Café / Bar / Restaurant Retail Software

"Ss. Cyril and Methodius" University in Skopje
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

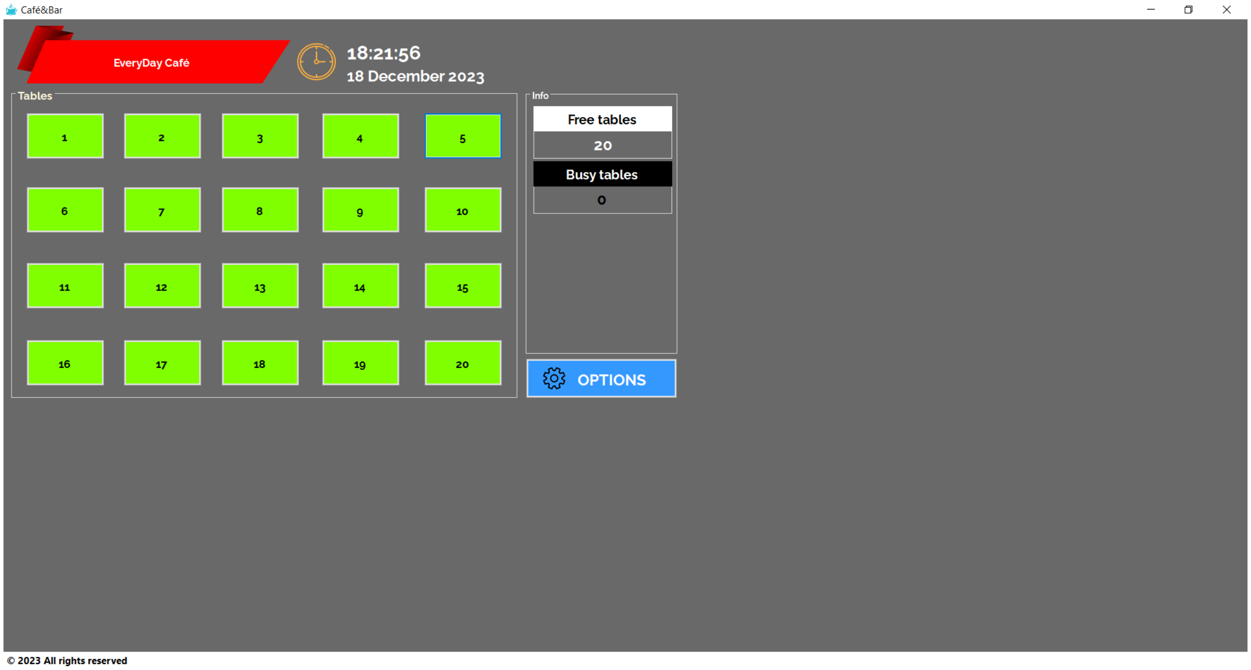
Developed by:	Alexxity

* Project developed as part of the Visual Programming course

The software is based on a level of visual display of content. The implementation is done in the C# programming language.

The main goal of the software is to provide easy manipulation of guests and their orders in one café/bar/restaurant without getting lost in its complexity. A light, simple and creative design has been implemented.

The software contains one main overview of the tables. The red color of the table indicates that the table is occupied, while the green color indicates that the table is free.



