных для предметной области военный комиссариат, что предполагало осуществление нескольких задач.

Для этого было необходимо рассмотреть особенности предметной области и её провести анализ, обеспечить создание необходимых компонентов (модулей) для работы.

Все этапы по реализации данных задач были выполнены, и поставленная цель была достигнута.

В ходе работы над курсовым проектом применялись знания и умения, полученные в ходе изучения учебных тем, полученных с лекций по С# работе с базами данных в Windows Forms.

Разработанное приложение отражает знания из предметной области, рассмотренной в курсовом проекте. Приложение содержит необходимые объекты, функциональность которых описана в пояснительной записке.

Таким образом, работа над созданием графического приложения для работы с базой данных предметной области военный комиссариат была завершена, что является достижением поставленной цели.

Курсовой проект основывается на условном примере, который не отражает всех особенностей предметной области военный комиссариат, поэтому в данной работе мы говорим о разработке приложения по управлению записями в базе данных. Однако выполнение данного курсового проекта позволило ознакомиться и научиться применять необходимые инструменты работы с базами данных, которые могут быть использованы для реальной предметной области.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Код. Тайный язык информатики - Чистяков И., Гончаров А.
2. Программирование на С# для начинающих - Васильев А.Н.
3. Pro Windows Forms 2.0 - Мэтью Макдоналд
4. Windows Forms в действии - Дэн Розенстарк
5. Паттерны проектирования на платформе .NET - Сергей Тепляков
6. Необходимые вещи в Windows Forms - Скотт Аллен
7. С# для профессионалов - Роман Шабельников
8. С# 9.0 вкратце - Джозеф и Бен Албанари
9. Искусство программирования — Дональд Кнут
10. С# - Турупалов Виктор Владимирович
11. Изучаем SQL - Алан Болье
12. SQL для чайников - Аллен Тейлор
13. SQL с нуля и быстро - Елена Литвак
14. Функциональный дизайн и архитектура — Александр Гранин
15. Чистый код — Роберт Мартин

ПРИЛОЖЕНИЕ А

Листинг 1. Program.cs

```
1 namespace WinFormsApp1
2 {
3     internal static class Program
4     {
5         static void Main()
6         {
7             ApplicationConfiguration.Initialize();
8             Application.Run(new Form1());
9         }
10    }
11 }
```

Листинг 2. Form1.cs

```
1 using System.Data.SQLite;
2
3 namespace WinFormsApp1
4 {
5     public partial class Form1 : Form
6     {
7         string connect = "Data Source = soldiers.db";
8         string pathToFile = "Soldiers";
9         public Form1()
10        {
11            InitializeComponent();
12        }
13
14        private void button1_Click(object sender, EventArgs e)
15        {
16            string text = textBox1.Text + " " + textBox2.Text + " " +
17            textBox3.Text + " " + textBox4.Text + " " + textBox5.Text + " " + textBox6.Text;
18            try
19            {
20                if (textBox1.Text.Length == 0 | textBox2.Text.Length == 0 |
21                textBox3.Text.Length == 0 | textBox4.Text.Length == 0 | textBox5.Text.Length ==
22                0 | textBox6.Text.Length == 0)
23                {
24                    throw new Exception("Одно из полей пустое!");
25                }
26                foreach (var x in textBox1.Text)
27                {
28                    if (Char.IsDigit(x))
29                    {
30                        throw new Exception("Поле 'ФИО' не может содержать
31                        цифры");
32                    }
33                }
34            }
35            catch { }
36        }
37    }
38 }
```

```

28             }
29             else if (Char.IsControl(x))
30             {
31                 throw new Exception("Поле 'ФИО' не может содержать
символы");
32             }
33         }
34         if (textBox2.Text.Length > 1)
35         {
36             throw new Exception("В поле 'категория годности' не
может быть более одной буквы");
37         }
38         foreach (var x in textBox2.Text)
39         {
40             if (Char.IsDigit(x))
41             {
42                 throw new Exception("В поле 'категория годности' не
может быть цифр");
43             }
44             else if (Char.IsControl(x))
45             {
46                 throw new Exception("В поле 'категория годности' не
может быть символов");
47             }
48         }
49         foreach (var x in textBox3.Text)
50         {
51             if (Char.IsDigit(x))
52             {
53                 throw new Exception("В поле 'род войск' не может
быть цифр");
54             }
55             else if (Char.IsControl(x))
56             {
57                 throw new Exception("В поле 'род войск' не может
быть символов");
58             }
59         }
60         foreach (var x in textBox5.Text)
61         {
62             if (Char.IsLetter(x))
63             {
64                 throw new Exception("В поле 'номер паспорта' не
может быть букв");
65             }
66             if (Char.IsControl(x))
67             {
68                 throw new Exception("В поле 'номер паспорта' не
может быть символов");
69             }
70         }
71         foreach (var x in textBox6.Text)
72         {
73             if (Char.IsLetter(x))
74             {

