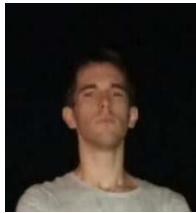




Federico García – 242015

Grupo: M2B

Docente: Luis Dentone

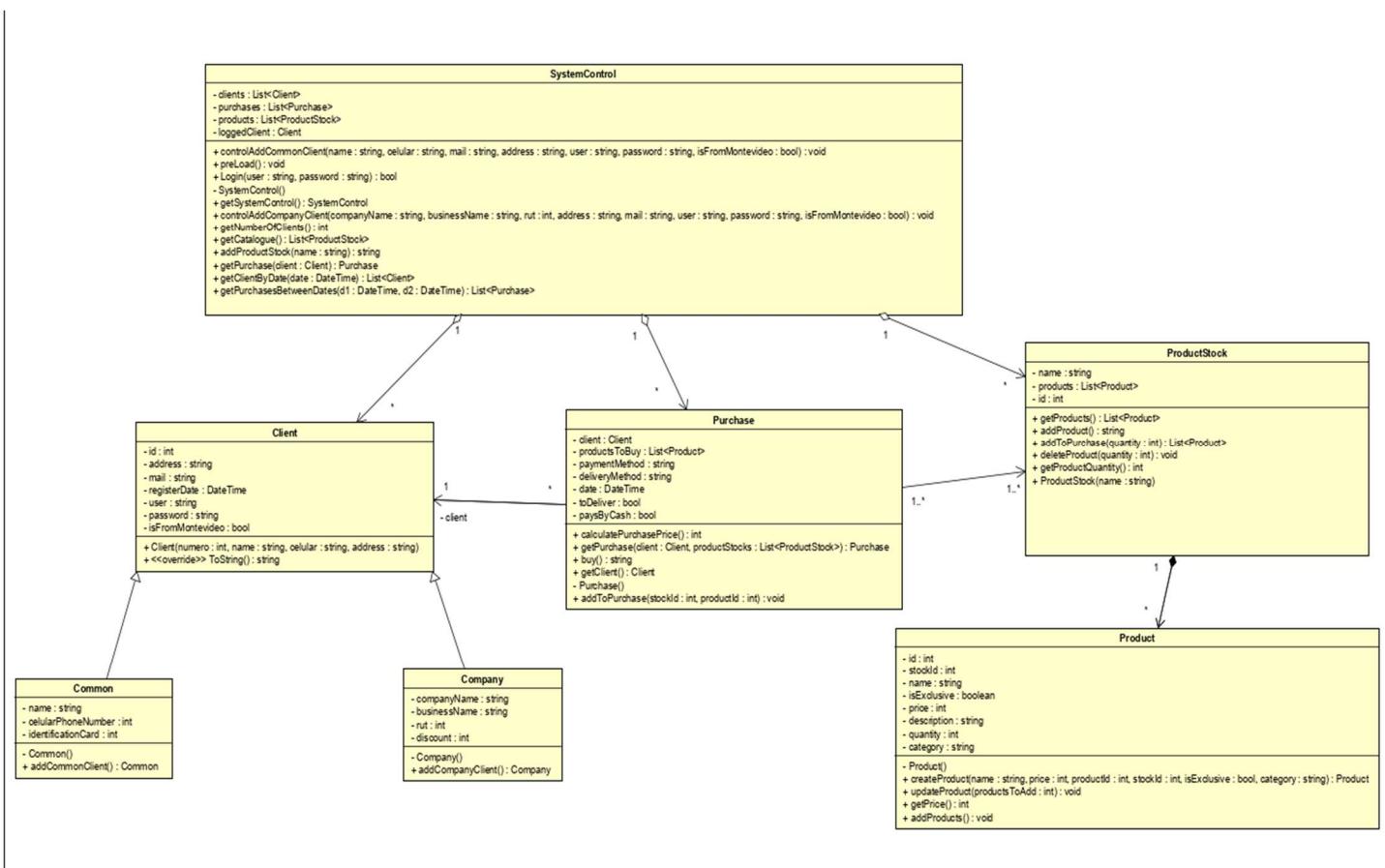


Andrés Valle Lisboa – 246109

Grupo: M2B

Docente: Luis Dentone

UML



Program

```
1  using System;
2  using System.Collections.Generic;
3  using ShopSystem;
4
5  namespace ConsoleApp
6  {
7      class Program
8      {
9          private static bool exit = false;
10         private static SystemControl _system; //Variable que contiene la dirección de memoria de la instancia de la clase administradora
11         static void Main(string[] args)
12         {
13             _system = SystemControl.getSystemControl(); //Variable de acceso a clase administradora(contains la instancia de la clase administradora)
14
15             _system.preLoad(); //Ejecuta la precarga
16
17             while (!exit)
18             {
19                 bool optionNotCorrect = true;
20                 while (optionNotCorrect)
21                 {
22                     Console.Clear();
23                     Console.WriteLine("Presione el número de opción en su teclado numérico para realizar la operación correspondiente");
24                     Console.WriteLine("-----");
25                     Console.WriteLine("1 - Iniciar Sesión");
26                     Console.WriteLine("2 - Registrarse");
27                     Console.WriteLine("3 - Sistema"); //Alta de productos en el catálogo va acá.
28                     Console.WriteLine("0 - Salir");
29                     Console.WriteLine("-----");
30                     ConsoleKey key = Console.ReadKey(true).Key;
31                     switch (key)
32                     {
33                         case ConsoleKey.NumPad1:
34                         case ConsoleKey.D1:
35                             Console.Clear();
36                             signIn();
37                             break;
38                         case ConsoleKey.NumPad2:
39                         case ConsoleKey.D2:
30                             Console.Clear();
31                             register();
32                             break;
33
34                         case ConsoleKey.NumPad3:
35                         case ConsoleKey.D3:
36                             Console.Clear();
37                             system();
38                             break;
39
40                         case ConsoleKey.NumPad0:
41                         case ConsoleKey.D0:
42                             Console.Clear();
43                             optionNotCorrect = false;
44                             exit = true;
45                             break;
46
47                     }
48
49                 }
50
51             }
52
53         }
54
55     }
56
57 }
58
59     private static void signIn()
60     {
61         try
62         {
63             bool signInSuccessful = false;
64             while (!signInSuccessful)
65             {
66                 Console.Clear();
67                 Console.WriteLine("Seleccione el tipo de cliente");
68                 Console.WriteLine("1 - Cliente común");
