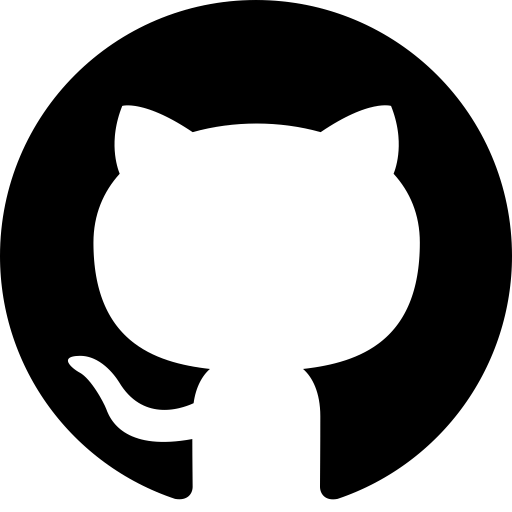


**[](https://github.com/imsalione/DooghAbali)**

بسمه‌تعالی

**فرم شناسنامه پروژه**

**درس آزمایشگاه مهندسی نرم‌افزار**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **دانشگاه آزاد اسلامی واحد پردیس- گروه کامپیوتر و فناوری‌اطلاعات** | | | | | | | | |
| **شماره گروه: ۱۸** | | | | | | | | |
| **ردیف** | **نام و نام خانوادگی** | **موقعیت شغلی** | **شماره دانشجویی** | | | | **لینک تلگرام** | |
| 1 | صالح عابدی‌نژاد | Backend developer  Databases Administrator | 40150140893010 | | | |  | |
| 2 | امیرحسین بیگی | UI/UX Designer | 4015140893009 | | | |  | |
| **موضوع پروژه**: تولید نرم افزار ثبت انبار دوغ آبعلی | | | | | | | | |
| **تعریف پروژه:** طراحی و پیاده سازی یک سیستم مدیریت فروش با استفاده از زبان برنامه نویسی سی شارپ و بانک اطلاعات MS sql server  **هدف:** ایجاد یک نرم افزار کاربردی و کارآمد برای مدیریت امور مختلف از جمله ثبت و نگهداری داده‌های فروش به استان‌ها و تهیه گزارش‌های موردی در زمان‌های مختلف  **خروجی:**   * یک نرم افزار تحت ویندوز با رابط گرافیکی کاربرپسند و منوهای مختلف برای انجام مربوط به ثبت داده‌ها * یک پایگاه داده رابطه‌ای برای ذخیره و بازیابی اطلاعات مورد نیاز سیستم * یک مستند فنی شامل توضیحات در مورد نیازمندی‌ها، طراحی، پیاده سازی، آزمون و نگهداری سیستم * یک مستند کاربردی شامل راهنمای نصب، راه اندازی و استفاده از نرم افزار * یک مجموع آزمون شامل موارد آزمون و نتایج آن‌ها برای ارزیابی کیفیت و عملکرد سیستم   **نتایج و فواید پروژه:**   * افزایش کارایی و بهره‌وری سیستم فروش شرکت دوغ آبعلی با کاهش زمان در ثبت و گزارش‌گیری * افزایش کیفیت خدمات شرکت با ارائه اطلاعات دقیق و به روز به شرکت‌های طرف قرارداد در استان‌ها * افزایش امنیت و حفاظت از اطلاعات حساس دوغ آبعلی با استفاده از روش‌های ثبت اطلاعات در بانک‌های اطلاعاتی * افزایش همکاری و ارتباط با سامانه شرکت‌های ذینفع در استان‌ها با استفاده از رابط کاربری مناسب | | | | | | | | |
| **زبان برنامه نویسی**: C# | | | | | **بانک اطلاعاتی**: MS SQL server | | | |
| **سایر ابزارهای مورد استفاده**: visual studio, wpf, corel, photoshop and other software | | | | | | | | |
| **ردیف** | **عنوان** | | | **تاریخ تحویل** | | **درصد پیشرفت** | | **تاریخ بررسی/ارزیابی** |
| 1 | برنامه زمانبندی | | | 10/08/1402 | | 100 | |  |
| 2 | شرح مساله؛ نتیجه مطالعات؛ بررسی کارهای مشابه | | | 15/08/1402 | | 100 | |  |
| 3 | Activity Diagram , Use Case Diagram | | | 20/08/1402 | | 100 | |  |
| 4 | Sequence Diagram , Class Diagram | | | 25/08/1402 | | ۱۰۰ | |  |
| 5 | طراحی بانک اطلاعاتی | | | 30/08/1402 | | 100 | |  |
| 6 | طراحی فرم ها | | | 05/09/1402 | | ۴۰ | |  |
| 7 | طراحی گزارشات | | | 10/09/1402 | | 0 | |  |
| 8 | نسخه اولیه نرم افزار | | | 15/09/1402 | | 40 | |  |
| 9 | تست پذیرش و آزمایش با داده های مختلف | | | 20/09/1402 | | ۱۰ | |  |
| 10 | نسخه نهایی نرم افزار | | | 25/09/1402 | | 0 | |  |
| 11 | راهنمای کاربران | | | 25/09/1402 | | ۳۰ | |  |
| 12 | راهنمای نصب | | | 25/09/1402 | | 0 | |  |

**فهرست مطالب**

**[شرح مسأله](#_Toc153901765)** [3](#_Toc153901765)

**[نتیجه مطالعات](#_Toc153901766)** [3](#_Toc153901766)

**[بررسی کارهای مشابه](#_Toc153901767)** [3](#_Toc153901767)

**[چرا از فریمورک WPF در ساخت این نرم افزار استفاده شده؟](#_Toc153901768)** [4](#_Toc153901768)

[Use Case Diagram 1](#_Toc153901769)

[Activity Diagram 2](#_Toc153901770)

[Class Diagram 3](#_Toc153901771)

[Sequence Diagram 4](#_Toc153901772)

[1](#_Toc153901773)

**[بانک اطلاعات نرم افزار بر اساس ساختار T-Sql](#_Toc153901774)** [1](#_Toc153901774)

**[Database Diagram](#_Toc153901775)** [3](#_Toc153901775)

[1](#_Toc153901776)

**[نسخه اولیه نرم افزار](#_Toc153901777)** [1](#_Toc153901777)

**[اطلاعات کلی نرم افزار:](#_Toc153901778)** [1](#_Toc153901778)

[صفحه‌ی ورود به نرم افزار (Login page) 1](#_Toc153901779)

[صفحه اصلی نرم افزار (Main page) 2](#_Toc153901780)

# **شرح مسأله**

دوغ آبعلی یک شرکت فرآورده‌های لبنیاتی و شیر می‌باشد. این شرکت اخیرا بخاطر فروش بالا و سیستم ثبت کاغذی فروش محصولات خود دچار مشکلاتی برای مدیریت فروش در سطح کشور شده و گزارشات بعضا هفته‌ها طول می‌کشد تا آماده شود. واحد تحقیق و توسعه مجاب شدند تا نرم افزاری جهت ثبت فروش خریداری نمایند. هدف نهایی این است که به وسیله نرم افزار ثبت فروش و همچنین گزارش فروش با سرعت بیشتری صورت پذیرد.

