**ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - ВАРНА**

**ФАКУЛТЕТ ПО ИЗЧИСЛИТЕЛНА ТЕХНИКА И АВТОМАТИЗАЦИЯ**

**КАТЕДРА: СОФТУЕРНИ И ИНТЕРНЕТ ТЕХНОЛОГИИ**

**СПЕЦИАЛНОСТ: СОФТУЕРНИ ТЕХНОЛОГИИ**

****

**Курсов проект по**

**“Технология на софтуерното производство”**

**на**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Имe** | ***Специалност*** | ***Фак.№*** | **Курс** | **Група** |
| **Едис Рейхан** | **СИТ** | **19621624** | **III** | **3a** |

**Срок за представяне: Водещ преподавател:**

**май 2022г. /доц. В. Божикова/**

I. ЗАДАНИЕ

**за курсов проект по дисциплината**

**“Технология на софтуерното производство”**

**Цел на курсовия проект:**

Изисква се да се разработи „Система за отчитане на стоки в хранителен магазин”.

**Функционални изисквания:**

Информационната система трябва да предоставя възможност за:

* поръчка и отчитане на стоки в хранителен магазин;
* добавяне/премахване на нов продукт;
* промяна на информация за съществуващ продукт(редакция на неговите атрибути);
* филтриране на продуктите по зададени категории;
* история на реализираните продажби;

Съществуват 2 вида потребители: администратор и продавач. Те трябва да изпълняват следните операции:

***Администратор:***

* добавяне, редактиране и премахване на даден продукт;
* **сортиране на продуктите по категории(плодове, зеленчуци, месни продукти, млечни продукти, напитки);**
* **добавяне, редактиране и премахване на продавач;**
* **добавяне, редактиране и премахване на категории продукти;**

**Продавач:**

* **избиране на продукти от списъка(въвеждане на количество за съответния продукт) и добавяне в кошницата;**
* **сортиране на списъка от продукти по категории;**
* **реализиране на продажба;**
* **отчет(история) на направените продажби, включващ информация за номер на фактура, име на продавач, дата на продажбата и общата сума за плащане;**

Системата трябва да съхранява данни (в БД) за:

* **продукти(Products);**
* **категории(Categories);**
* **продавачи(Sellers);**
* **фактури(Bills);**

