

*Бужинская Н.В.,*

*к.п.н., доцент*

*доцент кафедры информационных технологий*

*Нижнетагильский государственный социально-педагогический институт,*

*(ф) Российского государственного профессионально-педагогического*

*университета*

*Россия, г. Нижний Тагил*

*Котелевец Н.О.,*

*студент 4 курс,*

*факультет естествознания, математики и информатики*

*Нижнетагильский Государственный Социально-Педагогический*

*Институт, (ф) Российского государственного профессионально-*

*педагогического университета*

*Россия, г. Нижний Тагил*

## **РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА ДЛЯ АВТОМАТИЗАЦИИ УЧЕТА ОБОРУДОВАНИЯ НА ПРЕДПРИЯТИИ**

***Аннотация:** В настоящее время базы данных используются практически в любой области деятельности. Для их разработки могут использоваться различные средства, в том числе и языки программирования. В статье рассматриваются вопросы разработки базы данных для учета участников оборудования в городской поликлинике г. Нижнего Тагила.*

***Ключевые слова:** база данных, программирование, C#, WPF.*

***Annotation:** Currently, databases are used in almost every field of activity. Various tools, including programming languages, can be used to develop them. The article deals with the development of a database to account for the participants of the equipment in the city polyclinic of Nizhny Tagil.*

***Key words:** database, programming, C#, WPF.*

На современном этапе развития общества практически любая деятельность связана с использованием баз данных, так как многие системы в той или иной степени связаны с функциями долговременного хранения и обработки информации. При этом за счет увеличения объемов информации повысились и требования к скорости обработки данных. Например, административные решения требуют четкой и точной оценки ситуации и возможных перспектив ее изменения.

ГБУЗ СО «ГБ № 1 г. Н. Тагил» – современное многопрофильное лечебное учреждение. Это лечебно-профилактический комплекс, который оказывает первичную медико-санитарную помощь. Вопрос разработки программного продукта для учета оборудования в ГБУЗ СО «ГБ № 1 г. Н. Тагил» встал по причине отсутствия специализированного оборудования на предприятии.

Для разработки данного программного обеспечения используется язык программирования C#, а именно Windows Presentation Foundation (WPF) – подсистема для построения графических интерфейсов.

Для создания программы для автоматизации учета оборудования на предприятии, необходимо с помощью WPF разработать разметку с тремя вкладками:

- Оборудование;
- Программное обеспечение;
- Ремонты и обслуживание.

Ниже представлен код разметки на языке XAML.

Листинг 1

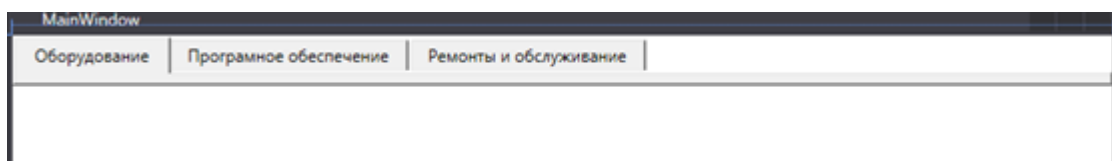
```
<Grid>
    <StackPanel>
        <TabControl>
            <TabItem>
                <TabItem.Header>
                    <TextBlock>Оборудование</TextBlock>
                </TabItem.Header>
            </TabItem>
        </TabControl>
    </StackPanel>
</Grid>
```

```

        <TabItem>
            <TabItem.Header>
                <TextBlock>Програмное
обеспечение</TextBlock>
            </TabItem.Header>
        </TabItem>
        <TabItem>
            <TabItem.Header>
                <TextBlock>Ремонты и
обслуживание</TextBlock>
            </TabItem.Header>
        </TabItem>
    </TabControl>
</StackPanel>
</Grid>
</Window>

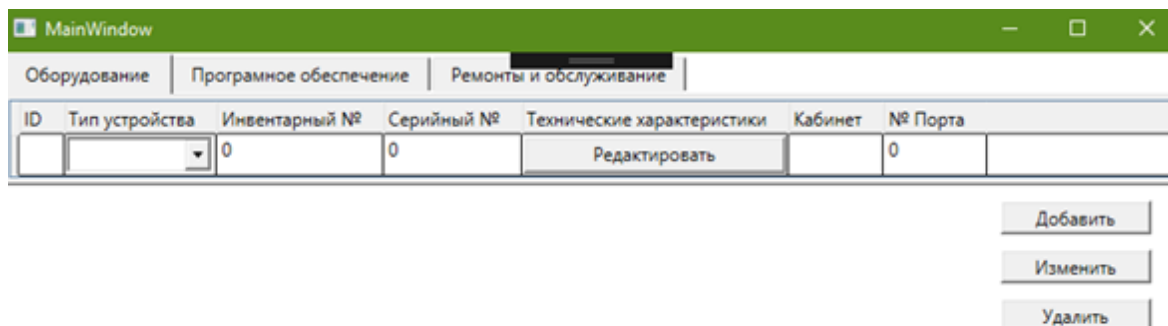
```

Итоговый результат представлен на рисунке 1.