## **نتیجه مطالعات**

شرکت دوغ آبعلی محصولاتی همچون دوغ یک نفره، دوغ خانواده، دوغ ۳ لیتری را تولید می‌کند. این شرکت با توان تولیدی ۱۰۰۰ عدد در هر محصول به ازای یک روز، قصد دارد تولیدات خود را دو برابر کرده و میزان فروش خود را به ۲۰۰۰ محصول در طول یک روز برساند. از جمله موانع پیش‌روی شرکت ثبت به موقع فروش محصول در استان‌ها می‌باشد که به علت دستی بودن ثبت این اطلاعات، تولید محصول از فروش آن عقب‌تر می‌باشد. این شرکت محصولات خود را در پنج کلان استان تهران، فارس، اصفهان، خراسان رضوی و آذربایجان غربی به فروش می‌رساند. نحوه توزیع محصول در هر استان به تناسب جمعیت آن استان می‌باشد. در جدول زیر پراکندگی فروش این شرکت و همچنین نوع محصولات جهت تهیه بانک اطلاعاتی آورد شده است:

|  |  |  |  |
| --- | --- | --- | --- |
| استان | میزان جمعیت (میلیون نفر) | میزان فروش محصول | میزان فروش هدف |
| تهران | 13 | 406/25 | 812/5 |
| فارس | 5 | 156/25 | 312/5 |
| اصفهان | 5 | 156/25 | 312/5 |
| خراسان رضوی | 6 | 187/5 | 375 |
| آذربایجان غربی | 3 | 93/75 | 187/5 |
| جمع کل | 32 | 1000 | 20000 |

## **بررسی کارهای مشابه**

|  |  |
| --- | --- |
|  |  |
| شرکت پگاه با تولید فراورده‌های لبنی متنوع از جمله شرکت‌های پیشتاز در این حوزه می‌باشد. این شرکت با داشتن تنوع بسیار زیاد در محصولات لبنی، شامل 27 نوع دوغ در طعم‌ها و اندازه‌های مختلف است که مدیریت فروش آن با نرم افزارهای فروش تحت وب انجام می‌شود. سیستم فروش محصول این شرکت بصورت b2b بوده و استان‌های متقاضی با ثبت میزان درخواست خود بصورت آنلاین، بصورت دلخواه می‌توانند برای روزهای آتی خود سفارش دهند. این سیستم برای شرکت پگاه بسیار کاربردی بوده و می‌تواند بسته به میزان درخواست خود در روزهای آتی، میزان تولید محصولات خود را تنظیم نماید. | دوغ ماهشام یک دوغ پروبیوتیک گرماندیده است که اصلاحا بهش دوغ یخچالی میگوییم . که سرشار از انواع ویتامین و مواد مغذی موجود در ماست هم هست. این شرکت به تمام استان‌های کشور محصول ارسال می‌کند و شما می‌توانید با سفارش در شهر یا شهرستان خود به میزان دلخواه، محصولات خود را سفارش دهید. نحوه سفارش از طریق تماس با شرکت بوده و اپراتور بعد از ثبت سفارش تلفنی اطلاعات سفارش را در نرم افزار ثبت سفارش این شرکت ثبت نموده و بعد از طی فرآیند تولید محصول، این سفارشات در تاریخ تعیین شده به دست عمده فروش استان یا شهرستان می‌رسد. |

# **چرا از فریمورک WPF در ساخت این نرم افزار استفاده شده؟**



WPF یا Windows Presentation Foundation یک فریم‌ورک برنامه‌نویسی است که توسط مایکروسافت برای توسعه برنامه‌های دسکتاپ ویندوز طراحی شده است. این فریم‌ورک به برنامه‌نویسان امکان می‌دهد برنامه‌های گرافیکی و ظاهری زیبا و پویا را برای ویندوز ایجاد کنند. تعدادی از ویژگی‌های کلیدی WPF عبارتند از:

**1. (Extensible Application Markup Language)**

این تکنولوژی از زبان XAML برای تعریف و طراحی رابط کاربری استفاده می‌کند. این زبان امکان تعریف گرافیکی و ویژگی‌های رابط کاربری را به صورت اعلانی فراهم می‌کند.

**2. Data Binding:**

از مفهوم Data Binding استفاده می‌کند که امکان ارتباط داده‌ها و رابط کاربری را فراهم می‌کند. این به ما این امکان را می‌دهد که تغییرات در داده‌ها به صورت خودکار در رابط کاربری نمایش دهیم. به طور مثال، ما می‌توانیم یک لیست از اشیاء را به یک لیست باگتریو ربط دهید تا تغییرات در لیست به صورت خودکار در رابط کاربری نمایش داده شود.

