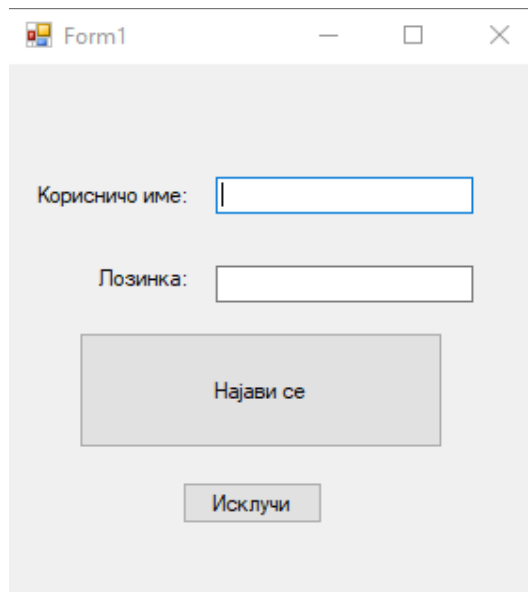


1. Опис на апликацијата

Апликацијата претставува систем со кој се контролира колку пијалоци има во барот. Во зависност дали е тоа корисник или вработен тие имат различни привилегии т.е. вработениот внесува пијалоци, додека корисникот купува.

2. Упатство за користење

2.1 Почетен екран за најава



The image shows a Windows-style window titled "Form1". Inside the window, there are two text input fields. The first field is labeled "Корисничко име:" and the second is labeled "Лозинка:". Below these fields are two buttons. The first button is labeled "Најави се" and the second button is labeled "Исклучи".

Кога се стартува програмата прво се прикажува прозорец за најава(слика 1). Овде може да се најави вработен или корисник. Корисничкото име за вработениот е "vraboten", додека за корисникот е "korisnik". Лозинката и за двата е "pas".

Нема да можеме да се најавиме додека не ги внесете точните податоци, а доколку погрешно внесеме се појавива прозорец кој не известува за тоа. Имаме и можост за прекин на програмата а тоа се прави со кликување на копчето "Исклучи".

2.2 Најавен како вработен

The screenshot shows a window titled 'Form2' with standard Windows window controls (minimize, maximize, close). Inside the window, there are three buttons: 'Внеси нов производ' (Add new product) at the top, 'Внеси веќе постоечки' (Add existing) below it, and 'Одјава' (Logout) at the bottom center.

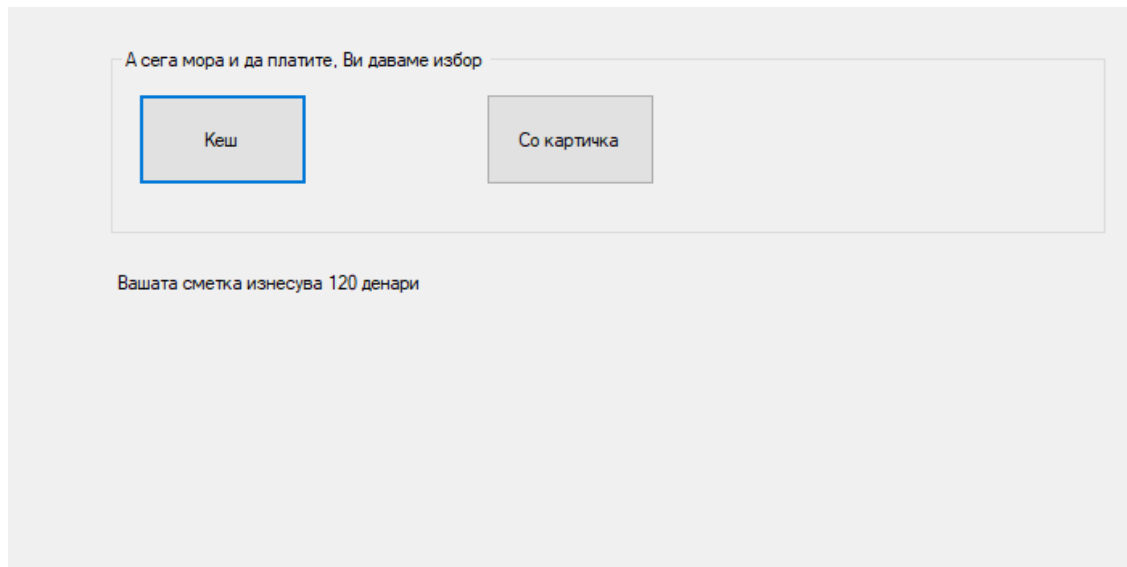
Овде имаме можност да избереме дали ќе внесеме нов производ или веќе постоечки. Доколку избереме да внесеме нов производ треба да внесеме име, количина и цена. Кај внесувањето на веќе постоечки пијалок треба да се селектира пијалок(само еден е дозволено) и треба да се внесе количина. Во било кој друг случај програмата нема да дозволи да внесете производ. Копчето за одјава не води на почетниот екран за најава.