**Рисунок 1. Начальная разметка**

Далее необходимо добавить элемент базы данных и задать необходимые поля. У каждой вкладки будут собственный набор полей. После окончания работы с разметкой необходимо настроить поля для возможности их заполнения и добавить кнопки.



**Рисунок 2. Добавление кнопок**

Аналогичным образом добавляем разметку для других вкладок.

Далее напишем код на языке С# для возможности заполнять базу данных, а именно таблицу «Оборудование». Для начала необходимо создать класс, представленный в листинге 2.

Листинг 2

```
public partial class MainWindow: Window
{
    public static int ID { get; set; }
    public static string Type { get; set; }
    public static int Invent { get; set; }
    public static int Serial { get; set; }
    public static string Tech { get; set; }
    public static string Kabinet { get; set; }
    public static int Port { get; set; }
    public static string IP { get; set; }
    public static string Date_buy { get; set; }
    public static string Garantia { get; set; }
    public static string The_end_garantia { get; set; }
    public static string Status { get; set; }
    public static string Date_add { get; set; }
```

Далее нужно создать структуру, которая будет обращаться к классу.

Листинг 3

```
struct MyData
{
    public MyData(int id, string type, int invent, int
serial, string tech, string kabinet, int port, string ip, string
date_buy, string garantia, string the_end_garantia, string status,
string date_add)
    {
        this.ID = id;
        this.Type = type;
        this.Invent = invent;
        this.Serial = serial;
        this.Tech = tech;
```

```

        this.Kabinet = kabinet;
        this.Port = port;
        this.IP = ip;
        this.Date_buy = date_buy;
        this.Garantia = garantia;
        this.The_end_garantia = the_end_garantia;
        this.Status = status;
        this.Date_add = date_add;
    }

    public int ID { get; set; }
    public string Type { get; set; }
    public int Invent { get; set; }
    public int Serial { get; set; }
    public string Tech { get; set; }
    public string Kabinet { get; set; }
    public int Port { get; set; }
    public string IP { get; set; }
    public string Date_buy { get; set; }
    public string Garantia { get; set; }
    public string The_end_garantia { get; set; }
    public string Status { get; set; }
    public string Date_add { get; set; }

}

```

Затем напишем код для заполнения БД в обработчике событий Loaded.

Листинг 4

```

private void EquipmentGrid_Loaded(object sender, RoutedEventArgs
e)
{
    MyData equipment = new MyData();
    List<MyData> result = new List<MyData>();

    {
        int id = equipment.ID;
        string type = equipment.Type;
    }
}

```

```

        int invent = Invent;
        int serial = equipment.Serial;
        string tech = equipment.IP + ", " +
equipment.Date_buy + ", " + equipment.Garantia + ", " +
equipment.The_end_garantia + ", " + equipment.Status + ", " +
equipment.Date_add;

        string kabinet = equipment.Kabinet;
        int port = equipment.Port;

        result.Add(new MyData(id, type,
invent, serial, kabinet, port));
        result.Add(new MyData(id, type, invent, serial,
kabinet, port));
        result.Add(new MyData(id, type, invent, serial,
kabinet, port));
        result.Add(new MyData(id, type, invent, serial,
kabinet, port));
        result.Add(new MyData(id, type, invent, serial,
kabinet, port));
    }
    equipmentGrid.ItemsSource = result;

```

В результате мы можем заполнять данными нашу таблицу «Оборудование».

ID	Тип устройства	Инвентарный №	Серийный №	Технические характеристики	Кабинет	№ Porta
	ПК	110560	198348	Редактировать	5	3
	Монитор	110561	146889	Редактировать	8	0
	ПК	110562	667498	Редактировать	8	8
	ПК	110563	584869	Редактировать	14	16
	Принтер	110564	498798	Редактировать	5	0

**Рисунок 3. Пример заполненной базы данных**

Таким образом, данный программный продукт определяет все подключенные устройства к сети. В дальнейшем для удобства работы системного администратора будет создана специальная форма на которой планируется реализовать схему всей сети предприятия, отобразить все коммутаторы и устройства, подключенные к ним.

### **ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ**

1. Албахари Д. С# 3.0: Справочник. СПб.: БХВ-Петербург, 2009. – 242 с. [Электронный ресурс]. URL: <http://padaread.com/?book=24312&pg=239> (дата обращения: 10.12.2018).

2. Ватсон К. С#. М.: Издательство Лори, 2004 – 879 с. [Электронный ресурс]. URL: [https://eknigi.org/nauka\\_i\\_ucheba/1386-s-karli-vatson.html](https://eknigi.org/nauka_i_ucheba/1386-s-karli-vatson.html) (дата обращения: 10.12.2018).

3. Гриффитс Й. Программирование на С# 5.0. М.: Издательство Эксмо, 2014. – 1134 с.

4. Макдональд М. WPF: Windows Presentation Foundation в .NET 4.0 с примерами на С# 2010 для профессионалов. – Издательство Вильямс, 2011 – 1019 с.