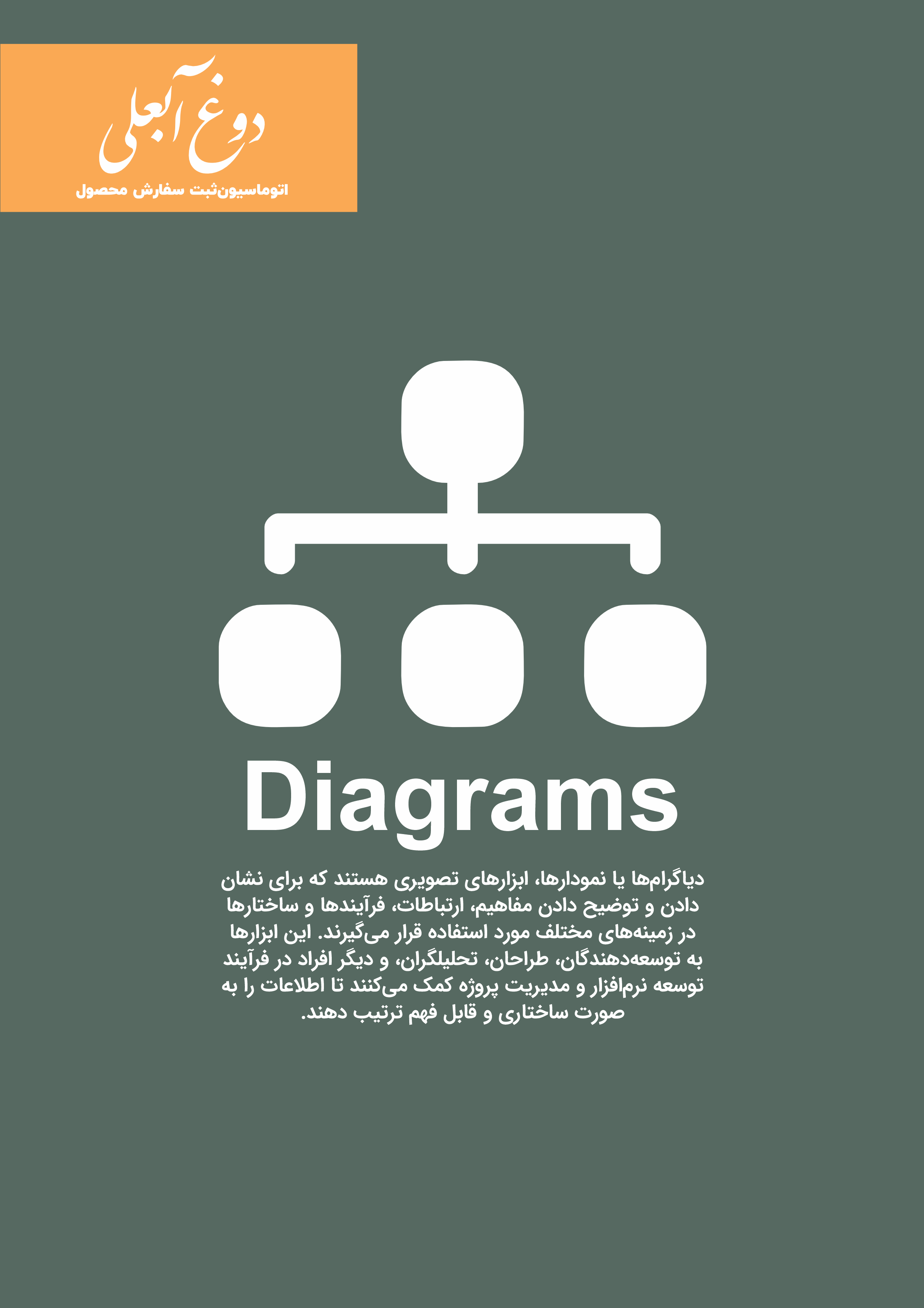
**3. Styles and Templates:**

از استایل‌ها و قالب‌ها برای تنظیم و طراحی ظاهر رابط کاربری استفاده می‌کند. این ویژگی‌ها به ما اجازه می‌دهند تا ظاهر را به طور یکنواخت تعیین کنیم و از قالب‌ها استاندارد برای سفارشی‌سازی عناصر کنترل استفاده نماییم.

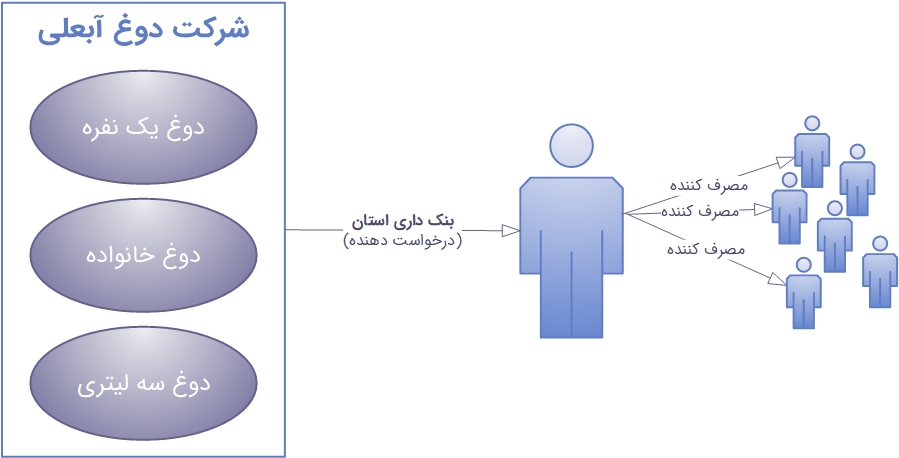
**4. Graphics and Multimedia Support**

از گرافیک‌های پیچیده و چندرسانه‌ای پشتیبانی می‌کند. این امکان را به ما می‌دهد تا تصاویر، ویدئوها، و انیمیشن‌های پویا را به راحتی به برنامه‌ی خود اضافه کنیم.

از آنجا که WPF مختص ویندوز است، به خوبی با اکوسیستم ویژه ویندوز ترکیب شده و به ما این امکان را می‌دهد که برنامه‌های دسکتاپ قدرتمند و با ظاهر زیبا ایجاد کنیم. برنامه‌های WPF بیشتر با استفاده از XAML تعریف می‌شوند. XAML یک زبان اعلانی است که به برنامه‌نویسان این امکان را می‌دهد تا رابط کاربری برنامه را با استفاده از تگ‌ها و ویژگی‌ها تعریف کنند. این کار امکان افزودن المان‌های گرافیکی، کنترل‌ها و ویژگی‌های دیگر را بسیار ساده می‌کند.

****

## Use Case Diagram



Use Case Diagram یک نمودار در مهندسی نرم‌افزار است که برای نمایش تعامل بین نقش‌های مختلف در یک سیستم استفاده می‌شود. این نمودار به صورت گرافیکی مانند شکل بالا نمایش می‌دهد که هر نقش چه وظایف و عملیاتی در سیستم دارد و چگونه با یکدیگر ارتباط دارند. در واقع، از Use Case Diagram برای مدل‌سازی نیازمندی‌های کلان سیستم و نحوه تعامل با کاربران استفاده می‌شود.

**بعضی از عناصر کلیدی Use Case Diagram عبارتند از:**

1. استفاده‌ها: یک Use Case نشان‌دهنده یک عملیات یا وظیفه خاص است که توسط سیستم انجام می‌شود. معمولاً به عنوان یک دستورالعمل یا یک سناریوی استفاده از سیستم توصیف می‌شود.

2. بازیگر: یک بازیگر نشان‌دهنده یک نقش یا یک کاربر است که با سیستم تعامل دارد. این می‌تواند فرد، سازمان یا سیستم دیگر باشد.