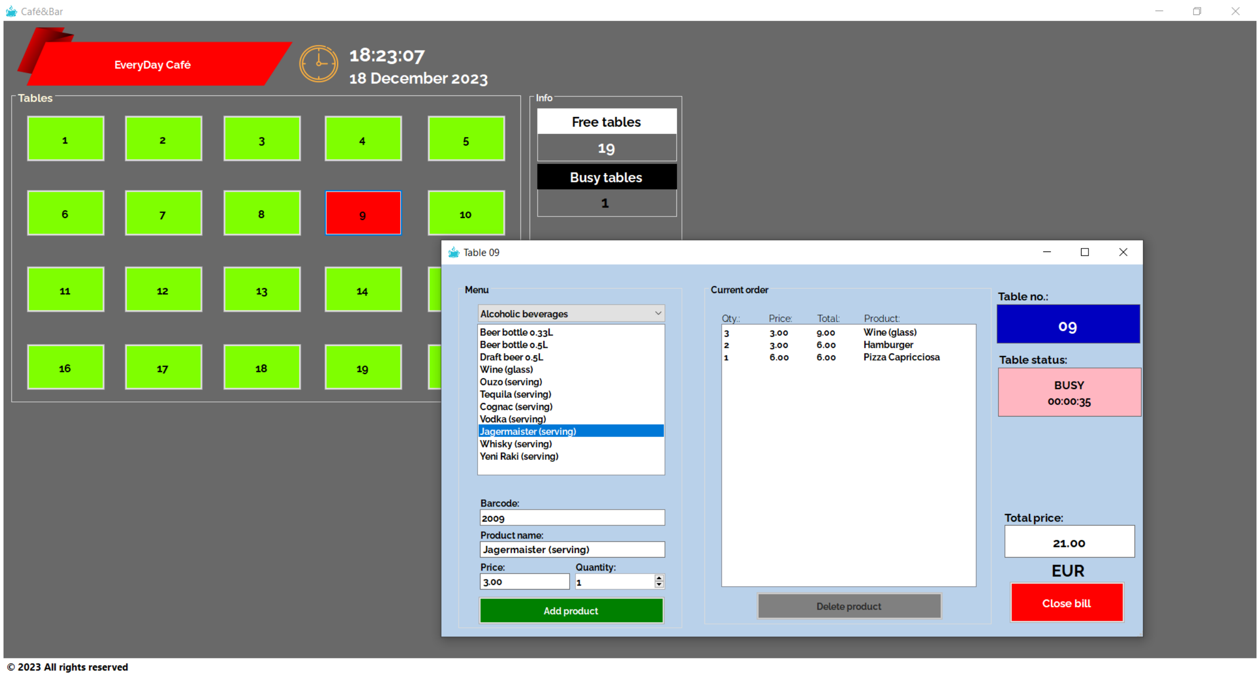
18:21:56
18 December 2023

Free tables
20

Busy tables
0

© 2023 All rights reserved

Opening a certain table displays a rich view of tools, options and information about that table. Three main options exist here: choosing a certain drink/food to add to that table (additional selection is also possible by barcode); the next option is an overview of what is currently ordered at the table, as well as an option to delete a certain drink/food from the bill/table; and the third option is to issue the receipt and add the receipt to the database of all receipts.



18:23:07
18 December 2023

Free tables
19
Busy tables
1

Table 09

Alcoholic beverages

- Beer bottle 0.33L
- Beer bottle 0.5L
- Draft beer 0.5L
- Wine (glass)
- Ouzo (serving)
- Tequila (serving)
- Cognac (serving)
- Vodka (serving)
- Jagermaister (serving)
- Whisky (serving)
- Yeni Raki (serving)

