



## Plan de pruebas de la aplicación SupermercadoOnline

Los niveles de prueba que se van a aplicar son los siguientes:

- Pruebas de aceptación. Las pruebas de aceptación se definirán siguiendo una estrategia basada en casos de uso y se ejecutarán de forma manual.
- Pruebas de integración. La estrategia para la definición del orden de las pruebas de integración será jerárquica. Se probará:
  - La integración entre la capa de negocio y la de persistencia. En este caso, para la definición de los casos de prueba se utilizarán técnica de métodos y caja negra y se utilizará JUnit.
  - La integración entre las tres capas. En este caso, para la definición de los casos de prueba se utilizarán técnica de casos de uso y se utilizarán Junit y FEST.
- Pruebas unitarias. Se utilizará la técnica de prueba de métodos, usando técnicas de caja negra (partición equivalente y AVL) para la definición de los casos de prueba de cada método de cada clase o componente. Será necesaria la utilización de JUnit, Mockito y FEST.

A continuación, se muestra una especificación detallada de los casos de prueba a aplicar en cada nivel mencionado anteriormente.

### PRUEBAS DE ACEPTACIÓN

En base a los casos de uso se identifican los siguientes escenarios:

#### A1. CU: Realizar pedido

- a. Pedido válido (usuario existente, stock suficiente, horario de recogida válido)
- b. Pedido válido (usuario existente, stock suficiente, horario de recogida válido, con descuento por más de 10 compras)
- c. Pedido no válido (usuario existente, añadir un artículo del pedido con stock insuficiente, horario de recogida válido)
- d. Pedido no válido (usuario inexistente)
- e. Pedido no válido (horario de recogida inválido)

Los casos de prueba definidos para cada uno de estos escenarios, suponiendo que el estado inicial del sistema es el siguiente:

#### Usuario registrado:

Jose Manuel García Sánchez, 27516415G, josemanuel@gmail.com, Calle Castilla 36

#### El sistema también contiene los siguientes elementos:

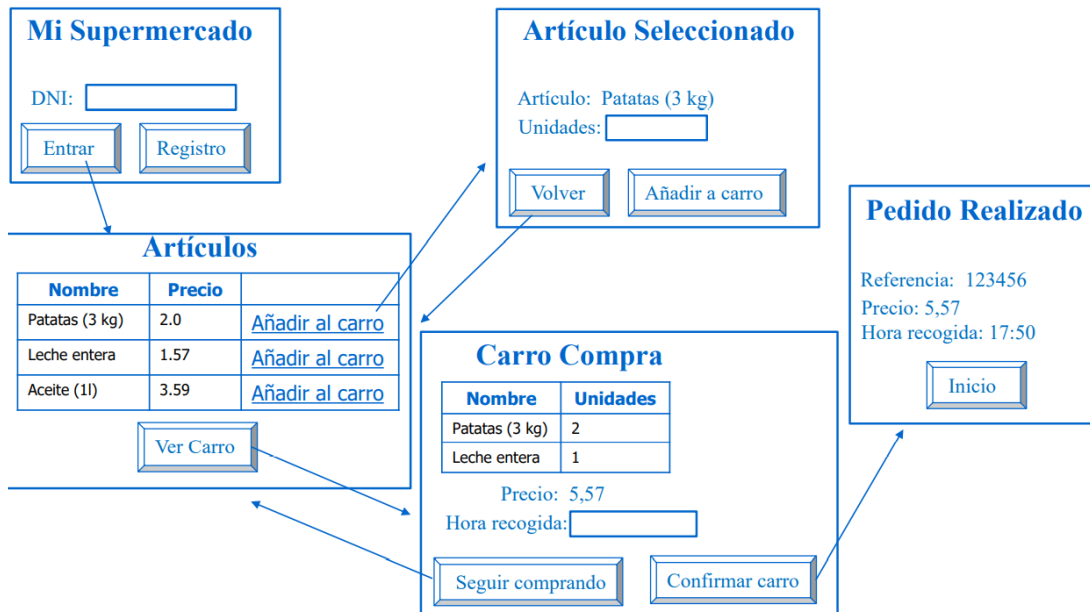


Tabla 1. Casos de prueba de aceptación (ejecutados en este orden)

Identificado r	Entrada	Resultado
A1.a	27516415G, 1 ud patatas, 17:50h	Pedido válido
A1.b	27516415G (con 11 compras), 1 ud patatas, 17:50h	Pedido válido
A1.c	27516415G, 5000 ud patatas, 17:50h	Stock insuficiente
A1.d	11111111A, 1 ud patatas, 17:50h	Usuario inexistente
A1.e	27516415G, 1 ud patatas, 23:00h	Fecha inválida

## PRUEBAS DE INTEGRACIÓN //TODO

El orden de las pruebas y los casos de prueba a realizar sería el siguiente:

1. GestionImpuestoCirculacion con ImpuestosDAO. Se usarían los mismos casos de prueba definidos como UGIC.x en la sección de pruebas unitarias, aquí renombrados como IGIC.x.
2. VistaFuncionario con GestionImpuestosCirculacion y con ImpuestosDAO. Estas pruebas coincidirían con las pruebas de aceptación, aunque en este caso se automatizarían utilizando la librería FEST.