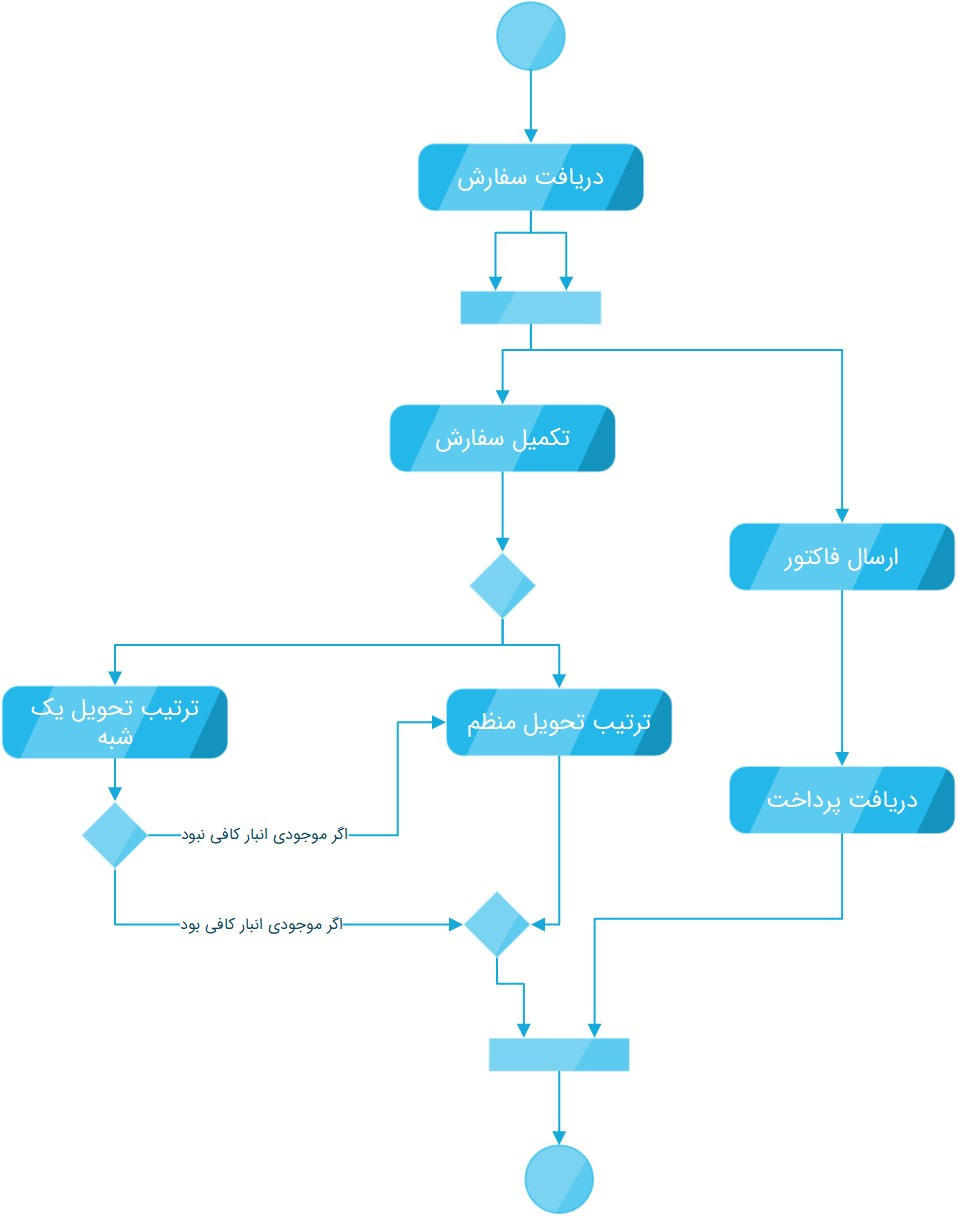
3. ارتباطات: خطوط اتصال بین Use Caseها و Actorها نشان‌دهنده رابطه و تعامل بین آنها هستند. این خطوط ممکن است با نمایانگر‌های خاصی مانند شکل‌های رأسی یا مثلثی برچسب‌گذاری شوند.

4. مرز سیستم: یک مرز یا مرز سیستم، محدوده فعالیتهای مورد بررسی را نشان می‌دهد و سیستم را از محیط خارجی جدا می‌کند.

Use Case Diagram به ما کمک می‌کند تا به طور سریع و ساده نیازمندی‌های سیستم و تعاملات مهم با کاربران را درک کنیم. این نمودار به توسعه‌دهندگان، تحلیلگران، و سایر افراد در فرآیند توسعه نرم‌افزار کمک می‌کند تا به درستی نیازمندی‌ها را درک کرده و به اشتراک بگذارند.

نوع فروش محصول به صورت کلی در دوغ آبعلی به این صورت می‌باشد که ابتدا محصول توسط عمده فروش در استان سفارش داده می‌شود سپس خورده فروش‌ها در استان محصول را دریافت می‌کنند و در نهایت محصول به دست مشتری یا مصرف کننده می‌رسد.