Barcode: 2009
Product name: Jagermaister (serving)
Price: 3.00
Quantity: 1

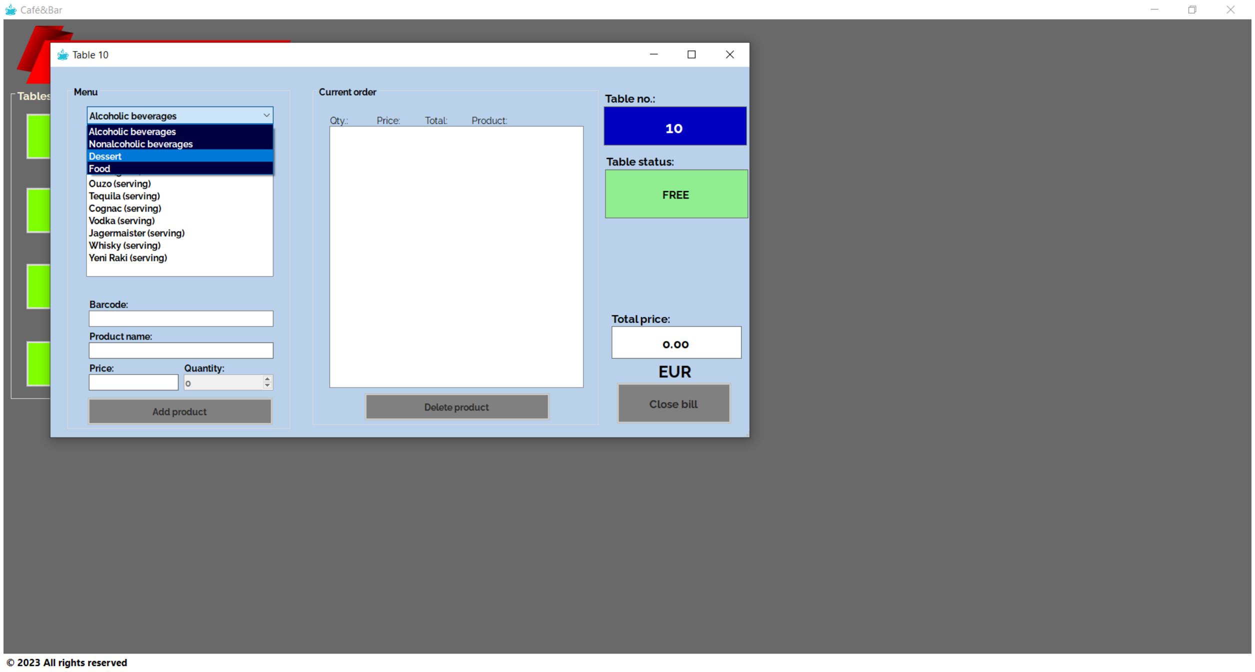
Current order

Qty.:	Price:	Total:	Product:
3	3.00	9.00	Wine (glass)
2	3.00	6.00	Hamburger
1	6.00	6.00	Pizza Capricciosa

Table no.: 09
Table status: BUSY 00:00:35
Total price: 21.00 EUR

Currently, the menu consists of 4 categories (can be easily extended):

- Alcoholic beverages
- Nonalcoholic beverages
- Desserts
- Food



Barcode:

Product name:

Price:

Quantity: 0

Table no.: 10

Table status: FREE

Total price: 0.00

EUR

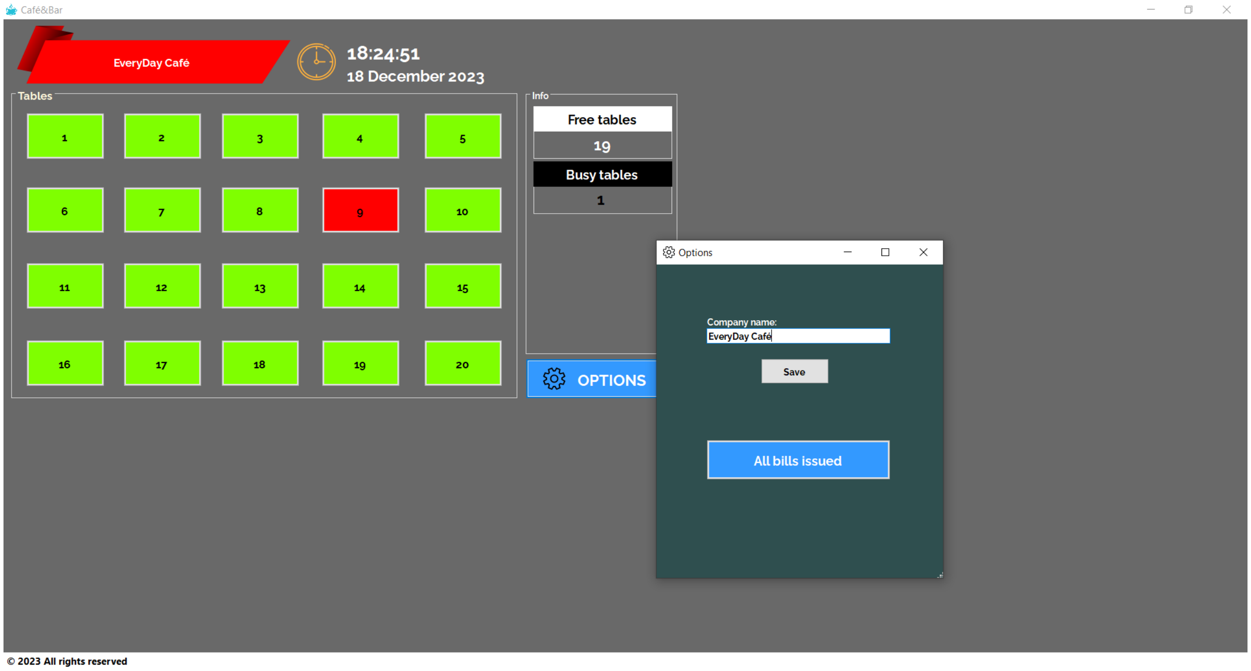
© 2023 All rights reserved

IMPORTANT:

Changing the menu of food and beverages is not allowed to be done by an employee. This can be achieved by modifying the database by an authorized manager only.

Additionally, there is an 'OPTIONS' setting where the name of the bar, café or restaurant can be changed.

In this view there is also an additional option to review all issued bills. They can be searched and displayed by a given date, but it is also possible to display all bills issued up until now. Detailed information can be seen separately for each bill (bill number, amount paid, when it was issued, which products it contains).

