Coffee bar menu

1. Краток опис

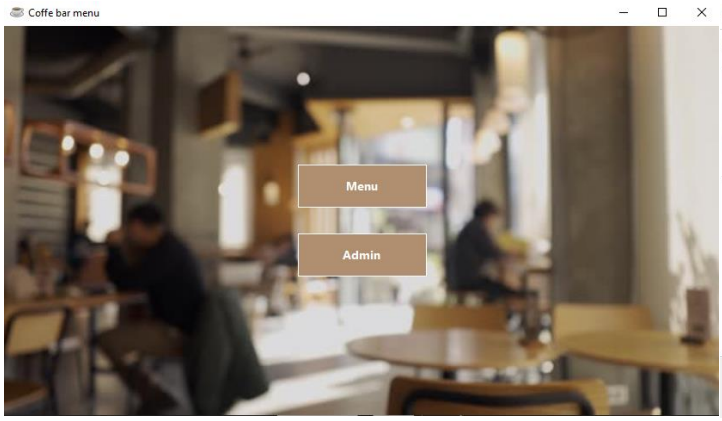
“Coffee bar menu” е апликација која е развиена се со цел да се олесни процесот на правење (генерирање) на едно почетно мени за еден кафе бар.

Главните две функционалности на апликацијата вклучуваат:

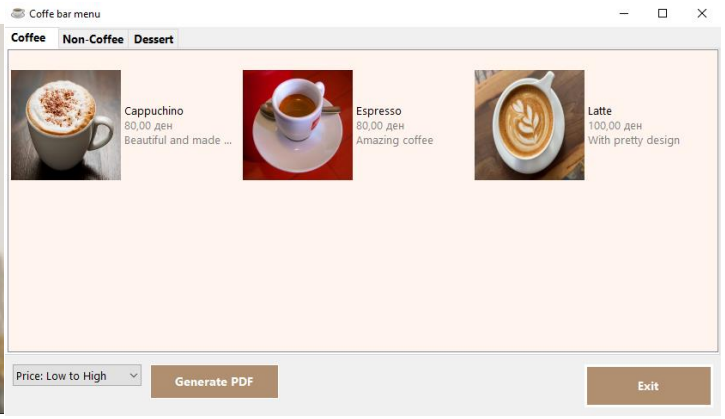
- Административен панел кој им овозможува на корисниците да менаџираат со ставките;
- Корисниците да имаат иницијален преглед на ставките по категории (coffee, Non-coffee и Deserts) и да го генерираат саканото мени во .pdf формат со сортирани ставки по желба;

Дополнително, сесијата секогаш се зачувува, па ставките нема да се избришат кога апликацијата ќе се затвори, овозможувајќи конзистентност на податоците. Со оваа апликација, управувањето со менито е едноставно и ефективно, овозможувајќи брза и лесна организација на производите.

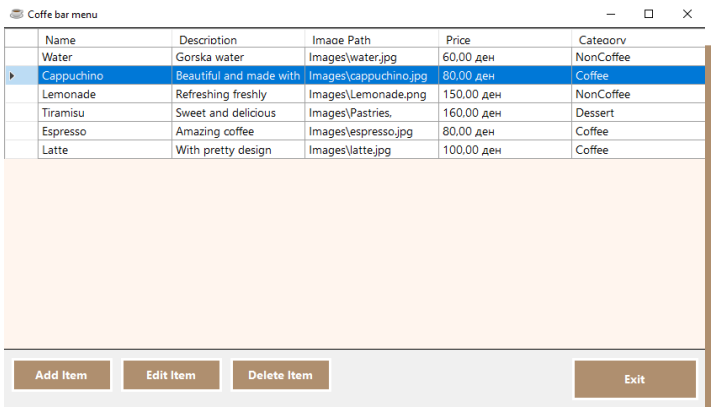
2. Упатство за користење



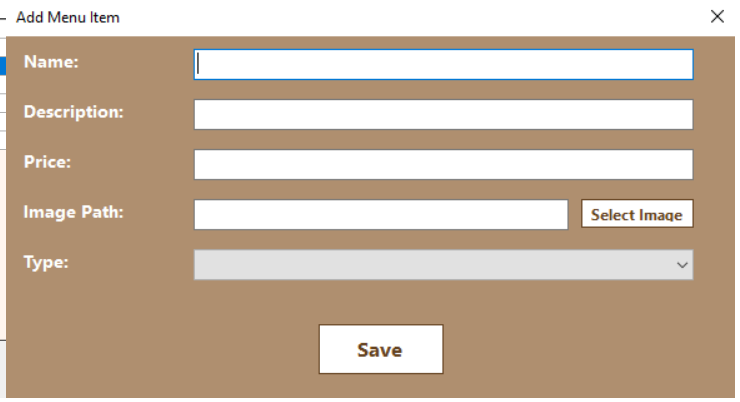
Home screen panel



Menu panel



Admin panel



Add item form

Апликацијата започнува со отварање на Home панелот. На него имаме опција за избор помеѓу две копчиња: “**Menu**” и “**Admin**”.