## Activity Diagram

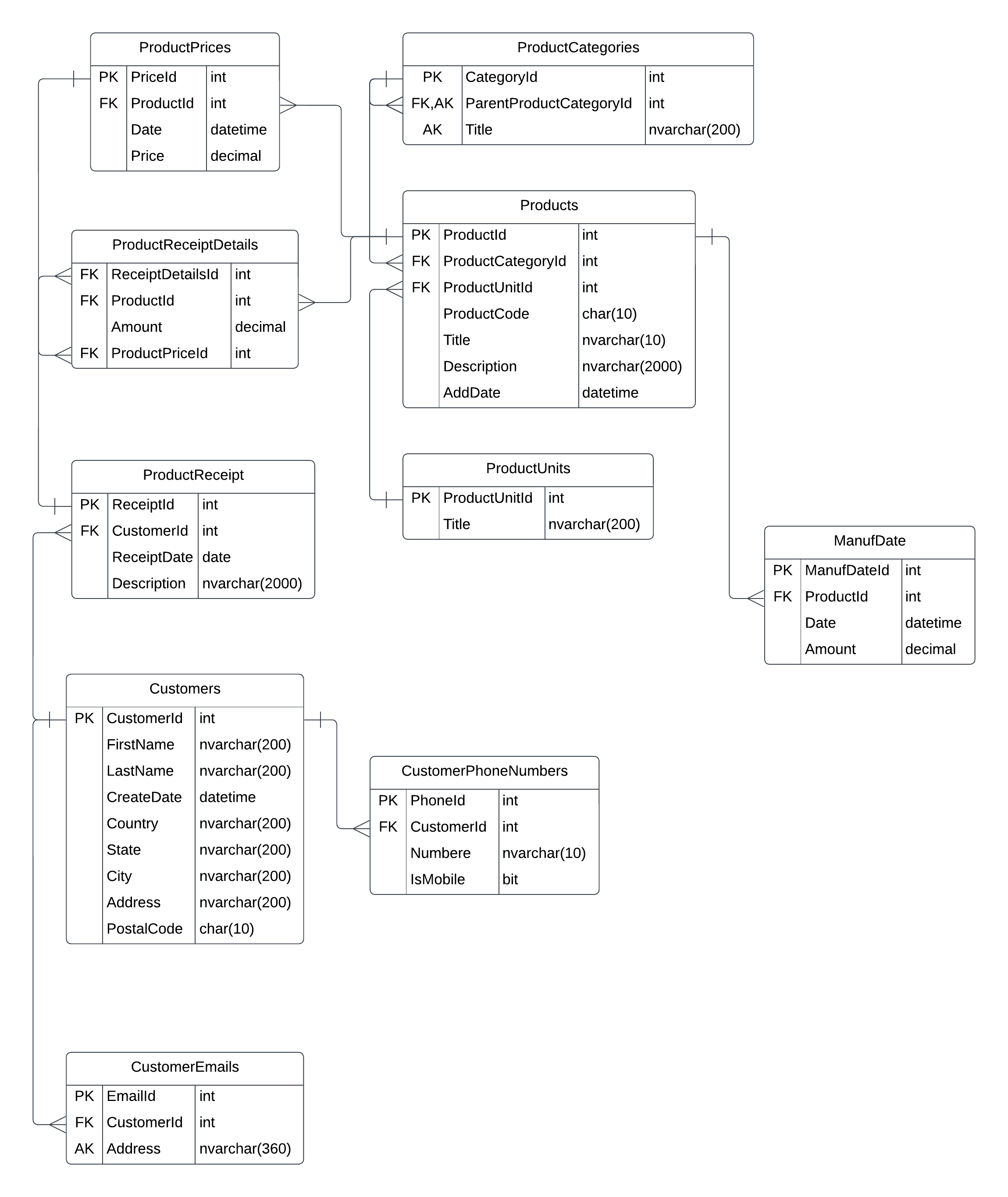


نمودار فعالیت یکی از انواع نمودارهای UML (Unified Modeling Language) است که برای نمایش جریان فعالیت‌ها و عملیات‌ها در یک سیستم استفاده می‌شود. این نمودار به ویژه برای نمایش فرایندها و فعالیت‌های یک سیستم، از زمان شروع تا پایان، مناسب است. در واقع، نمودار فعالیت به تصویر کشیدن جریان کار یک فرآیند و تبدیل‌های مختلف آن پرداخته و از آن برای مدل‌سازی فعالیت‌ها، وابستگی‌ها و تصمیم‌گیری‌ها استفاده می‌شود.

**مهمترین عناصر نمودار فعالیت عبارتند از:**

1. فعالیت (Activity): نشان‌دهنده عملیات یا فعالیت‌های انجام شده در یک مرحله مشخص از فرآیند.
2. گره شروع (Initial Node): نمایانگر آغاز فرآیند است و معمولاً با یک دایره کوچک نشان داده می‌شود.
3. گره پایان (Final Node): نمایانگر پایان فرآیند است و معمولاً با یک دایره دیگر یا یک مستطیل کوچک نمایش داده می‌شود.
4. تصمیم (Decision): برای نمایش تصمیم‌گیری‌ها در فرآیند و جریان انتخاب بین مسیرهای مختلف.
5. گره ادغام (Merge Node): برای ادغام مسیرهای مختلف جریان کار به یکدیگر.
6. شعاع جریان (Flow Edge): نشان‌دهنده جریان فعالیت از یک فعالیت به فعالیت دیگر است.

## Class Diagram

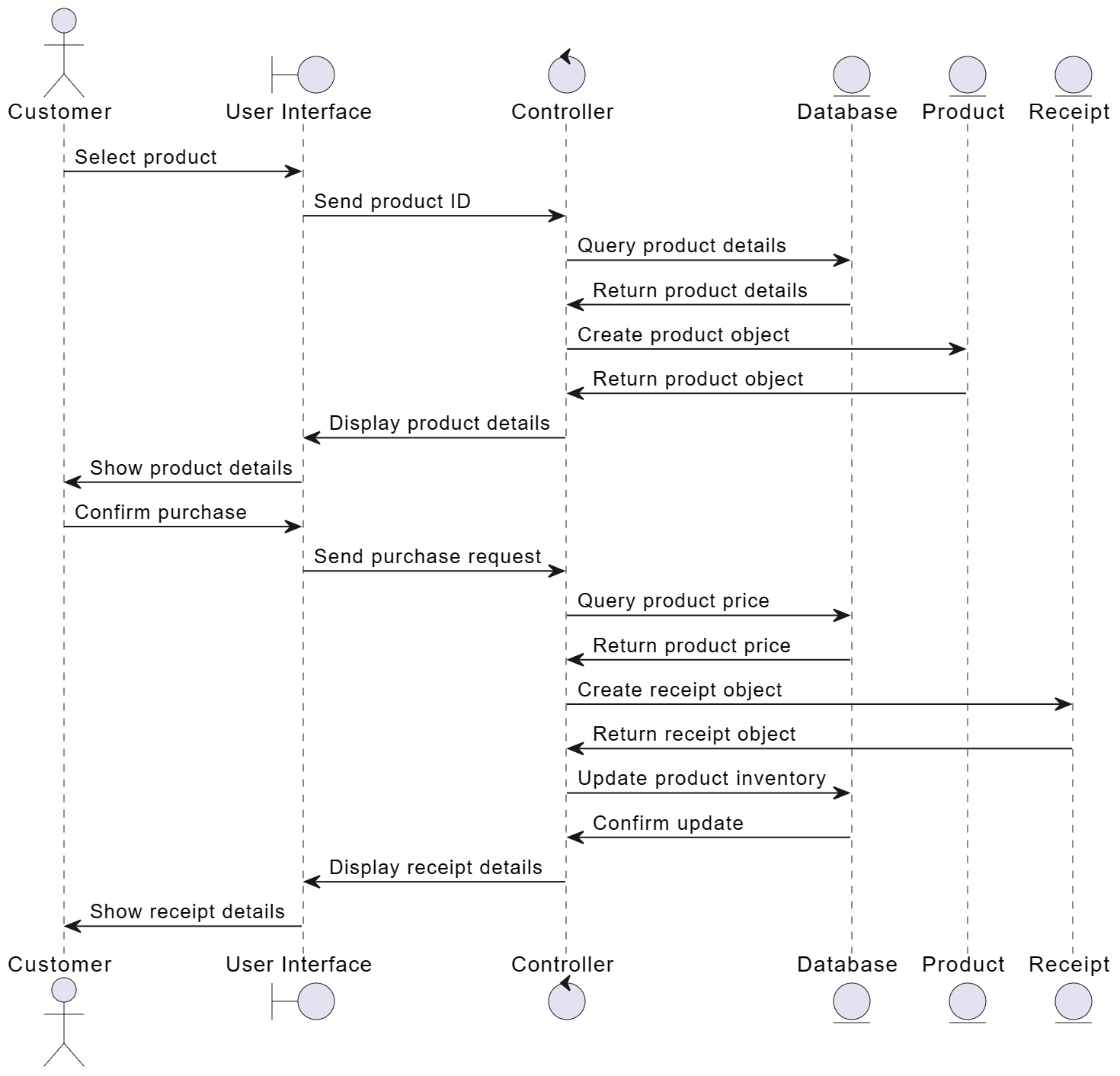


نمودار کلاس یکی دیگر از نمودارهای مهم UML (Unified Modeling Language) است که برای نمایش ساختارهای مختلف یک سیستم، اعم از کلاس‌ها، ارتباطات، و ویژگی‌ها استفاده می‌شود. این نمودار به تیم‌های تحلیل و طراحی سیستم کمک می‌کند تا به شکل گرافیکی و قابل فهم، اجزای اصلی ساختار سیستم را نمایش دهند.