```

```

75             throw new Exception("В поле 'серия паспорта' не
может быть букв");
76         }
77         else if (Char.IsControl(x))
78         {
79             throw new Exception("В поле 'серия паспорта' не
может быть символов");
80         }
81     }
82     checkedListBox1.Items.Add(text);
83
84 }
85 catch (Exception ex)
86 {
87     MessageBox.Show("Ошибка! " + ex.Message);
88 }
89 }
90
91 private void button2_Click(object sender, EventArgs e)
92 {
93     try
94     {
95         if (checkedListBox1.Items.Count > 0)
96         {
97             checkedListBox1.Items.Clear();
98         }
99         else
100         {
101             throw new Exception("Список пуст, очищать нечего");
102         }
103     }
104     catch (Exception ex)
105     {
106         MessageBox.Show("Error! " + ex.Message);
107     }
108 }
109
110 private void button3_Click(object sender, EventArgs e)
111 {
112     Task task = button3_ClickTask();
113     task.Wait();
114     MessageBox.Show("Загрузка в БД завершена");
115 }
116 async private Task button3_ClickTask()
117 {
118     Soldier soldier = new Soldier(textBox1.Text, textBox2.Text,
textBox3.Text, textBox4.Text, textBox5.Text, textBox6.Text);
119
120     using (var connection = new SQLiteConnection(connect))
121     {
122         await connection.OpenAsync();
123         string createTable = @"
124 CREATE TABLE IF NOT EXISTS Soldiers(
125 id INTEGER NOT NULL PRIMARY KEY,
126 FIO nvarchar (50),

```

```

127         ValidityCategory nvarchar (50),
128         MilitaryBranch nvarchar (50),
129         Address nvarchar (50),
130         PassportNumber nvarchar (50),
131         PassportSeries nvarchar (50)
132     );
133     string sql = @"
134     INSERT INTO Soldiers(FIO, ValidityCategory, MilitaryBranch,
Address, PassportNumber, PassportSeries)
135     SELECT @FIO, @ValidityCategory, @MilitaryBranch, @Address,
@PassportNumber, @PassportSeries
136     ";
137     using (var command = new SQLiteCommand(createTable,
connection))
138     {
139         await command.ExecuteNonQueryAsync();
140     }
141
142     using (var command = new SQLiteCommand(sql, connection))
143     {
144         command.Parameters.AddWithValue("@FIO", soldier.FIO);
145         command.Parameters.AddWithValue("@ValidityCategory",
soldier.ValidityCategory);
146         command.Parameters.AddWithValue("@MilitaryBranch",
soldier.MilitaryBranch);
147         command.Parameters.AddWithValue("@Address",
soldier.Address);
148         command.Parameters.AddWithValue("@PassportNumber",
soldier.PassportNumber);
149         command.Parameters.AddWithValue("@PassportSeries",
soldier.@PassportSeries);
150         await command.ExecuteNonQuery();
151     }
152 }
153 }
154 }
155
156 private void button4_Click(object sender, EventArgs e)
157 {
158     checkedListBox1.Items.Remove(checkedListBox1.SelectedItem!);
159 }
160 private void button5_Click(object sender, EventArgs e)
161 {
162     foreach (var x in checkedListBox1.Items)
163     {
164         File.AppendAllText(pathToFile, $"{x.ToString()}\n");
165     }
166
167     MessageBox.Show("Запись в файл завершена!");
168 }
169
170 private void button6_Click(object sender, EventArgs e)
171 {
172     File.WriteAllText(pathToFile, "");
173     Application.Exit();

```



```

174     }
175
176     private void button7_Click(object sender, EventArgs e)
177     {
178         checkedListBox1.Items.Remove(checkedListBox1.SelectedItem!);
179     }
180
181     private void button9_Click(object sender, EventArgs e)
182     {
183         checkedListBox1.Items.Remove(checkedListBox1.SelectedItem!);
184     }
185
186     private void button10_Click(object sender, EventArgs e)
187     {
188         checkedListBox1.Items.Remove(checkedListBox1.SelectedItem!);
189     }
190
191     private void button8_Click(object sender, EventArgs e)
192     {
193         checkedListBox1.Items.Remove(checkedListBox1.SelectedItem!);
194     }
195 }
196 }

