**Configuración de escenarios para las pruebas de integración**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Clase** | **Stage** |
| setUpUnitmeasure1 | ProductServiceTest | Se crea una nueva unidad de medida y se guarda en su repositorio. El ID es autogenerado, por lo que no es necesario hacer ningún tipo de set al objeto. |
| setUpUnitmeasure1op2 | ProductServiceTest | Se crea una nueva unidad de medida y se guarda en su repositorio. El ID es autogenerado, por lo que no es necesario hacer ningún tipo de set al objeto. |
| setUpUnitmeasure2 | ProductServiceTest | Se crea una nueva unidad de medida y se guarda en su repositorio. El ID es autogenerado, por lo que no es necesario hacer ningún tipo de set al objeto. |
| setUpProductCategory | ProductServiceTest | Se crea una nueva categoría y se guarda en su repositorio. El ID es autogenerado, se hace set de la fecha de modificación, del nombre y del Rowguid. |
| setUpProductCategoryop2 | ProductServiceTest | Se crea una nueva categoría y se guarda en su repositorio. El ID es autogenerado, se hace set de la fecha de modificación, del nombre y del Rowguid. |
| setUpProductSubcategory | ProductServiceTest | Se crea una nueva subcategoría y se guarda en su repositorio. El ID es autogenerado, se hace set de la fecha de modificación, del nombre, de la categoría del producto y del Rowguid. |
| setUpProductSubcategoryop2 | ProductServiceTest | Se crea una nueva subcategoría y se guarda en su repositorio. El ID es autogenerado, se hace set de la fecha de modificación, del nombre, de la categoría del producto y del Rowguid. |
| setUpProduct | ProductServiceTest | Se crea un nuevo producto y se guarda en su repositorio. El ID es autogenerado, se hace set de múltiples atributos como el nombre, el numero del producto, la subcategoría, la fecha de inicio y fin de venta, el tamaño, peso y las dos unidades de medida. |
| setUpProductCategory | ProductSubcategoryTest | Se crea una nueva categoría y se guarda en su repositorio. El ID es autogenerado, se hace set de la fecha de modificación, del nombre y del Rowguid. |
| setUpProductCategoryop2 | ProductSubcategoryTest | Se crea una nueva categoría y se guarda en su repositorio. El ID es autogenerado, se hace set de la fecha de modificación, del nombre y del Rowguid. |
| setUpProductSubcategory | ProductSubcategoryTest | Se crea una nueva subcategoría y se guarda en su repositorio. El ID es autogenerado, se hace set de la fecha de modificación, del nombre, de la categoría del producto y del Rowguid. |
| setUpProductSubcategoryop2 | ProductSubcategoryTest | Se crea una nueva subcategoría y se guarda en su repositorio. El ID es autogenerado, se hace set de la fecha de modificación, del nombre, de la categoría del producto y del Rowguid. |
| setUpProduct | WorkorderServiceTest | Se crea un nuevo producto y se guarda en su repositorio. El ID es autogenerado, se hace set de múltiples atributos como el nombre, el número del producto, la subcategoría, la fecha de inicio y fin de venta, el tamaño, peso y las dos unidades de medida. |
| setUpProductop2 | WorkorderServiceTest | Se crea un nuevo producto y se guarda en su repositorio. El ID es autogenerado, se hace set de múltiples atributos como el nombre, el número del producto, la subcategoría, la fecha de inicio y fin de venta, el tamaño, peso y las dos unidades de medida. |
| setUpScrapreason | WorkorderServiceTest | Se crea una nueva scrapreason y se guarda en su repositorio. . El ID es autogenerado, por lo que no es necesario hacer ningún tipo de set al objeto. |
| setUpScrapreasonop2 | WorkorderServiceTest | Se crea una nueva scrapreason y se guarda en su repositorio. . El ID es autogenerado, por lo que no es necesario hacer ningún tipo de set al objeto. |

**Diseño de casos de prueba integración**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-1 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveCategory frente a un caso valido donde no deberían de lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +saveCategory\_ValidCase() |  | ModifiedData = System.currentTimeMillis()  name = “Category”  rowGuid = 0 | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-2 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveCategory frente a un caso inválido donde el nombre no cumple con lo exigido. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +saveCategory\_noValidName() |  | ModifiedData = System.currentTimeMillis()  name = “c”  rowGuid = 0 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-3 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso válido donde no deberían lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +editCategory\_ValidCaseAllChange() | setUpProductCategory() | ModifiedData = System.currentTimeMillis()  name = “Category2”  rowGuid = 202  id = pc.getProductcategoryid() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-4 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido debido a que el nombre no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +editCategory\_noValidName() | setUpProductCategory() | ModifiedData = System.currentTimeMillis()  name = “c”  rowGuid = 0  id = pc.getProductcategoryid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-5 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido debido a que la id no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +editCategory\_noValidId () |  | ModifiedData = System.currentTimeMillis()  name = “Category2”  rowGuid = 0  id = 101 | El método lanza una excepción debido a que la fecha es posterior a la actual. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-1 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso válido en el que se cumplan con todas las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_ValidCase() | setUpProductCategory()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis();  size = “30”  weight = BigDecimal(“30.0”  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-2 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde el número del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidProductNumber() | setUpProductCategoryop2()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis();  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-3 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde las fechas del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidDate() | setUpProductCategoryop2()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = 100  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-4 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde el peso del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidWeigth() | setUpProductCategoryop2()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“-30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-5 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde el tamaño del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidSize() | setUpProductCategoryop2()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-6 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde la subcategoría del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidSubcategory() | setUpProductCategoryop2()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = 78  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-7 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde la categoría del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidCategory() | setUpProductCategoryop2()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = 809  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-8 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde la unidad de medida 1 del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidUnitMeasure1() | setUpProductCategoryop2()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = 897  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-9 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde la unidad de medida 2 del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidUnitMeasure2() | setUpProductCategoryop2()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = 876  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-10 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde el id del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidId() | setUpProductCategoryop2()  setUpUnitmeasure1op2()  setUpUnitmeasure2op2()  setUpProductSubcategory()  setUpProduct() | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-11 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso válido donde se cumplen todas las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +saveProduct\_validCase() | setUpProductCategory()  setUpUnitmeasure1()  setUpUnitmeasure2()  setUpProductSubcategory() | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** U-12 | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido en el que el número del producto no cumple con las verificaciones | | | | | | | | |
| **Clase** | **Método** | | | **Stage** | | **Input** | | **Output** |
| ProductServiceTest | +saveProduct\_noValidProductNumber() | | | setUpProductCategory()  setUpUnitmeasure1()  setUpUnitmeasure2()  setUpProductSubcategory() | | name = “Product1”  productNumber = “”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |
| **Identificador:** U-13 | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la fechas no cumplen con las verificaciones | | | | | | | | |
| **Clase** | | **Método** | **Stage** | | **Input** | | **Output** | |
| ProductServiceTest | | +saveProduct\_noValidDate() | setUpProductCategory()  setUpUnitmeasure1()  setUpUnitmeasure2()  setUpProductSubcategory()  setUpProduct() | | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = 200  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones | |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** U-14 | | | | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde el peso no cumplen con las verificaciones | | | | | | | | | | | |
| **Clase** | **Método** | | | | | **Stage** | | **Input** | | | **Output** |
| ProductServiceTest | +saveProduct\_noValidWeigth() | | | | | setUpProductCategory()  setUpUnitmeasure1()  setUpUnitmeasure2()  setUpProductSubcategory() | | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“-30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |
| **Identificador:** U-15 | | | | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde el tamaño no cumplen con las verificaciones | | | | | | | | | | | |
| **Clase** | | **Método** | | | **Stage** | | **Input** | | **Output** | | |
| ProductServiceTest | | +saveProduct\_noValidSize() | | | setUpProductCategory()  setUpUnitmeasure1()  setUpUnitmeasure2()  setUpProductSubcategory() | | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “-30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones | | |
| **Identificador:** U-16 | | | | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la subcategoría no cumplen con las verificaciones | | | | | | | | | | | |
| **Clase** | | | **Método** | **Stage** | | | **Input** | | | **Output** | |
| ProductServiceTest | | | +saveProduct\_noValidSubcategory() | setUpProductCategory()  setUpUnitmeasure1()  setUpUnitmeasure2()  setUpProductSubcategory() | | | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = 3242  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-17 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la categoría no cumplen con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +saveProduct\_noValidCategory() | setUpProductCategory()  setUpUnitmeasure1()  setUpUnitmeasure2()  setUpProductSubcategory() | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = 324  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-18 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la unidad de medida 1 no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +saveProduct\_noValidUnitMeasure1() | setUpProductCategory()  setUpUnitmeasure1()  setUpUnitmeasure2()  setUpProductSubcategory() | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = 434  measureunit2id = un2.getUnitmeasurecode() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-19 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la unidad de medida 2 no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +saveProduct\_noValidUnitMeasure2() | setUpProductCategory()  setUpUnitmeasure1()  setUpUnitmeasure2()  setUpProductSubcategory()  setUpProduct() | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = 234 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-1 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveSubcategory frente a un caso válido donde no deberían lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +saveSubcategory\_ValidCase() | setUpProducCategory() | ModifiedDate = System.currentTimeMillis()  name = “Subcategory”  rowGuid = 0  productCategoryId = pc.getProductCategoryid() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-2 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveSubcategory frente a un caso inválido en que la categoría no cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +saveSubcategory\_noCategory() |  | ModifiedDate = System.currentTimeMillis()  name = “Subcategory”  rowGuid = 0  productCategoryId = 212 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-3 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveSubcategory frente a un caso inválido en que el nombno cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +saveSubcategory\_noValidName() |  | ModifiedDate = System.currentTimeMillis()  name = “Sub”  rowGuid = 0  productCategoryId = pc.getProductCategoryid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-4 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editSubcategory frente a un caso válido donde no deberían lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +editSubcategory\_ValidCaseAllChange() | setUpProductCategoryop2()  setUpProductCategory()  setUpProductSubcategory() | Id = sc.getProductsubcategoryid()  ModifiedDate = System.currentTimeMillis()  name = “Subcategory1”  rowGuid = 4  productCategoryId = pcop2.getProductCategoryid() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-5 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido donde la categoría no cumple con las validaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +editSubcategory\_noCategory() | setUpProductCategory()  setUpProductSubcategory() | Id = sc.getProductsubcategoryid()  ModifiedDate = System.currentTimeMillis()  name = “Subcategory1”  rowGuid = 4  productCategoryId = 24 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-6 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido donde