**تعدادی از عناصر اصلی که در نمودار کلاس استفاده می‌شوند عبارتند از:**

1. کلاس (Class): نمایش یک نوع از اجسام یا دستگاه‌هایی که در سیستم وجود دارند. هر کلاس می‌تواند شامل ویژگی‌ها (متغیرها) و عملیات‌ها (توابع یا متدها) باشد.
2. ویژگی (Attribute): ویژگی‌ها یا متغیرها نشان‌دهنده ویژگی‌های کلاس هستند.
3. عملیات (Operation): عملیات‌ها یا متدها نشان‌دهنده عملیات‌هایی هستند که یک کلاس می‌تواند انجام دهد.
4. ارتباط (Association): نشان‌دهنده ارتباط بین دو یا چند کلاس است. این ارتباط ممکن است دارای جهت (دوطرفه یا یکطرفه) باشد.
5. وراثت (Inheritance): نمایش وابستگی بین کلاس‌ها، که یک کلاس (کلاس فرزند) ویژگی‌ها و عملیات‌های یک کلاس دیگر (کلاس والد) را به ارث می‌برد.
6. رابطه (Interface): نمایش روابط میان کلاس‌ها و یک رابط، که می‌تواند شامل تعدادی از عملیات یا ویژگی‌ها باشد.

## Sequence Diagram

این دیاگرام نشان می‌دهد که چگونه یک مشتری می‌تواند با استفاده از یک رابط کاربری یک فاکتور خرید از محصولات موجود در بانک اطلاعاتی دریافت کند. این دیاگرام شامل شش شیء است: مشتری، رابط کاربری، کنترل‌کننده، بانک اطلاعاتی، محصول و فاکتور. این دیاگرام همچنین شامل چندین پیام است که بین اشیاء رد و بدل می‌شوند. این دیاگرام را می‌توانید در بالا ببینید.

این دیاگرام Sequence نشان‌دهنده فرآیند ایجاد مشتری جدید است. CustomerController از CustomerRepository برای ذخیره اطلاعات مشتری استفاده می‌کند. سپس EmailService و PhoneService به ترتیب از EmailRepository و PhoneRepository برای ذخیره اطلاعات ایمیل و شماره تلفن استفاده می‌کنند.

****

# **بانک اطلاعات نرم افزار بر اساس ساختار T-Sql**

CREATE DATABASE [DoogheAbaliDB]

ON

(

NAME = N'DoogheAbaliDB',

FILENAME = N'C:\Data\DoogheAbaliDB.mdf',

SIZE = 5MB,

FILEGROWTH = 5MB

)

LOG ON

(

NAME = N'DoogheAbaliDB\_log',

FILENAME = N'C:\Data\DoogheAbaliDB\_log.mdf',

SIZE = 5MB,

FILEGROWTH = 5MB

)

GO

USE DoogheAbaliDB

GO

CREATE TABLE Customers (

[CustomerId] int identity NOT NULL PRIMARY KEY,

[FirstName] NVARCHAR(200) NOT NULL,

[LastName] NVARCHAR(200) NOT NULL,

[CreateDate] DATETIME NOT NULL DEFAULT(GETDATE()),

[Country] NVARCHAR(200) NOT NULL,

[State] NVARCHAR(200) NOT NULL,

[City] NVARCHAR(200) NOT NULL,

[Address] NVARCHAR(200) NOT NULL,

[PostalCode] CHAR(10) NULL

);

GO

CREATE TABLE [CustomerEmails] (

[EmailId] INT IDENTITY NOT NULL PRIMARY KEY,

[CustomerId] INT NOT NULL REFERENCES[Customers]([CustomerId]),

[Address] NVARCHAR(360) NOT NULL,

UNIQUE([Address])

);

GO

CREATE TABLE [CustomerPhoneNumbers] (

[PhoneId] INT IDENTITY NOT NULL PRIMARY KEY,

[CustomerId] INT NOT NULL REFERENCES[Customers]([CustomerId]),

[Numbere] NVARCHAR(10) NOT NULL,

[IsMobile] BIT NOT NULL DEFAULT(0)

);

GO

CREATE TABLE [ProductCategories] (

[CategoryId] INT IDENTITY NOT NULL PRIMARY KEY,

[ParentProductCategoryId] INT NULL REFERENCES[ProductCategories]([CategoryId]),

[Title] NVARCHAR(200) NOT NULL,

UNIQUE([ParentProductCategoryId],[Title])

);

GO

CREATE TABLE [ProductUnits] (

[ProductUnitId] INT IDENTITY NOT NULL PRIMARY KEY,

[Title] NVARCHAR(200) NOT NULL,

);

GO

CREATE TABLE [Products] (

[ProductId] INT IDENTITY NOT NULL PRIMARY KEY,

[ProductCategoryId] INT NOT NULL REFERENCES[ProductCategories]([CategoryId]),

[ProductUnitId] INT NOT NULL REFERENCES[ProductUnits]([ProductUnitId]),

[ProductCode] CHAR(10),

[Title] NVARCHAR(10) NOT NULL,

[Description] NVARCHAR(2000) NOT NULL,

[AddDate] DATETIME NOT NULL DEFAULT(GETDATE())

);

GO

CREATE TABLE [ProductPrices] (

[PriceId] INT IDENTITY NOT NULL PRIMARY KEY,

[ProductId] INT NOT NULL REFERENCES[Products]([ProductId]),

[Date] DATETIME NOT NULL DEFAULT(GETDATE()),

[Price] DECIMAL NOT NULL

);

GO

CREATE TABLE [ManufDate] (

[ManufDateId] INT IDENTITY NOT NULL PRIMARY KEY,

[ProductId] INT NOT NULL REFERENCES[Products]([ProductId]),

[Date] DATETIME NOT NULL DEFAULT(GETDATE()),

[Amount] DECIMAL NOT NULL

CONSTRAINT CHK\_ManufDate\_Amount CHECK([Amount] < 0)

);

GO

CREATE TABLE [ProductReceipt] (

[ReceiptId] INT IDENTITY NOT NULL PRIMARY KEY,

[CustomerId] INT NOT NULL REFERENCES[Customers]([CustomerId]),

[ReceiptDate] DATE NOT NULL DEFAULT(GETDATE()),

[Description] NVARCHAR(2000) NOT NULL,

);

