# Plan de pruebas Supermercado UC

Este documento recoge la especificación del plan de pruebas para la aplicación SupermercadoUC,

Contiene las pruebas:

Aceptación – Basadas en la especificación de los casos de uso, se ejecutan de forma manual

Unitarias –

* Capa DAO
* Capa Bussines

**Pruebas de Aceptación**

En base a los casos de uso se identifican los siguientes escenarios (suponemos que la interfaz por construcción no permite dejar ningún campo requerido en blanco, para evitar casos de prueba referidos a información no disponible):

A1. CU: Realizar Pedido

1. Pedido realizado con éxito.
2. Pedido no realizado (Horario indicado no valido).
3. Error en acceso a base de datos. (Vuelve a resumen carrito)

A1.1 CU : Articulo Añadido

1. Articulo añadido con éxito.
2. Articulo no añadido (Stock insuficiente).
3. Error en acceso a base de datos. (Vuelve a lista articulos)

A1.2 CU : Login Usuario

1. Login correcto
   1. Se muestra lista de artículos
   2. Error acceso a base de datos. (Vuelve a login)
2. Login incorrecto (Usuario inexistente). (Vuelve a login)
3. Error en acceso a bse de datos (Vuelve a login)

**Pruebas unitarias**

**Pruebas unitarias de la capa DAO**

Para la implementación de las pruebas será necesario el uso de mocks para las entidades Query y EntityManager, se implementarán estos mocks mediante mockito

Clase ArticulosDAO

|  |  |  |
| --- | --- | --- |
| UP1.a | Articulo(10,“A”,10), 10 | True |
| UP1.b | Articulo(5,“B”,5), 10 | False |
| UP1.c | Articulo(100,“F”,20), 20 | DataAccesException |

* Método articuloConStock(Articulo articulo, int cantidad): boolean throws DataAccesException

Para la implementación de las pruebas será necesario el uso de mocks para las interfaces DAO correspondiente, se implementarán estos mocks mediante mockito.

**Pruebas unitarias de la capa Bussines**

Clase GestionPedidos

* Método incluyeArticulo(Articulo articulo, int cantidad) throws DataAccessException, StockInsuficienteException

|  |  |  |
| --- | --- | --- |
| UB1.a | articuloA = Articulo(“A”,20.0,30), 10 | LineaPedido(ArticuloA,10,200.0) (Comprobar que se ha añadido en la lista del componente) |
| UB1.b | Con LineaPedido de UB1.a en GestionPedidos  ArticuloA = Articulo(“A”,20.0,30),10 | LineaPedido(ArticuloA,20,400.0) (Comprobar que se ha actualizado en la lista del componente) |
| UB1.c | Con LineaPedido de UB1.b en GestionPedidos  ArticuloA = Articulo(“A”,20.0,30),20 | LineaPedido(ArticuloA,20,400.0) (Comprobar que no se ha actualizado en la lista del componente) y se lanza excepción  StockInsuficienteException |
| UB1.e | articuloB = Articulo(“B”,20.0,0), 10 | StockInsuficienteException |
| UB1.f | articuloF = Articulo(“F”,20.0,10), 10 | DataAccessException |

**Informe Ejecución de las pruebas**

Durante la ejecución de las pruebas unitarias de la capa Bussines, se encontraron fallos en las pruebas UB1.b y UB1.c en la actualización de la línea pedido ya existente. Corrigiendo el método para que cumpliese la prueba no se encontraron más fallos.

La ejecución de las pruebas de la capa de persistencia fue satisfactoria.