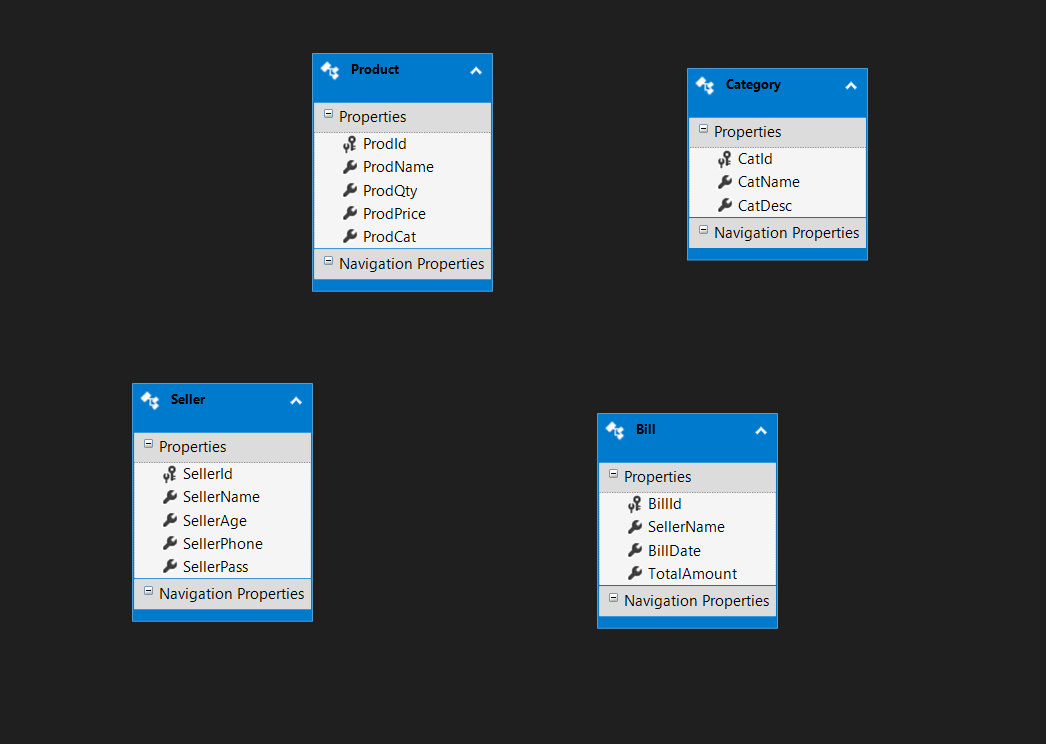
**Нефункционални изисквания:**

* Използва се форма за логване в системата от всички потребители (достъпа до системата се осъществява след въведено потребителско име и парола);
* Във всяка от таблиците в БД се реализират стандартните операции: въвеждане, корекция и изтриване на данни;
* генериране на необходимите справки: както на екран, така и на печат;

II. РЕАЛИЗАЦИЯ НА ПРОЕКТА

**1.Проектиране**

**-Описание на БД (ER диаграма):**

****

**2.Програмиране**

**Проектът е реализиран на езика C#, използвайки .NET фреймуърк. Използваната среда за разработка (IDE) e Visual Studio 2022.**

**Дефинирани са класовете *Splash, Form1, Product, Category, Seller, Selling.***

***Класът Splash представлява началната точка на приложението. Добавени са таймер и прогрес-бар, който се задейства когато приложението стартира. Когато прогрес-бара стигне до края, таймера спира и се отваря прозореца за логване на потребител.***

***Класът Form1 имплементира формата за логване на потребител в системата. За връзка с БД (SQL Server) се използва класа SqlConnection. Добавени са нужните контроли: label, textbox, combobox, button, circleButton. Написана е логика за валидиране на потребителското име и паролата, така че в системата да се логват само действителни потребители (съществуващи в БД).***

***Класът Product имплементира логика за управлението на продуктите. Написана е функцията showData(), която служи за селектиране на всички данни от таблица Product и визуализацията им в контрола DataGridView. За целта се използват класовете SqlDataAdapter, SqlCommandBuilder и DataSet, който представлява локалното копие на БД и визуализира информацията в табличен вид. Налице е и функцията fillCombo(), предназначена да попълва категориите в combobox контрола. При избор на съответната категория се показват само продуктите от тази категория. Добавени са бутони за добавяне, премахване и редактиране на продукт, които използват стандартни SQL заявки (CRUD операции). Бутонът Refresh служи за обновяване на предишното състояние на списъка с продукти.***

***Класът Category е за управление на категориите. Използвани са функциите, дефинирани в класът Product и са добавени бутони за аналогичните операции. Съществуват връзки от прозореца на този клас към класовете Seller и Product и изход от прозореца (препращане към логин формата).***

***Класът Seller имплементира логика за управление на продавачите. Налице са вече дефинираните и използвани функции и операции за въвеждане, редактиране и премахване на продавач. Съответно продавачите се показват в DataGridView. Съществува навигация към класовете Product и Category и възможност за препращане към логин страницата.***

***Класът Selling e имплементация на логика за продажба на продукти. Освен*** *showData() и fillCombo(), е дефинирана функцията showBillData(), която показва в DataGridView информация за номер на фактура, име на продавач, дата на продажба и общата сума за плащане. При отваряне на прозореца на класа Selling, добавените продукти се визуализират в DataGridView. От тях се избират продукти и се въвеждат количество и цена за всеки един. При натискане на бутон AddProduct продуктите се добавят в друг DataGridView. След избора на продукти и тяхното добавяне, се въвежда номер на фактура. С бутона Add се добавят съответните продукти с необходимата информация в списъка с продажбите (SellList). Добавен е label, който служи за визуализиране на актуалната дата във формат ден/месец/година. За целта се използва структурата DateTime. Бутонът Print служи за отпечатване на фактура за избраната продажба. Използва се системният клас Printing и класът Graphics за изрисуване на страницата. Разпечатката съдържа информация за: номер на фактура, име на продавач, дата и сума за плащане. Добавена е картина в контекста на приложението с помощта на класа Image, който чрез метода FromFile() създава изображение от подадения като параметър път на съществуващ файл.*

III. ПРИЛОЖЕНИЕ

**Тестови резултати:**

****