GO

CREATE TABLE [ProductReceiptDetails] (

[ReceiptDetailsId] INT NOT NULL REFERENCES[ProductReceipt]([ReceiptId]),

[ProductId] INT NOT NULL REFERENCES[Products]([ProductId]),

[Amount] DECIMAL NOT NULL,

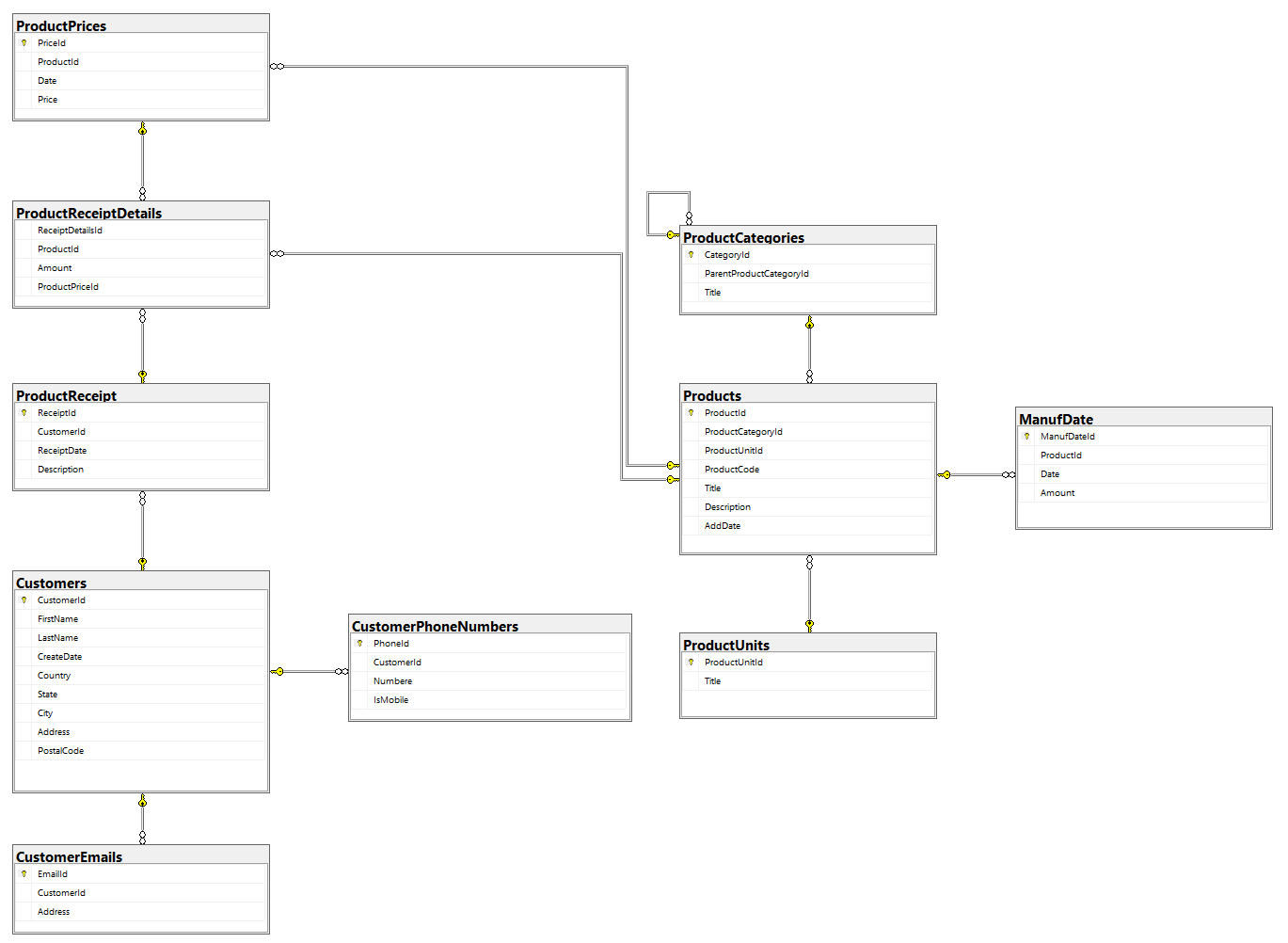
[ProductPriceId] INT NOT NULL REFERENCES[ProductPrices]([PriceId]),

);

GO

[لینک دانلود T-Sql دیتابیس](https://drive.google.com/drive/folders/1Lq5YtbiQg9S01Ycefd_dKjr-Y8ufGUwM?usp=sharing)

# **Database Diagram**



**تحلیل دیاگرام:**

**این دیتابیس برای یک سیستم مدیریت محصولات و مشتریان (Product and Customer Management System) طراحی شده است. در زیر توضیحاتی در مورد کارکرد اجزای این دیتابیس آورده شده است:**

1. **Customers (مشتریان):**
   1. اطلاعات مشتریان نظیر نام، نام خانوادگی، تاریخ ایجاد حساب، کشور، استان، شهر، آدرس، و کد پستی در جدول `Customers` ذخیره می‌شوند.
   2. هر مشتری دارای یک شناسه یکتا (`CustomerId`) است.
2. **CustomerEmails (ایمیل‌های مشتریان):**
   1. اطلاعات ایمیل‌های مشتریان در جدول `CustomerEmails` ذخیره می‌شوند.
   2. از کلید خارجی به `CustomerId` اشاره شده است تا ارتباط میان ایمیل و مشتری برقرار شود.
3. **CustomerPhoneNumbers (شماره‌های تلفن مشتریان):**
   1. اطلاعات شماره‌های تلفن مشتریان در جدول `CustomerPhoneNumbers` ذخیره می‌شوند.
   2. از کلید خارجی به `CustomerId` اشاره شده است تا ارتباط میان شماره تلفن و مشتری برقرار شود.
4. **ProductCategories (دسته‌بندی محصولات):**
   1. دسته‌بندی‌های مختلف محصولات در جدول `ProductCategories` ذخیره می‌شوند.
   2. از کلید خارجی به `ParentProductCategoryId` اشاره شده است تا ارتباط بین دسته‌بندی‌های مختلف محصولات برقرار شود.
5. **ProductUnits (واحدهای محصولات):**
   1. اطلاعات واحدهای محصولات در جدول `ProductUnits` ذخیره می‌شوند.
6. **Products (محصولات):**
   1. اطلاعات محصولات نظیر دسته‌بندی، واحد، کد محصول، عنوان، توضیحات، و تاریخ افزودن در جدول `Products` ذخیره می‌شوند.
   2. از کلید خارجی به `ProductCategoryId` و `ProductUnitId` اشاره شده است تا ارتباط میان محصولات و دسته‌بندی‌ها و واحدها برقرار شود.
7. **ProductPrices (قیمت‌های محصولات):**
   1. اطلاعات قیمت محصولات در جدول `ProductPrices` ذخیره می‌شوند.
   2. از کلید خارجی به `ProductId` اشاره شده است تا ارتباط میان قیمت‌ها و محصولات برقرار شود.
8. **ManufDate (تاریخ تولید محصولات):**
   1. اطلاعات مربوط به تاریخ تولید محصولات در جدول `ManufDate` ذخیره می‌شوند.
   2. یک قید (CHECK CONSTRAINT) برای بررسی محدودیت‌های خاص در تاریخ تولید اعمال شده است.
9. **ProductReceipt و ProductReceiptDetails (رسید و جزئیات رسید محصول):**
   1. اطلاعات مربوط به رسید محصولات در جدول `ProductReceipt` ذخیره می‌شوند.
   2. جزئیات هر رسید در جدول `ProductReceiptDetails` ذخیره می‌شوند.