2.3 Најавен како корисник

The screenshot shows the application interface for a logged-in user. On the left, under the heading 'Постоечки продукти:' (Existing products:), there is a list box containing one item: 'Pepsi-6-60'. To the right of this list, there is a text box with the instruction: 'Купи го селектираниот. Доколку не внесете количина се подразбира 1' (Buy the selected one. If you do not enter a quantity, it is assumed to be 1). Below this text box is an empty input field for quantity. Further down is a button labeled 'Плати' (Pay), and at the bottom is a button labeled 'Одјава' (Logout). On the far right, there is a large empty rectangular box, likely for displaying the total or receipt.

Го селектираме производот кој што сакаме да го купиме. Ако сакаме поголема количина од 1 ја внесуваме, доколку не внесеме ништо се зима дека сакаме да купиме 1. Производите кои што ги купуваме се внесуваат во листата за купени производи. Следен чекор е плаќањето или доколку ништо не сме купиле може да се одјавиме.

2.4 Плаќање

A screenshot of a payment selection interface. At the top, a message reads "А сега мора и да платите. Ви даваме избор" (Now you must pay. We give you a choice). Below this, there are two buttons: "Кеш" (Cash) and "Со картичка" (With card). The "Кеш" button is highlighted with a blue border. Below the buttons, a message states "Вашата сметка изнесува 120 денари" (Your bill amounts to 120 denari).

Овде треба да избереме дали ќе платиме во готово или со картичка.

2.5 Плаќање во готово

Ја внесуваме сумата која ја дава корисникот, а системот ни враќа повратна информација дали упешно сме платиле или не.

2.6 Плаќање со картичка

Се внесуваат точни бројки, бројот на картичката мора да е со 16 бројки а бројот на CVN треба да е со 3, доколку е се во ред плаќањето ќе биде успешно.

3. Претсавување на проблемот

3.1 Податочни структури

За да се реши овој проблем ја имам креирано класата "Pijalok" и 3 форми.

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace WindowsFormsApp17
8  {
9      [Serializable]
10     public class Pijalok
11     {
12         public String ime { get; set; }
13         public int kolicina { get; set; }
14         public int cena { get; set; }
15         public Pijalok(String s,int k,int c)
16         {
17             cena = c;
18             ime = s;
19             kolicina = k;
20         }
21         public override string ToString()
22         {
23             return String.Format("{0}-{1}-{2}", ime, kolicina,cena);
24         }
25     }
26 }
27
28

```

Овде го чуваме името на пијалокот, количината и цена.

```

[Serializable]
public partial class Form2 : Form
{
    public List<Pijalok> pijaloci=new List<Pijalok>();
    public List<Pijalok> pijalocil1 = new List<Pijalok>();
    public Form3 forma3;
    public int kv { get; set; } // 1-korisnik 0-vraboten
    public String FileName { get; set; }
}

```

Во Form2 имаме две листи од кои едната служи за сите пијалоци кои моментално ги имаме, а другата за пијалоците кои ги купува корисникот.

3.2 Сериализација

За да ги чуваме сите податоци кои ги внесува еден или различни вработени е искористено сериализаија. Кога играчот прв пат ја вклучува апликацијата на својот компјутер, се креира документ. Ако се направат промени, тие се ажурираат.