1. “**Menu**” копчето нè носи на Menu панелот. На него, можат да се видат ставките поделени по категории, односно тие се поделени во три tabs (Coffee, Non-Coffee и Deserts). Со едноставен клик на еден од нив може да се смени погледот кон одредена категорија.

   На панелот се наоѓа ComboBox кој што ни овозможува сортирање на стваките на четири начини:

   - Според името – во растечки редослед
   - Според името – во опаѓачки редослед
   - Според цената – во растечки редослед
   - Според цената – во опаѓачки редослед

   До ComboBox-от се наоѓа копчето GeneratePDF, кое овозможува генерирање на мени во .pdf формат, идентично на она како ставките се поделени по категории во панелот како и задржувајќи го одбраното сортирање.
2. “**Admin**” копчето нè носи на Admin панелот. На средина на панелот се наоѓа листа од сите ставки. Таа исто така може да се сортира со тоа што ќе се кликне на едно од полињата на header-от, дали според растечки или опаѓачки редослед соодветно.

   На дното се наоѓаат три копчиња AddItem, EditItem и DeleteItem.

   При клик на AddItem или EditItem се отвара нова форма во која можат да се додадат или променат атрибутите на ставките соодветно. Промените се зачувуваат со клик на копчето Save или при клик на *Enter* на тастатурата. Исклучоци: забрането е додавање на ставка со веќе постоечко име, забрането е не пополнување на сите полиња во формата, забрането е ставање на цена во не бројчан формат, забрането е додавање на слика со невалидна патека или формат на слика (дозволени формати .jpg и .png).

   Третото копче Delete овозможува отстранување на една селектирана ставка од листата. Пред тоа да се изведе на корисникот му се покажува MessageBox, кој ќе го праша дали е сигурен за тоа дали сака да ја отстрани соодветната ставка. Доколку одбере *OK*, ставката се отстранува доколку одбере *Cancel*, отстранувањето не се извршува. Истото може да се постигне и со клик на копчето *Delete* на тастатура.

На долниот десен агол на двата панели се наоѓа копчето Exit, кое овозможува враќање на Home панелот, дали при клик на него или со притискање на *Escape* преку тастатурата.

3. Податочни структури

Во оваа апликација, податоците се менаџирани во неколку податочни структури.

```
private List<BarMenuItem> items;
private ImageList imageList;
private Dictionary<string, int> imagePathIndexMap = new Dictionary<string, int>();
```

На почеток се наоѓа главната листа **items** која ги содржи самите ставки од менито, каде за секоја ставка чуваме име, опис, цена, категорија како и релативна патека до сликата.

```
public string Name { get; set; }
8 references
public string Description { get; set; }
30 references
public string ImagePath { get; set; }
12 references
public decimal Price { get; set; }
9 references
public ItemType Category { get; set; }
```

```
public enum ItemType
{
    Coffee,
    NonCoffee,
    Dessert
}
```

BarMenuItem.class

При вчитување на апликацијата, се повикува методот **InitializeBarMenuData()** каде се случува десеријализација користејќи BinaryFormatter, односно самата листа се иницијализира со соодветните ставки. Дополнително се иницијализира imageList листата со вчитување на сликите преку релативните патеки, а паралелно и речникот imagePathIndexMap каде за клуч се чува патеката од сликата, а како вредност индексот на сликата во imageList-от.

```
private ListView listViewCoffee;
private ListView listViewNonCoffee;
private ListView listViewDessert;
```

За да добиеме убав визуелен изглед за preview на самото мени во апликацијата, ја користиме ListView контролата, каде имаме по еден за секоја категорија (Coffee, Non-Coffee и Deserts). ImageList-от се доделува на LargeImageList property-то на овие три ListView-а, за подоцна при иницијализација на истите преку изминување на главната листа, да доделиме и ImageIndex на секој ListViewItem, кој пак ќе го извлечеме од самиот речник.

```
LargeImageList = imageList
listViewItem.ImageIndex = imagePathIndexMap[item.ImagePath]
```

ImageList-от е неопходен за ListView контролите, како би можеле да бидат прикажани самите слики од ставките, додека пак речникот служи како помошник за поедноставно менаџирање на ImageList-от при **CRUD** операциите врз ставките.

Исто така, при секоја од овие операции се повикува методот **SaveMenuItemData()**, каде се случува серијализација на главната листа items во датотеката menuItems.dat а сликите соодветно се зачувуваат и бришат во директориумот bin/Debug/Images.