69                 Console.WriteLine("2 - Cliente empresa");
70                 ConsoleKey key = Console.ReadKey(true).Key;
71                 if (key == ConsoleKey.NumPad1 || key == ConsoleKey.D1)
72                 {
73                     Console.Clear();
74                     Console.WriteLine("Ingrese su nombre");
75                     string name = Console.ReadLine();
76                     Console.WriteLine("Ingrese su número de cédula");
77                     int identificationCard = Convert.ToInt32(Console.ReadLine());
78                     Console.WriteLine("Ingrese su celular");
79                     string celular = Console.ReadLine();
80                     Console.WriteLine("Ingrese su dirección");
81                     string address = Console.ReadLine();
82                     Console.WriteLine("Ingrese su mail");
83                     string mail = Console.ReadLine();
84
85                     signInSuccessful = true;
86
87                 }
88             }
89         }
90     }
91
92 }
```

```

84     Console.WriteLine("Ingrese su usuario");
85     string user = Console.ReadLine();
86     Console.WriteLine("Ingrese su contraseña");
87     string password = Console.ReadLine();
88     Console.WriteLine("Ingrese su Departamento");
89     string departamento = Console.ReadLine();
90     bool isFromMontevideo = false;
91     if (departamento == "Montevideo" || departamento == "montevideo" || departamento == "MONTEVIDEO")
92     {
93         isFromMontevideo = true;
94     }
95     var result = _system.addCommonClient(name, identificationCard, celular, mail, address, user, password, isFromMontevideo);
96     registerSuccessful = result.wasRegisterSuccessful;
97     string message = result.message;
98     Console.Clear();
99 }
100
101 else if (key == ConsoleKey.NumPad2 || key == ConsoleKey.D2) //EJEMPLO DE MENU
102 {
103     Console.Clear();
104     Console.WriteLine("Ingrese el nombre de su empresa");
105     string companyName = Console.ReadLine();
106     Console.WriteLine("Ingrese su razón social");
107     string businessName = Console.ReadLine();
108     Console.WriteLine("Ingrese su rut");
109     int rut = Convert.ToInt32(Console.ReadLine());
110     Console.WriteLine("Ingrese su dirección");
111     string address = Console.ReadLine();
112     Console.WriteLine("Ingrese su mail");
113     string mail = Console.ReadLine();
114     Console.WriteLine("Ingrese su número telefónico");
115     string phone = Console.ReadLine();
116     Console.WriteLine("Ingrese su usuario");
117     string user = Console.ReadLine();
118     Console.WriteLine("Ingrese su contraseña");
119     string password = Console.ReadLine();
120     Console.WriteLine("Ingrese su Departamento");
121     string departamento = Console.ReadLine();
122     bool isFromMontevideo = false;
123     Console.WriteLine("Ingrese el descuento de su empresa");
124     int discount = Convert.ToInt32(Console.ReadLine());
125     if (departamento == "Montevideo" || departamento == "montevideo" || departamento == "MONTEVIDEO")

