



Federico García – 242015

Grupo: M2B

Docente: Luis Dentone

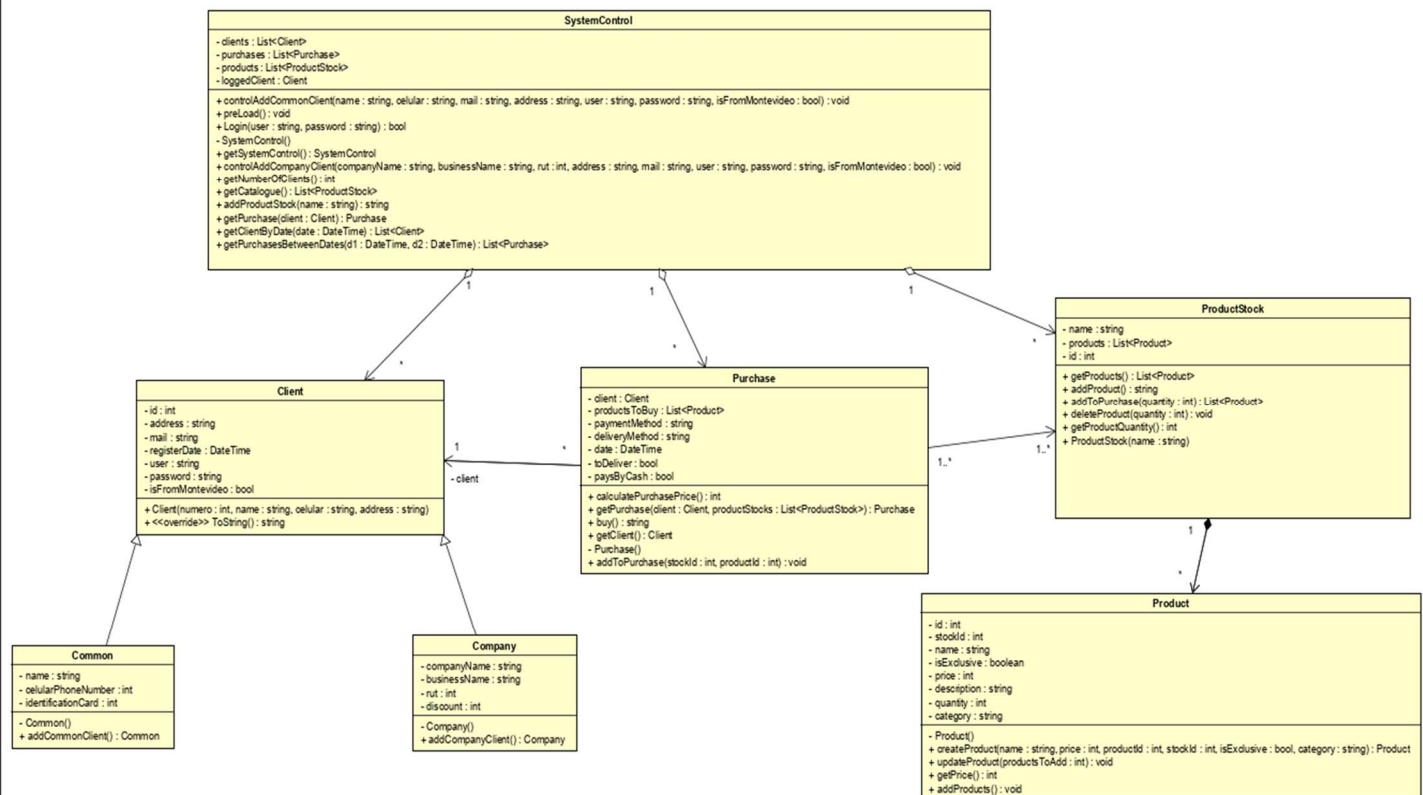


Andrés Valle Lisboa – 246109

Grupo: M2B

Docente: Luis Dentone

## UML



# Program

[illegible]

```

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 bussinesName = 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, bussinesName, 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 correcto");
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     errorHandler(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         else
182         {
183             Console.WriteLine("El usuario o la contraseña no fueron correctos");
184             Console.WriteLine("Presione una tecla para volver a intentarlo o presione \"v\" para volver al menú principal");
185             ConsoleKey _key = Console.ReadKey(true).Key;
186             if (_key == ConsoleKey.V)
187             {
188                 wasLoginSuccessful = true;
189             }
190         }
191     }
192 }
193 catch (Exception err)
194 {
195     errorHandler(err);
196 }
197 }
198
199 private static void _system()
200 {
201     bool optionIsNotCorrect = true;
202     while (optionIsNotCorrect)
203     {
204         bool _optionNotCorrect = true;
205         while (_optionNotCorrect)
206         {
207             Console.Clear();
208             Console.WriteLine("1 - Agregar productos");