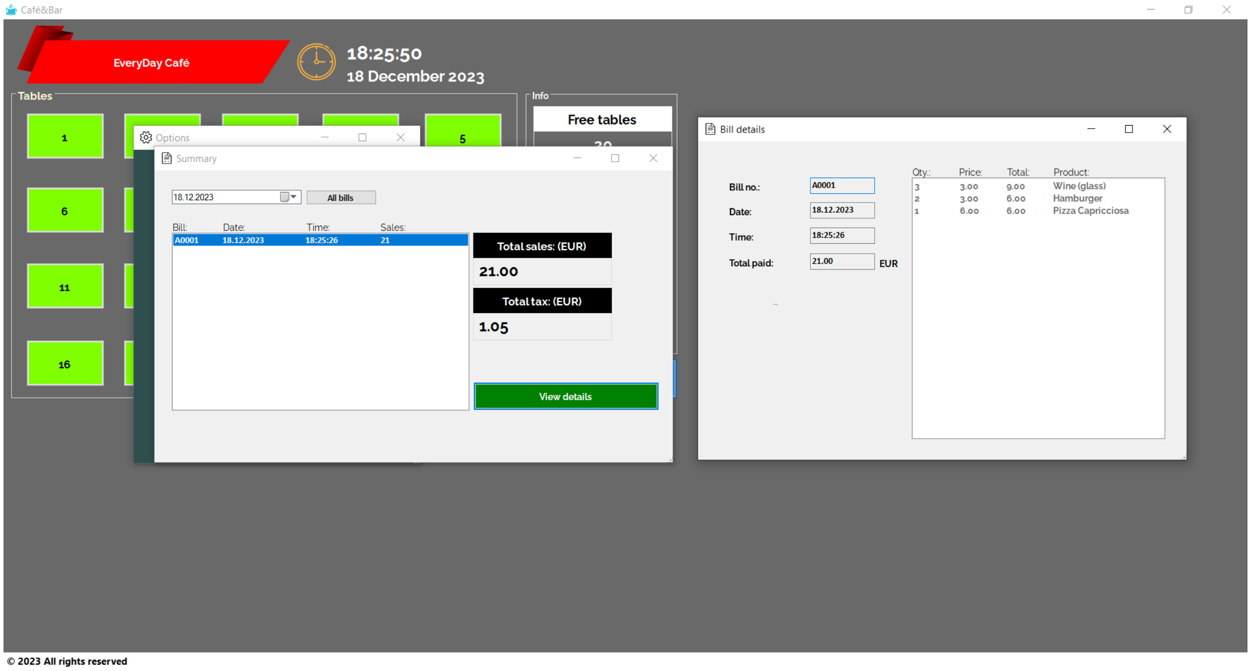


18:24:51
18 December 2023

Free tables
19
Busy tables
1

Company name: EveryDay Café

© 2023 All rights reserved



18:25:50
18 December 2023

Free tables

18.12.2023

Bill:	Date:	Time:	Sales:
A0001	18.12.2023	18:25:26	21

Total sales: (EUR)
21.00

Total tax: (EUR)
1.05

Bill no.: A0001
Date: 18.12.2023
Time: 18:25:26
Total paid: 21.00 EUR

Qty.:	Price:	Total:	Product:
3	3.00	9.00	Wine (glass)
2	3.00	6.00	Hamburger
1	6.00	6.00	Pizza Capricciosa

© 2023 All rights reserved

SOLUTION

One of the most important segments of this software is saving all the issued receipts somewhere in memory. That data later helps for statistical reviews and analyzes of the given company about its revenue.

Regarding the issued bills, it is implemented with a simple `Bill` class that contains:

- Bill number
- Date and time of issue
- Bill amount
- List of products

For saving all these bills, there is a class ***BillDoc*** which has a list of all the bills. To save its data in memory, serialization is used.

When opening the software initially, the code creates an instance of the ***BillDoc*** class that is ***null***.

```
public BillDoc billDoc { get; set; }
```

By using the **`LoadBills()`** function, this ***null*** instance now points to the previously created instance of the ***BillDoc*** class. While the software is active, changes are made at the software level, on the local **`billDoc`** instance. When closing the software, the **`SaveBills(BillDoc billDoc)`** function is called, which serializes (saves in the memory) the updated version of the local **`billDoc`** instance.

The main file from which the bill data is read and written is ***bills.dat***, located in **`software-en\Content\content`**.