```

```

126     {
127         isFromMontevideo = true;
128     }
129     var result = _system.addCompanyClient(companyName, businessName, rut, address, mail, phone, user, password, isFromMontevideo, discount);
130     registerSuccessful = result.wasRegisterSuccessful;
131     string message = result.message;
132     Console.Clear();
133 }
134 else
135 {
136     continue;
137 }
138 if (registerSuccessful)
139 {
140     Console.WriteLine("El usuario fue registrado correctamente");
141     Console.WriteLine("Presione una tecla para continuar");
142     ConsoleKey _key = Console.ReadKey(true).Key;
143 }
144 else
145 {
146     Console.WriteLine("Los datos no son correctos");
147     Console.WriteLine("Presione la tecla 'V' para volver al menú principal o cualquier otra tecla para volver a intentarlo");
148     ConsoleKey _key = Console.ReadKey(true).Key;
149     if (_key == ConsoleKey.V)
150     {
151         registerSuccessful = true;
152     }
153 }
154 }
155 }
156 catch(Exception err)
157 {
158     errorHandling(err);
159 }
160 }
161 private static void signIn()
162 {
163     try
164     {
165         bool wasLoginSuccessful = false;
166         while (!wasLoginSuccessful)

```

```

167     {
168         Console.WriteLine("Ingrese su usuario");
169         string user = Console.ReadLine();
170         Console.WriteLine("Ingrese su contraseña");
171         string password = Console.ReadLine();
172         wasLoginSuccessful = _system.login(user, password);
173         Console.Clear();
174         if (wasLoginSuccessful)
175         {
176             Console.WriteLine("Los datos son correctos, la sesión ha sido iniciada \n \n Presione una tecla para continuar");
177             Console.ReadKey();
178             Console.Clear();
179             clientMenu();
180         }
181     }
182     else
183     {
184         Console.WriteLine("El usuario o la contraseña no fueron correctos");
185         Console.WriteLine("Presione una tecla para volver a intentarlo o presione \"v\" para volver al menú principal");
186         ConsoleKey _key = Console.ReadKey(true).Key;
187         if (_key == ConsoleKey.V)
188         {
189             wasLoginSuccessful = true;
190         }
191     }
192 }
193 catch (Exception err)
194 {
195     errorHandling(err);
196 }
197 }

private static void system()
{
    bool optionIsNotCorrect = true;
    while (optionIsNotCorrect)
    {
        bool _optionNotCorrect = true;
        while (_optionNotCorrect)
        {
            Console.Clear();
            Console.WriteLine("1 - Agregar productos");

209
210     Console.WriteLine("2 - Ver clientes registrados");
211     Console.WriteLine("3 - Compras realizadas");
212     Console.WriteLine("0 - Salir");

213     ConsoleKey _key = Console.ReadKey(true).Key; //Obtiene la tecla presionada por el usuario y la almacena dentro de la variable _key
214     switch (_key)
215     {
216         case ConsoleKey.NumPad1:
217         case ConsoleKey.D1:
218             Console.Clear();
219             _optionNotCorrect = false;
220             AddProducts();
221             break;
222         case ConsoleKey.NumPad2:
223         case ConsoleKey.D2:
224             Console.Clear();
225             _optionNotCorrect = false;
226             DisplayRegisteredClients();
227             break;
228         case ConsoleKey.NumPad3:
229         case ConsoleKey.D3:
230             printPurchasesList();
231             break;
232         case ConsoleKey.NumPad0:
233         case ConsoleKey.D0:
234             Console.Clear();
235             _optionNotCorrect = false;
236             optionIsNotCorrect = false;
237             break;
238     }
239 }
240 }
241 }
242 }
243 private static void AddProducts()
244 {
245     Console.Clear();
246     Console.WriteLine("Ingrese el nombre de su producto");
247     string productName = Console.ReadLine();
248 }