```

```

209             Console.WriteLine("2 - Ver clientes registrados");
210             Console.WriteLine("3 - Compras realizadas");
211             Console.WriteLine("0 - Salir");
212
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                 Console.Clear();
324                 List<ProductStock> productStocks = _system.getCatalogue();
325                 string line = "-----";
326                 Console.WriteLine(line);
327                 Console.WriteLine("|Nombre      |      Precio      Cantidad      |      StockId      |      ProductId|");
328                 Console.WriteLine(line);
329                 switch (k)
330                 {
331                     case ConsoleKey.NumPad1:
332                     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         break;
348     case ConsoleKey.NumPad2:
349     case ConsoleKey.D2:
350         foreach (ProductStock _productStock in productStocks)
351         {
352             if (_productStock.Name == "Congelados")
353             {
354                 List<Product> _products = _productStock.Products;
355                 foreach (Product p in _products)
356                 {
357                     Console.WriteLine(line);
358                     Console.WriteLine("|" + p.Name + " | " + p.Price + " | " + p.Quantity + " | " + p.StockId + " | " + p.Id);
359                     Console.WriteLine(line);
360                 }
361             }
362         }
363         break;
364     case ConsoleKey.NumPad3:
365     case ConsoleKey.D3:
366         foreach (ProductStock _productStock in productStocks)
367         {
368             if (_productStock.Name == "Hogar")
369             {
370                 List<Product> _products = _productStock.Products;
371                 foreach (Product p in _products)
372                 {
373                     Console.WriteLine(line);
374                     Console.WriteLine("|" + p.Name + " | " + p.Price + " | " + p.Quantity + " | " + p.StockId + " | " + p.Id);
375                     Console.WriteLine(line);
376                 }
377             }
378         }
379         break;
380     case ConsoleKey.NumPad4:
381     case ConsoleKey.D4:
382         foreach (ProductStock _productStock in productStocks)
383         {
384             if (_productStock.Name == "T xtiles")

```

```

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     bool 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());

```



```

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     }
470     _optionCorrect = false;
471     while (!_optionCorrect)
472     {
473         Console.Clear();
474         Console.WriteLine("Especifique si pasará a retirar el producto o quiere que se lo envíen");
475         Console.WriteLine("1 - Retiro");
476         Console.WriteLine("2 - Envío");
477         ConsoleKey key = Console.ReadKey(true).Key;
478         switch (key)
479         {
480             case ConsoleKey.NumPad1:
481             case ConsoleKey.D1:
482                 toDeliver = false;
483                 _optionCorrect = true;
484                 break;
485             case ConsoleKey.NumPad2:
486             case ConsoleKey.D2:
487                 toDeliver = true;
488                 _optionCorrect = true;
489                 break;
490             case ConsoleKey.NumPad2:
491             case ConsoleKey.D2:
492                 toDeliver = true;
493                 _optionCorrect = true;
494                 break;
495         }
496     }
497     purchase.PaysByCash = paysByCash;
498     purchase.ToDeliver = toDeliver;
499     message = purchase.buy();
500     Console.WriteLine(message);
501     Console.WriteLine("Presione una tecla para continuar");
502     Console.ReadKey();
503     hasPurchaseEnded = true;
504 }
505 }
506 else
507 {
508     {
509         optionCorrect = true;
510     }
511 }
512 }
513 catch (Exception err)
514 {
515     errorHandler(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     isLogged = 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                 }
557                 if (p.Client.GetType() == typeof(Company))
558                 {
559                     Console.WriteLine("| " + ((Company)p.Client).CompanyName + " | " + p.Date + " | " + p.ProductsQuantity + " |");
560                 }
561             }
562         }
563     }
564     else
565     {
566         Console.WriteLine("Ingrese fechas correctas");
567     }
568     Console.WriteLine("presione \"v\" para volver al menú o presione cualquier tecla para volver a ingresar una fecha");
569     ConsoleKey _k = Console.ReadKey(true).Key;
570     if (_k == ConsoleKey.V) datesWereCorrect = true;
571 }
572 }
573
574 private static void settings() //User settings
575 {
576     throw new NotImplementedException();
577 }
578
579 private static void errorHandler(Exception err)
580 {
581     Console.Clear();
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

Nom bre	Cédul a	Tel.	Mail	Direcc ión	Usua rio	Contras eña	Montevi deo?
Jorge	45876212	091425631	<a href="mailto:jorgito@gmail.com">jorgito@gmail.com</a>	Bulevar Artigas 87546	Jorge	Jorge	true
Javier	54789653	091879564	<a href="mailto:javier@gmail.com">javier@gmail.com</a>	Bulevar Artiga	javier	javier	true



				s 9746 3			
Juan	16549 829	091879 564	<a href="mailto:juan@gmail.com">juan@gmail.com</a>	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 evid e?	Des cuen to
com pan y1	com pan y1 s.a.	154 684 654	<a href="mailto:company1@gmail.com">company1@gmail.com</a>	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	<a href="mailto:company2@gmail.com">company2@gmail.com</a>	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	<a href="mailto:company3@gmail.com">company 3@gmail. com</a>	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