Here are the two functions **`LoadBills()`** and **`SaveBills(BillDoc billDoc)`**:

```
private BillDoc LoadBills()
{
        string filePath = "../../Content/content/bills.dat";

        using (Stream stream = File.Open(filePath, FileMode.Open))
        {
                var binaryFormatter = new
                System.Runtime.Serialization.Formatters.Binary.BinaryFormatter();

                return (BillDoc)binaryFormatter.Deserialize(stream);
        }
}

private void SaveBills(BillDoc billDoc)
{
        string filePath = "../../Content/content/bills.dat";

        using (Stream stream = File.Open(filePath, FileMode.Create))
        {
                var binaryFormatter = new
                System.Runtime.Serialization.Formatters.Binary.BinaryFormatter();

                binaryFormatter.Serialize(stream, billDoc);
        }
}
```

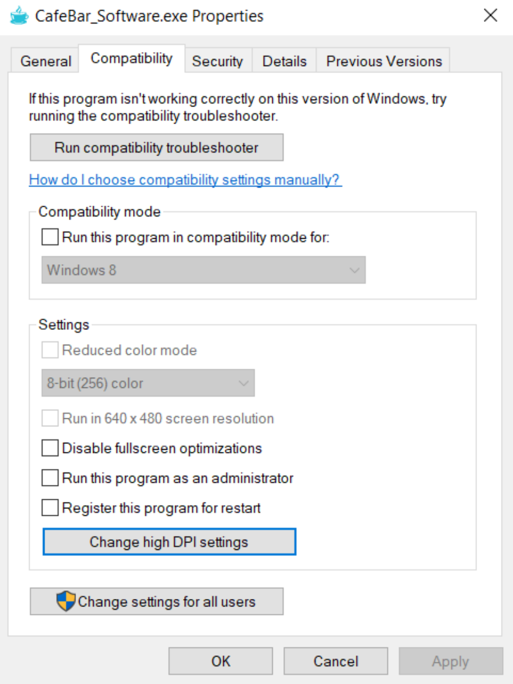
ADDITIONAL FUNCTIONALITIES

- When opening a given table *(ex. Table 5)*, the details for that table is opened and the *TextBox* for entering a barcode is directly focused. After entering the barcode and pressing *Enter*, the product quantity text box is focused. The next time you press *Enter*, the product is added to the order (with the entered number for quantity, initially set to 1). If the barcode does not exist, an error is displayed.
- Focused display with the café/bar/restaurant name, current time and today's date. This is achieved by using *timer*.
- It is impossible to delete a single product from the active table (an active order must have at least one product)
- Closing of the software not allowed without prior approval

IMPORTANT

In order to avoid visual errors in the software, before the initial run of the software, it is suggested to do the following:

1) Go into the folder `software-en\bin\Release`
2) Right click of the main `.exe` file `(CafeBar_Software.exe)`, then *Properties*
3) Go into the tab `Compatibility`, select the option `Override high DPI scaling behavior (or 'Change high DPI settings' and then the overriding)`
4) Save this *(OK)*


CafeBar_Software.exe Properties
General
Compatibility
Security
Details
Previous Versions
Run compatibility troubleshooter
How do I choose compatibility settings manually?
Change high DPI settings
Change settings for all users
OK
Cancel
Apply

If this program isn't working correctly on this version of Windows, try running the compatibility troubleshooter.

Compatibility mode

☐ Run this program in compatibility mode for:

Windows 8

Settings

☐ Reduced color mode

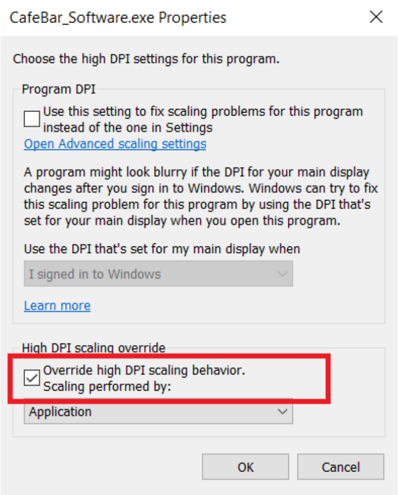
8-bit (256) color

☐ Run in 640 x 480 screen resolution

☐ Disable fullscreen optimizations

☐ Run this program as an administrator

☐ Register this program for restart


CafeBar_Software.exe Properties
Open Advanced scaling settings
Learn more
OK
Cancel

Choose the high DPI settings for this program.

Program DPI

☐ Use this setting to fix scaling problems for this program instead of the one in Settings

A program might look blurry if the DPI for your main display changes after you sign in to Windows. Windows can try to fix this scaling problem for this program by using the DPI that's set for your main display when you open this program.

Use the DPI that's set for my main display when

I signed in to Windows

High DPI scaling override

☑ Override high DPI scaling behavior. Scaling performed by:

Application

PREREQUISITES FOR THE SOFTWARE

- Have the fonts `Raleway` and `Calibri` installed
- Have `MySQL server` installed