249
250 private static void DisplayRegisteredClients()

```

```

251     {
252         bool wasDateEnteredCorrect = false;
253         while (!wasDateEnteredCorrect)
254         {
255             Console.WriteLine("Ingrese la fecha de límite de registro");
256             string dateToParse = Console.ReadLine();
257             DateTime date;
258             if (DateTime.TryParse(dateToParse, out date))
259             {
260                 Console.Clear();
261                 Console.WriteLine("-----");
262                 Console.WriteLine("Nombre | Mail | Tipo de cliente");
263                 Console.WriteLine("-----");
264                 wasDateEnteredCorrect = true;
265                 var clients = _system.getClientsByDate(date);
266                 if (clients.Count > 0)
267                 {
268                     foreach (Client c in clients)
269                     {
270                         Console.WriteLine("-----");
271                         if (c.GetType() == typeof(Common))
272                         {
273                             Console.WriteLine("|" + ((Common)c).Name + " | " + c.Mail + " | Common |");
274                         }
275                         if (c.GetType() == typeof(Company))
276                         {
277                             Console.WriteLine("|" + ((Company)c).CompanyName + " | " + c.Mail + " | Company |");
278                         }
279                         Console.WriteLine("-----");
280                     }
281                 }
282                 else Console.WriteLine("No hay clientes registrados antes de la fecha");
283             }
284             else Console.WriteLine("Ingrese una fecha válida");
285         }
286         Console.ReadKey();
287     }
288
289     private static void clientMenu()
290     {
291         bool islogged = true;
292         while (islogged)
293     {
294         Console.Clear();
295         Console.WriteLine("Presione el número de opción en su teclado numérico para realizar la operación correspondiente");
296         Console.WriteLine("-----");
297         Console.WriteLine("1 - Ver catálogo de productos");
298         Console.WriteLine("2 - Configuraciones");
299         Console.WriteLine("0 - Salir");
300         Console.WriteLine("-----");
301
302         ConsoleKey _key = Console.ReadKey(true).Key; //Guarda la tecla presionada
303         switch (_key)
304         {
305             case ConsoleKey.NumPad1:
306             case ConsoleKey.D1:
307                 try
308                 {
309                     var purchase = _system.getPurchase();
310                     bool hasPurchaseEnded = false;
311                     while (!hasPurchaseEnded)
312                     {
313                         Console.Clear();
314                         Console.WriteLine("Seleccione la categoría");
315                         Console.WriteLine("0 - Todas");
316                         Console.WriteLine("1 - Frescos");
317                         Console.WriteLine("2 - Congelados");
318                         Console.WriteLine("3 - Hogar");
319                         Console.WriteLine("4 - Textiles");
320                         Console.WriteLine("5 - Tecnología");
321                         ConsoleKey k = Console.ReadKey(true).Key;
322
323
324                         Console.Clear();
325                         List<ProductStock> productStocks = _system.getCatalogue();
326                         string line = "-----";
327                         Console.WriteLine(line);
328                         Console.WriteLine("Nombre | Precio | Cantidad | StockId | ProductId|");
329                         Console.WriteLine(line);
330                         switch (k)
331                         {
332                             case ConsoleKey.NumPad1:
333                             case ConsoleKey.D1:

```

```

335     foreach (ProductStock _productStock in productStocks)
336     {
337         if (_productStock.Name == "Frescos")
338         {
339             List<Product> _products = _productStock.Products;
340             foreach (Product p in _products)
341             {
342                 Console.WriteLine(line);
343                 Console.WriteLine(" | " + p.Name + " | " + p.Price + " | " + p.Quantity + " | " + p.StockId + " | " + p.Id);
344                 Console.WriteLine(line);
345             }
346         }
347     }
348     break;
349     case ConsoleKey.NumPad2:
350     case ConsoleKey.D2:
351     foreach (ProductStock _productStock in productStocks)
352     {
353         if (_productStock.Name == "Congelados")
354         {
355             List<Product> _products = _productStock.Products;
356             foreach (Product p in _products)
357             {
358                 Console.WriteLine(line);
359                 Console.WriteLine(" | " + p.Name + " | " + p.Price + " | " + p.Quantity + " | " + p.StockId + " | " + p.Id);
360                 Console.WriteLine(line);
361             }
362         }
363     }
364     break;
365     case ConsoleKey.NumPad3:
366     case ConsoleKey.D3:
367     foreach (ProductStock _productStock in productStocks)
368     {
369         if (_productStock.Name == "Hogar")
370         {
371             List<Product> _products = _productStock.Products;
372             foreach (Product p in _products)
373             {
374                 Console.WriteLine(line);
375                 Console.WriteLine(" | " + p.Name + " | " + p.Price + " | " + p.Quantity + " | " + p.StockId + " | " + p.Id);
376                 Console.WriteLine(line);
377             }
378         }
379     }
380     break;
381     case ConsoleKey.NumPad4:
382     case ConsoleKey.D4:
383     foreach (ProductStock _productStock in productStocks)
384     {
385         if (_productStock.Name == "Textiles")
386     }

```

```

386     {
387         List<Product> _products = _productStock.Products;
388         foreach (Product p in _products)
389         {
390             Console.WriteLine(line);
391             Console.WriteLine(" | " + p.Name + " | " + p.Price + " | " + p.Quantity + " | " + p.StockId + " | " + p.Id);
392             Console.WriteLine(line);
393         }
394     }
395     break;
396     case ConsoleKey.NumPad5:
397     case ConsoleKey.D5:
398     foreach (ProductStock _productStock in productStocks)
399     {
400         if (_productStock.Name == "Tecnologia")
401         {
402             List<Product> _products = _productStock.Products;
403             foreach (Product p in _products)
404             {
405                 Console.WriteLine(line);
406                 Console.WriteLine(" | " + p.Name + " | " + p.Price + " | " + p.Quantity + " | " + p.StockId + " | " + p.Id);
407                 Console.WriteLine(line);
408             }
409         }
410     }
411     break;
412     default:
413     foreach (ProductStock _productStock in productStocks)
414     {
415         List<Product> _products = _productStock.Products;
416         foreach (Product p in _products)
417         {
418             Console.WriteLine(line);
419             Console.WriteLine(" | " + p.Name + " | " + p.Price + " | " + p.Quantity + " | " + p.StockId + " | " + p.Id);
420             Console.WriteLine(line);
421         }
422     }
423     break;
424
425     Console.WriteLine("presione \"a\" para agregar productos al carrito");
426     Console.WriteLine("presione \"v\" para volver");
427     optionCorrect = false;
428     while (!optionCorrect)
429     {
430         ConsoleKey o = Console.ReadKey(true).Key;
431         if (o == ConsoleKey.A)
432         {
433             optionCorrect = true;
434             Console.WriteLine("Ingrese el id del stock del producto que desea agregar");
435             int stockId = Convert.ToInt32(Console.ReadLine());
436         }