```

Листинг 3. Soldier.cs

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace WinFormsApp1
8 {
9     internal class Soldier
10    {
11        public string FIO { get; set; }
12        public string ValidityCategory { get; set; }
13        public string MilitaryBranch { get; set; }
14        public string Address { get; set; }
15        public string PassportNumber { get; set; }
16        public string PassportSeries { get; set; }
17        public Soldier(string FIO, string validityCategory, string
militaryBranch, string address, string passportNumber, string passportSeries)
18        {
19            this.FIO = FIO;
20            this.ValidityCategory = validityCategory;
21            this.MilitaryBranch = militaryBranch;
22            this.Address = address;
23            this.PassportNumber = passportNumber;
24            this.PassportSeries = passportSeries;
25        }

```

```
26     }
27 }
```

Листинг 4. Soldier.cs

```
1 namespace WinFormsApp1
2 {
3     partial class Form1
4     {
5         private System.ComponentModel.IContainer components = null;
6         protected override void Dispose(bool disposing)
7         {
8             if (disposing && (components != null))
9             {
10                 components.Dispose();
11             }
12             base.Dispose(disposing);
13         }
14
15         #region Windows Form Designer generated code
16         private void InitializeComponent()
17         {
18             textBox1 = new TextBox();
19             textBox2 = new TextBox();
20             textBox3 = new TextBox();
21             textBox4 = new TextBox();
22             textBox5 = new TextBox();
23             textBox6 = new TextBox();
24             button1 = new Button();
25             button2 = new Button();
26             button3 = new Button();
27             checkedListBox1 = new CheckedListBox();
28             button4 = new Button();
29             button5 = new Button();
30             button6 = new Button();
31             button7 = new Button();
32             button8 = new Button();
33             button9 = new Button();
34             button10 = new Button();
35             SuspendLayout();
36             //
37             // textBox1
38             //
39             textBox1.Location = new Point(12, 275);
40             textBox1.Name = "textBox1";
41             textBox1.Size = new Size(159, 23);
42             textBox1.TabIndex = 1;
43             textBox1.Text = "ФИО";
44             //
45             // textBox2
46             //
47             textBox2.Location = new Point(12, 304);
```

```

48     textBox2.Name = "textBox2";
49     textBox2.Size = new Size(159, 23);
50     textBox2.TabIndex = 2;
51     textBox2.Text = "Категория годности";
52     //
53     // textBox3
54     //
55     textBox3.Location = new Point(12, 333);
56     textBox3.Name = "textBox3";
57     textBox3.Size = new Size(159, 23);
58     textBox3.TabIndex = 3;
59     textBox3.Text = "Род войск";
60     //
61     // textBox4
62     //
63     textBox4.Location = new Point(12, 362);
64     textBox4.Name = "textBox4";
65     textBox4.Size = new Size(159, 23);
66     textBox4.TabIndex = 4;
67     textBox4.Text = "Адрес";
68     //
69     // textBox5
70     //
71     textBox5.Location = new Point(12, 391);
72     textBox5.Name = "textBox5";
73     textBox5.Size = new Size(159, 23);
74     textBox5.TabIndex = 5;
75     textBox5.Text = "Номер паспорта";
76     //
77     // textBox6
78     //
79     textBox6.Location = new Point(12, 420);
80     textBox6.Name = "textBox6";
81     textBox6.Size = new Size(159, 23);
82     textBox6.TabIndex = 6;
83     textBox6.Text = "Серия паспорта";
84     //
85     // button1
86     //
87     button1.Location = new Point(499, 261);
88     button1.Name = "button1";
89     button1.Size = new Size(289, 23);
90     button1.TabIndex = 7;
91     button1.Text = "Добавить";
92     button1.UseVisualStyleBackColor = true;
93     button1.Click += button1_Click;
94     //
95     // button2
96     //
97     button2.Location = new Point(499, 435);
98     button2.Name = "button2";
99     button2.Size = new Size(289, 23);
100    button2.TabIndex = 8;
101    button2.Text = "Очистить список";
102    button2.UseVisualStyleBackColor = true;

```