Една од главните оптимизации која ја нуди оваа апликација е менаџирањето со сликите. Односно, не се чува за секоја ставка посебна слика, т.е. сите ставки доколку имаат иста слика таа ја споделуваат истата од директориумот bin/Debug/Images, за да се заштеди на простор.

4. Опис на една функција (GeneratePDF)

```
private void buttonGeneratePDF_Click(object sender, EventArgs e)
{
    GeneratePDF(items);
}
1 reference
private void GeneratePDF(List<BarMenuItem> items)
{
    SaveFileDialog saveFileDialog = new SaveFileDialog
    {
        Filter = "PDF files (*.pdf)|*.pdf",
        Title = "Save menu as PDF"
    };

    if (saveFileDialog.ShowDialog() == DialogResult.OK)
    {
```

Методот **buttonGeneratePDF_Click** е event handler за копчето GeneratePDF. Тој го повикува методото **GeneratePDF**, пренесувајќи му ја листата од BarMenuItem објектите (ставките).

Се користи SaveFileDialog за да му се овозможи на корисникот да ја избере локацијата и името за PDF-датотеката. Доколку се одбере Save процесот на продолжува и посакуваното мени се генерира на одбраната локација.

```
try
{
    string pdfPath = saveFileDialog.FileName;
    using (PdfWriter writer = new PdfWriter(pdfPath))
    {
        PdfDocument pdf = new PdfDocument(writer);
        iText.Layout.Document document = new iText.Layout.Document(pdf);

        Paragraph header = new Paragraph("Coffee Bar Menu").SetTextAlignment(TextAlignment.CENTER).SetFontSize(20).SetBold();

        document.Add(header);
        document.Add(new Paragraph("\n"));

        List<BarMenuItem> Coffees = new List<BarMenuItem>();
        List<BarMenuItem> NonCoffees = new List<BarMenuItem>();
        List<BarMenuItem> Deserts = new List<BarMenuItem>();

        foreach (var item in items)
        {
            switch (item.Category)
            {
                case ItemType.Coffee:
                    Coffees.Add(item);
                    break;
                case ItemType.NonCoffee:
                    NonCoffees.Add(item);
                    break;
                case ItemType.Dessert:
                    Deserts.Add(item);
                    break;
            }
        }

        addNewTable(Coffees, document);
        addNewTable(NonCoffees, document);
        addNewTable(Deserts, document);

        document.Close();
    }

    MessageBox.Show("PDF generated successfully.");
}
catch (Exception ex)
{
    MessageBox.Show($"Error generating PDF: {ex.Message}");
}
```

PdfWriter (на одбраната патека) и PdfDocument се инстанцирани за да се создаде PDF-датотеката. Се креира нов документ (iText.Layout.Document) и се додава параграф за заглавие со наслов „Menu“. Кодот потоа ги дели ставките од менито во три листи според категориите: Cafe, Non-Coffee и Deserts врз основа на нивниот тип што го имаат како атрибут.

```
public void addNewTable(List<BarMenuItem> menu, iText.Layout.Document document)
{
    if (menu.Count != 0)
    {
        Paragraph title = new Paragraph(menu[0].Category.ToString()).SetTextAlignment(TextAlignment.LEFT).SetFontSize(17).SetBold();

        document.Add(title);
        document.Add(new LineSeparator(new SolidLine()));

        Table table = new Table(UnitValue.CreatePercentArray(new float[] { 35, 65 }));
        foreach (var item in menu)
        {

            table.SetWidth(UnitValue.CreatePercentValue(100)); // Set the table to take the full width of the document

            if (File.Exists(item.ImagePath))
            {
                ImageData imageData = ImageDataFactory.Create(item.ImagePath);
                Image image = new Image(imageData).SetHeight(85);
                Cell imageCell = new Cell().Add(image).SetBorder(Border.NO_BORDER).SetPaddingTop(20);
                table.AddCell(imageCell);
            }
            else
            {
                // Add an empty cell if the image does not exist
                table.AddCell(new Cell().SetBorder(Border.NO_BORDER));
            }

            // Create the text content
            Paragraph textContent = new Paragraph().Add(new Text($"{item.Name} - {item.Price:C}" + "MKD.\n").SetFontSize(15)).Add(new Text(item.Description));

            Cell textCell = new Cell().Add(textContent).SetBorder(Border.NO_BORDER).SetPaddingTop(20);
            table.AddCell(textCell);
        }

        // Add the table to the document
        document.Add(table);
    }
}
```

Методот **addNewTable** се повикува за секоја листа. Секоја табела вклучува:

- Наслов на категоријата;
- Ќелија со слика на ставката;
- Ќелија со име, цена и опис на ставката;

На крајтот секоја табела се додава во документот со целосна ширина и соодветно форматирање.

Ако се појави некој исклучок за време на генерирањето PDF, на корисникот му се прикажува порака за грешка. Доколку се е успешно се прикажува пораката “PDF generated successfully”.