```

```

437     Console.WriteLine("Ingrese el Id del producto que desea agregar");
438     int productId = Convert.ToInt32(Console.ReadLine());
439     Console.WriteLine("Ingrese la cantidad de productos que desea agregar");
440     int quantity = Convert.ToInt32(Console.ReadLine());
441     string message = purchase.addToPurchase(stockId, productId, quantity);
442     Console.Clear();
443     Console.WriteLine(message);
444     Console.WriteLine("Precione \'c\' para finalizar la compra o cualquier otra tecla para continuar comprando");
445     ConsoleKey _k = Console.ReadKey(true).Key;
446     if (_k == ConsoleKey.C)
447     {
448         bool _optionCorrect = false; bool paysByCash = false; bool toDeliver = false;
449         while (!_optionCorrect)
450         {
451             Console.Clear();
452             Console.WriteLine("Elija su forma de pago");
453             Console.WriteLine("1 - Efectivo");
454             Console.WriteLine("2 - Tarjeta");
455             ConsoleKey _k = Console.ReadKey(true).Key;
456             switch (_k)
457             {
458                 case ConsoleKey.NumPad1:
459                 case ConsoleKey.D1:
460                     paysByCash = true;
461                     _optionCorrect = true;
462                     break;
463                 case ConsoleKey.NumPad2:
464                 case ConsoleKey.D2:
465                     paysByCash = false;
466                     _optionCorrect = true;
467                     break;
468             }
469             _optionCorrect = false;
470             while (!_optionCorrect)
471             {
472                 Console.Clear();
473                 Console.WriteLine("Especifique si pasará a retirar el producto o quiere que se lo envíen");
474                 Console.WriteLine("1 - Retiro");
475                 Console.WriteLine("2 - Envío");
476                 ConsoleKey key = Console.ReadKey(true).Key;
477                 switch (key)
478                 {
479                     case ConsoleKey.NumPad1:
480                     case ConsoleKey.D1:
481                         toDeliver = false;
482                         _optionCorrect = true;
483                         break;
484                     case ConsoleKey.NumPad2:
485                     case ConsoleKey.D2:
486                         toDeliver = true;
487                         _optionCorrect = true;
488                         break;
489                     case ConsoleKey.NumPad2:
490                     case ConsoleKey.D2:
491                         toDeliver = true;
492                         _optionCorrect = true;
493                         break;
494                 }
495             }
496             purchase.PaysByCash = paysByCash;
497             purchase.ToDeliver = toDeliver;
498             message = purchase.buy();
499             Console.WriteLine(message);
500             Console.WriteLine("Presione una tecla para continuar");
501             Console.ReadKey();
502             hasPurchaseEnded = true;
503         }
504         if (!hasPurchaseEnded)
505         {
506             if (hasPurchaseEnded)
507             {
508                 optionCorrect = true;
509             }
510         }
511     }
512 }
513 catch (Exception err)
514 {
515     errorHandling(err);
516 }
517 break;
518 case ConsoleKey.NumPad2:
519 case ConsoleKey.D2:
520     settings();
521     break;
522 case ConsoleKey.NumPad0:
523 case ConsoleKey.D0:
524     isLoggedIn = false;
525     break;
526 }
527 }
528 }
529 }
530 }
531 private static void printPurchasesList()
532 {
533     bool datesWereCorrect = false;
534     while (!datesWereCorrect)
535     {
536         Console.Clear();
537         Console.WriteLine("Ingrese la fecha inicial");