```

103         button2.Click += button2_Click;
104         //
105         // button3
106         //
107         button3.Location = new Point(499, 464);
108         button3.Name = "button3";
109         button3.Size = new Size(285, 23);
110         button3.TabIndex = 14;
111         button3.Text = "Загрузить в БД";
112         button3.UseVisualStyleBackColor = true;
113         button3.Click += button3_Click;
114         //
115         // checkedListBox1
116         //
117         checkedListBox1.FormattingEnabled = true;
118         checkedListBox1.Location = new Point(12, 17);
119         checkedListBox1.Name = "checkedListBox1";
120         checkedListBox1.Size = new Size(777, 238);
121         checkedListBox1.TabIndex = 16;
122         //
123         // button4
124         //
125         button4.Location = new Point(500, 290);
126         button4.Name = "button4";
127         button4.Size = new Size(289, 23);
128         button4.TabIndex = 17;
129         button4.Text = "Удалить выбранного";
130         button4.UseVisualStyleBackColor = true;
131         button4.Click += button4_Click;
132         //
133         // button5
134         //
135         button5.Location = new Point(500, 493);
136         button5.Name = "button5";
137         button5.Size = new Size(285, 23);
138         button5.TabIndex = 18;
139         button5.Text = "Записать в файл";
140         button5.UseVisualStyleBackColor = true;
141         button5.Click += button5_Click;
142         //
143         // button6
144         //
145         button6.Location = new Point(12, 513);
146         button6.Name = "button6";
147         button6.Size = new Size(135, 23);
148         button6.TabIndex = 19;
149         button6.Text = "ВЫХОД";
150         button6.UseVisualStyleBackColor = true;
151         button6.Click += button6_Click;
152         //
153         // button7
154         //
155         button7.Location = new Point(500, 319);
156         button7.Name = "button7";
157         button7.Size = new Size(289, 23);

```

```

158         button7.TabIndex = 20;
159         button7.Text = "Отправить на СВ0";
160         button7.UseVisualStyleBackColor = true;
161         button7.Click += button7_Click;
162         //
163         // button8
164         //
165         button8.Location = new Point(500, 348);
166         button8.Name = "button8";
167         button8.Size = new Size(289, 23);
168         button8.TabIndex = 21;
169         button8.Text = "Освободить от призыва";
170         button8.UseVisualStyleBackColor = true;
171         button8.Click += button8_Click;
172         //
173         // button9
174         //
175         button9.Location = new Point(500, 377);
176         button9.Name = "button9";
177         button9.Size = new Size(289, 23);
178         button9.TabIndex = 22;
179         button9.Text = "Отправить в учебный центр";
180         button9.UseVisualStyleBackColor = true;
181         button9.Click += button9_Click;
182         //
183         // button10
184         //
185         button10.Location = new Point(500, 406);
186         button10.Name = "button10";
187         button10.Size = new Size(289, 23);
188         button10.TabIndex = 23;
189         button10.Text = "Отправить в военный ВУЗ";
190         button10.UseVisualStyleBackColor = true;
191         button10.Click += button10_Click;
192         //
193         // Form1
194         //
195         AutoScaleDimensions = new.SizeF(7F, 15F);
196         AutoScaleMode = AutoScaleMode.Font;
197         ClientSize = new.SizeF(797, 548);
198         Controls.Add(button10);
199         Controls.Add(button9);
200         Controls.Add(button8);
201         Controls.Add(button7);
202         Controls.Add(button6);
203         Controls.Add(button5);
204         Controls.Add(button4);
205         Controls.Add(checkedListBox1);
206         Controls.Add(button3);
207         Controls.Add(button2);
208         Controls.Add(button1);
209         Controls.Add(textBox6);
210         Controls.Add(textBox5);
211         Controls.Add(textBox4);
212         Controls.Add(textBox3);

```

```

213         Controls.Add(textBox2);
214         Controls.Add(textBox1);
215         Name = "Военкомат";
216         Text = "Военкомат";
217         ResumeLayout(false);
218         PerformLayout();
219     }
220
221     #endregion
222     private TextBox textBox1;
223     private TextBox textBox2;
224     private TextBox textBox3;
225     private TextBox textBox4;
226     private TextBox textBox5;
227     private TextBox textBox6;
228     private Button button1;
229     private Button button2;
230     private Button button3;
231     private CheckedListBox checkedListBox1;
232     private Button button4;
233     private Button button5;
234     private Button button6;
235     private Button button7;
236     private Button button8;
237     private Button button9;
238     private Button button10;
239 }
240 }

```

ПРИЛОЖЕНИЕ Б

Информационная система управления деятельностью военкомата

ФИО|

Категория годности

Род войск

Адрес

Номер паспорта

Серия паспорта

ВЫХОД

Добавить

Удалить выбранного

Отправить на СВО

Освободить от призыва

Отправить в учебный центр

Отправить в военный ВУЗ

Очистить список

Загрузить в БД

Записать в файл

Рис. 1: Главное меню

- ☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
- ☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
- ☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
- ☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
- ☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
- ☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
- ☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123

Рис. 2: Добавление солдат

Информационная система управления деятельностью военкомата

☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
☒ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123
☐ Коваленко Владислав Петрович Д Ракетные войска Моревка 123 123

Рис. 3: Удаление определенного солдата