**استفاده از این دیتابیس به شما این امکان را می‌دهد که اطلاعات مشتریان، محصولات، و تاریخچه محصولات را به طور سازمان‌یافته ذخیره کنید و از آنها در تحلیل‌ها و گزارش‌های مختلف برای مدیریت و برنامه‌ریزی استفاده نمایید.**

****

# **نسخه اولیه نرم افزار**

🧑🏽‍💻

## **اطلاعات کلی نرم افزار:**

|  |  |  |
| --- | --- | --- |
| کد رنگ‌های مورد استفاده | فریمورک |  |
|  | WPF |  |

## صفحه‌ی ورود به نرم افزار (Login page)



**نکات امنیتی صفحه لاگین:**

فیلد ورود رمز از نوع PasswordBox و با محدودیت کاراکتر (۶ کاراکتر) در نظر گرفته شده تا محتوای این فیلد نمایش داد نشود و کاربر نتواند تعداد کاراکتر زیادتر از استانداردِ این نرم افزار وارد نماید. کاربر هنگام ورود می‌باید رمز عبور خود را وارد نماید. این کار برای ثبت گزارش کاربرانِ نرم افزار و اطلاعات ثبت شده توسط این فرد مورد استفاده قرار می‌گیرد.

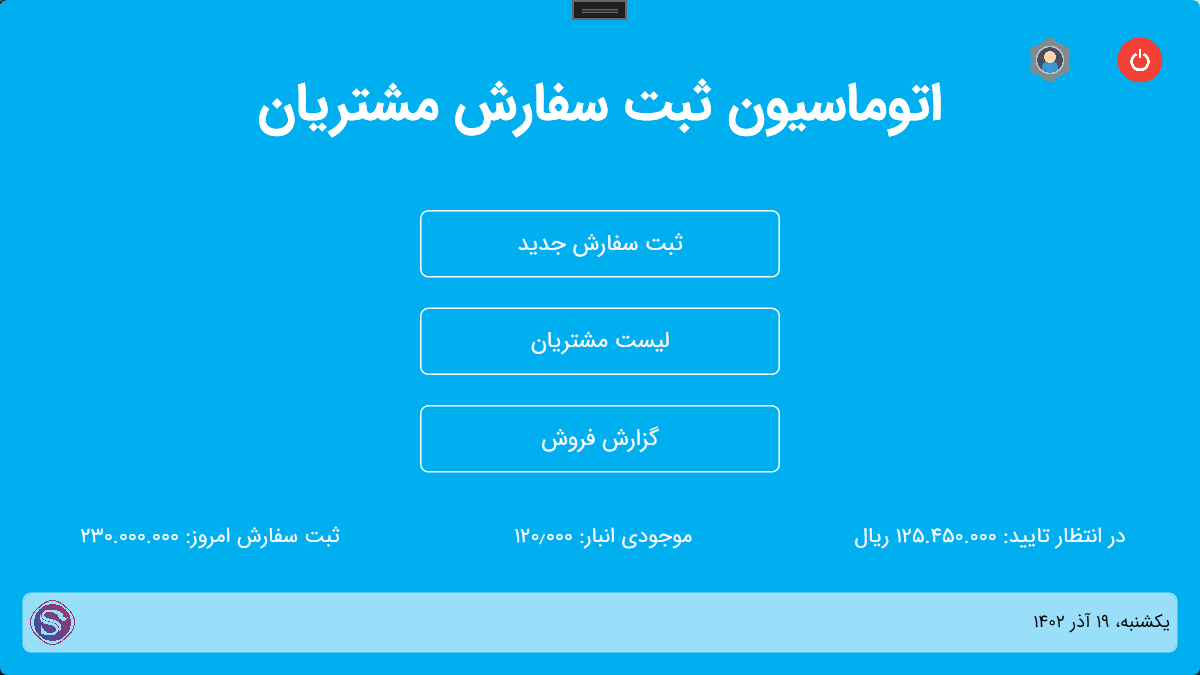
میزان تلاش برای ورود رمز ۳ بار می‌باشد. بعد از آن نرم افزار به مدت ۱۰ دقیقه قفل خواهد شد و بعد از این زمان دوباره می‌توان مجددا تا ۳ بار تلاش برای ورود رمز انجام داد. این امکان برای ایجاد امنیت و عدم استفاده افراد غیر کاربر ایجاد شده است. در آپدیت‌های بعدی فیلد نام کاربر (User name) جهت گزارش گیری از عملکرد کاربران به این صفحه اضافه خواهد شد.

نکته: چنانچه رمز وارد شده اشتباه باشد، پیغام رمز ورود اشتباه است برای کاربر در زیر دکمه ورود چاپ خواهد شد.

**نکات طراحی بصری صفحه لاگین:**

در طراحی رابط کاربری این صفحه تلاش شده تا نهایت تکنیک مینیمالیسم جهت استفاده راحت کاربر پیاده سازی شود. همچنین سعی شده تا لوگوی شرکت به طور محسوسی در این صفحه چشم نوازی کند.

## صفحه اصلی نرم افزار (Main page)



**نکات کلی**

در طراحی این صفحه نیز تلاش شده تا نهایت سادگی جهت استفاده راحت کاربر فراهم شود. در قسمت بالای صفحه (Title row) نام نرم افزار و همچنین کلیدهای خروج از نرم افزار و کلید ورود به صفحه تنظیمات نرم افزار تعبیه شده. جهت اطمینان از خروج از نرم افزار بعد از فشردن این دکمه یک پیغام اطمینان از خروج برای کاربر نمایش داده می‌شود و کاربر بعد از زدن دکمه Yes از نرم افزار به طور کامل خارج می‌شود.

دکمه تنظیمات برای مشاهده گزارشات کاربران، ایجاد کاربر جدید، تغییر رمز ورود برای کاربران و تنظیم فونت‌های گزارشات ایجاد شده است.

در قسمت وسط صفحه (Body row) سه دکمه تعبیه شده که کاربر می‌تواند با فشردن هر کدام از آن‌ها به پنجره مربوطه هدایت شود. کاربر برای ثبت سفارش جدید باید بر روی دکمه آن کلیک کرده تا به صفحه مربوطه هدایت شود.

همچنین یک صفحه برای مدیریت مشتریان شرکت در نظر گرفته شده و کاربر می‌تواند از این صفحه برای ایجاد و ویرایش مشتریان استفاده کند. اطلاعات این صفحه در قسمت ثبت سفارش کاربرد دارد.

همچنین در این قسمت یک گزارش کلی به اختصار از وضعیت موجودی کالا در انبار، سفارشات ثبت شده روز جاری و همچنین سفارشات در انتظار تایید نمایش داده می‌شود تا کاربر در هر زمان آگاهی لازم از سفارشات را داشته باشد.

در قسمت پایین صفحه (Toolbar row) تاریخ جاری و همچنین لوگوی طراح نرم افزار گذاشته شده. کاربر با کلیک بر روی این لوگو به وبسایت طراح نرم افزار هدایت خواهد شد.