```

```

538     string d1ToParse = Console.ReadLine();
539     DateTime d1;
540     Console.WriteLine("Ingrese la fecha final");
541     string d2ToParse = Console.ReadLine();
542     DateTime d2;
543     Console.Clear();
544     if (DateTime.TryParse(d1ToParse, out d1) && DateTime.TryParse(d2ToParse, out d2))
545     {
546         var purchases = _system.getPurchasesBetweenDates(d1, d2);
547         Console.WriteLine("| Nombre de cliente | fecha de compra | cantidad total de productos |");
548         if (purchases.Count == 0) Console.WriteLine("No hay compras entre las fechas indicadas");
549         else
550         {
551             foreach (Purchase p in purchases)
552             {
553                 if (p.Client.GetType() == typeof(Common))
554                 {
555                     Console.WriteLine("| " + ((Common)p.Client).Name + " | " + p.Date + " | " + p.ProductsQuantity + " |");
556                 if (p.Client.GetType() == typeof(Company))
557                 {
558                     Console.WriteLine("| " + ((Company)p.Client).CompanyName + " | " + p.Date + " | " + p.ProductsQuantity + " |");
559                 }
560             }
561         }
562     }
563     else
564     {
565         Console.WriteLine("Ingrese fechas correctas");
566     }
567     Console.WriteLine("Presione \\v\\ para volver al menú o presione cualquier tecla para volver a ingresar una fecha");
568     ConsoleKey _k = Console.ReadKey(true).Key;
569     if (_k == ConsoleKey.V) datesWereCorrect = true;
570 }
571 }
572 }
573 private static void settings() //User settings
574 {
575     throw new NotImplementedException();
576 }
577 }
578 private static void errorHandling(Exception err)
579 {
580     Console.Clear();
581     Console.WriteLine(" ");
582     Console.WriteLine(" ");
583     Console.WriteLine(" ");
584     Console.WriteLine(" ");
585     Console.WriteLine(" ");
586     Console.WriteLine(" ");
587     Console.WriteLine(" ");
588     Console.WriteLine(" ");

```

```

589     Console.WriteLine(err.Message);
590     Console.WriteLine(" ");
591     Console.WriteLine(" ");
592     Console.WriteLine(" ");
593     Console.WriteLine("Presione una tecla para continuar");
594     Console.ReadKey();
595 }
596 }
597 }
598

```

Cliente Común

Nombre	Cédula	Tel.	Mail	Dirección	Usuario	Contrasena	Montevideo?
Jorge	45876 212	091425 631	jorgito@gmail.com	Bulevar Artigas 87546	Jorge	Jorge	true
Javier	54789 653	091879 564	javier@gmail.com	Bulevar Artiga	javier	javier	true

				s 9746 3				
Juan	16549 829	091879 564	juan@gmail.com	Bulev ar Artiga s 1087 465	juan	juan		false

Cliente Empresa

No mbr e	Raz ón soci al	Rut	Mail	Tel.	Dire cció n	Usu ario	Cont rase ña	Mont evide o?	Des cue nto
com pan y1	com pan y1 s.a.	154 684 654	company 1@gmail. com	25842 14357 9	Luis Albe rto de Herr era 154 843 223	com pan y1	com pany 1	true	3
com pan y2	com pan y2 s.a.	878 746 845	company 2@gmail. com	65487 65132 1	Luis Albe rto de Herr era 648	com pan y2	com pany 2	true	7

					948 455					
com pan y3	com pan y3 s.a.	346 456 148	company 3@gmail. com	32456 45615 51	Luis Albe rto de Herr era 878 456 456	com pan y3	com pany 3	false	5	

Productos

Nombre	Precio	Descripción	Montevideo?
Escarola	59	Precio por Kg	False
Espinaca	24	Precio por Kg	False
Croquetas	99	Precio por Kg	False
Buñuelo	40	Precio por Kg	True
Detergente	60	Precio por L	False
Jabón de manos	35	Precio por unidad	true
Toallas	70	Precio por unidad	False
Sábanas	150	Precio por unidad	False
PC Gamer	12500	Precio por unidad	False
Televisor Led	1500	Precio por unidad	False