Podría hacerse una prueba previa de integración VistaFuncionario y GestionImpuestosCirculacion usando un Mock de la DAO, pero no se considera necesario.

En este ejemplo, la prueba de la integración de los componentes DAO con la BBDD no tiene sentido pues se implementa con un fichero.



## PRUEBAS UNITARIAS

### ● Pruebas unitarias de la capa DAO `ArticulosDao`

Se aplica prueba de métodos, siendo los casos de prueba definidos para cada método los siguientes y partiendo de una situación inicial en la que la base de datos tiene el artículo (1 kg de patatas, 30 unidades de stock y 5 euros de precio):

- Método `Articulo creaArticulo(Articulo a)`:

Identificador	Entrada	Valor esperado
UID.1a	Pan, 30 unidades, 0.3 euros	Devuelve el Artículo Pan ya creado
UID.1b	1 kg de patatas, 30 unidades de stock y 5 euros	null

- Método `Articulo articuloPorNombre(String nombre)`:

Identificador	Entrada	Valor esperado
UID.2a	1 kg de patatas	Devuelve el Artículo 1 kg de patatas
UID.2b	Solomillo	null

- Método `List<Articulo> articulos()`:

Identificador	Entrada	Valor esperado
UID.3a	Situación inicial, tienes el Artículo 1 kg de patatas	Lista con 1 elemento (el artículo 1 kg de patatas)
UID.3b	No tienes ningún Artículo creado	Lista Vacía

- Método `Articulo modificaArticulo(Articulo a)`

Identificador	Entrada	Valor esperado
UID.4a	Artículo 1 kg de patatas	Devuelve el Artículo 1 kg de patatas.
UID.4b	Artículo Vino	Null, debido a que el artículo no existe en la BBDD.

- Método `Articulo eliminaArticulo(Articulo a)`



Identificador	Entrada	Valor esperado
UID.5a	Artículo 1 kg de patatas	Devuelve el Artículo 1 kg de patatas.
UID.5b	Artículo Vino	Null, debido a que el artículo no existe en la BBDD.

- Método Artículo articuloPorId(long id)

Identificador	Entrada	Valor esperado
UID.6a	Id del Artículo 1 kg de patatas	Devuelve el Artículo con esa id (1 kg de patatas).
UID.6b	Id de un Artículo que no está en la BBDD	null

• **Pruebas unitarias de la capa de negocio GestionPedidos //TODO**

Para poder llevar a cabo estas pruebas, será necesario el uso de mocks para las interfaces IContribuyentesDAO e IVehiculosDAO. Se aplica prueba de métodos, siendo los casos de prueba definidos para cada método los siguientes:

- Método altaVehiculo(): Se pueden usar los mismos casos identificados para el caso de uso 3 (A3a, A3b, A3c), que aquí se renombrarían según aparece en la tabla.

Identificador	Entrada	Valor esperado
UGIC.1a	(2020BBB, 10/10/2003, 75) a 22222222B	2020BBB 10/10/2003 75

- Método bajaVehiculo(): Se pueden usar los mismos casos identificados para el caso de uso 4, que aquí se renombrarían según aparece en la tabla.

Identificador	Entrada	Valor esperado
UGIC.2a	2020BBB 22222222B	Vehiculo con matrícula 2020BBB
UGIC.2b	2020CCC 33333333C	null
UGIC.2c	2020BBB 88888888G	null
UGIC.2d	1237HDT 11111111A	OperacionNoValida

- Método vehiculo()



Identificador	Entrada	Valor esperado
UGIC.3a	3441DDB	3441DDB 01/01/2009 22
UGIC.3b	2220AAB	null

- Método altaContribuyente: Se pueden usar los mismos casos identificados para el caso de uso 1, que aquí se renombrarían según aparece en la tabla.

Identificador	Entrada	Valor esperado
UGIC.4a	Lidia López Revuelta, 77777777G	Lidia López Revuelta, 77777777G
UGIC.4b	Alba Muñiz López, 11111111A	null

- Método contribuyente(): Se pueden usar los mismos casos identificados para el caso de uso 5, que aquí se renombrarían según aparece en la tabla.

Identificador	Entrada	Valor esperado
UGIC.6a	44444444D	Amador Rivas Cobo, 44444444D, etc.
UGIC.6b	77777777G	null

- Método totalContribuyente():

- Método bajaContribuyente(): Se pueden usar los mismos casos identificados para el caso de uso 2, que aquí se renombrarían según aparece en la tabla.

Identificador	Entrada	Valor esperado
UGIC.8a	77777777G	(77777777G, Lidia López Revuelta)
UGIC.8b	11111111A	OperacionNoValida
UGIC.8c	88888888G	null

*Pedro Monje Onandía*



Práctica 3 Plan de pruebas  
Procesos de la IS 4º Grado en Ing. Informática  
Facultad de Ciencias Universidad de Cantabria  
**Pablo Almohalla Gómez** **Pedro Monje Onandía**