el nombre no cumple con las validaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +editSubcategory\_noValidName() | setUpProductCategory()  setUpProductSubcategory() | Id = sc.getProductsubcategoryid()  ModifiedDate = System.currentTimeMillis()  name = “sc”  rowGuid = 4  productCategoryId = pcop2.getProductCategoryid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-7 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido donde la id no cumple con las validaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +editSubcategory\_noValidId() | setUpProductCategory()  setUpProductSubcategory() | Id = 423  ModifiedDate = System.currentTimeMillis()  name = “Subcategory1”  rowGuid = 4  productCategoryId = pcop2.getProductCategoryid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-1 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso válido donde no deberían lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_ValidCase() | setUPProduct()  setUpScrapreason() | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = 1  scrappedqty = 2  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-2 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido donde el producto no cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noProduct() | setUPProduct()  setUpScrapreason() | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = 1  scrappedqty = 2  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = 322 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-3 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido donde screapreason no cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noScrapreason() | setUPProduct()  setUpScrapreason() | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = 1  scrappedqty = 2  startdate = 1000  scrapreasonId = 523  productid = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** U-4 | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido donde orderQty no cumple con las verificaciones. | | | | | | | | |
| **Clase** | **Método** | | | **Stage** | | **Input** | | **Output** |
| WorkorderService | +saveWorkorder\_orderQtySmaller0() | | | setUPProduct()  setUpScrapreason() | | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = -1  scrappedqty = 2  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = p.getProductid() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |
| **Identificador:** U-5 | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido donde el scrappedqty no cumple con las verificaciones. | | | | | | | | |
| **Clase** | | **Método** | **Stage** | | **Input** | | **Output** | |
| WorkorderService | | +saveWorkorder\_scrappedqtySmaller0() | setUPProduct()  setUpScrapreason() | | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = 1  scrappedqty = -22  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = p.getProductid() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-6 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido en el que las fechas no cumplen con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_startDateGreaterEndDate() | setUPProduct()  setUpScrapreason() | duedate = 100  enddate = 10  modifieddate = 100  orderqty = 1  scrappedqty = 2  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-7 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso válido en el que no deberían lanzarse excepciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +editWorkorder\_ValidCaseAllChange() | setUPProduct()  setUpScrapreason()  setUpWorkorder()  setUpProductop2()  setUpScrapreasonop2() | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-8 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el producto no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noProduct() | setUpWorkorder()  setUpProductop2()  setUpScrapreasonop2() | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = 234 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-9 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el scrapreason no cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noScrapreason() | setUpScrapreason()  setUpWorkorder()  setUpProductop2()  setUpScrapreasonop2() | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = 324  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-10 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el orderQty no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_orderQtySmaller0() | setUpScrapreason()  setUpWorkorder()  setUpProductop2()  setUpScrapreasonop2() | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = -5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-11 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el scrappedqty no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_scrappedqtySmaller0() | setUPProduct()  setUpScrapreason()  setUpWorkorder()  setUpProductop2()  setUpScrapreasonop2() | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = -6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-12 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que las fechas no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_startDateGreaterEndDate() | setUpScrapreason()  setUpWorkorder()  setUpProductop2()  setUpScrapreasonop2() | Id = wo.getWorkorderid()  duedate = 200  enddate = 2  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-13 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el id no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noValidID() | setUpScrapreason()  setUpWorkorder()  setUpProductop2()  setUpScrapreasonop2() | Id = 3243  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

**Diseño de casos de pruebas unitarias usando Mockito**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-1 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveCategory frente a un caso valido donde no deberían de lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +saveCategory\_ValidCase() |  | ModifiedData = System.currentTimeMillis()  name = “Category”  rowGuid = 0 | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-2 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveCategory frente a un caso inválido donde el nombre no cumple con lo exigido. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +saveCategory\_noValidName() |  | ModifiedData = System.currentTimeMillis()  name = “c”  rowGuid = 0 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-3 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso válido donde no deberían lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +editCategory\_ValidCaseAllChange() |  | ModifiedData = System.currentTimeMillis()  name = “Category2”  rowGuid = 202  id = pc.getProductcategoryid() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-4 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido debido a que el nombre no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +editCategory\_noValidName() |  | ModifiedData = System.currentTimeMillis()  name = “c”  rowGuid = 0  id = pc.getProductcategoryid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-5 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido debido a que la id no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductCategoryServiceTest | +editCategory\_noValidId () |  | ModifiedData = System.currentTimeMillis()  name = “Category2”  rowGuid = 0  id = 101 | El método lanza una excepción debido a que la fecha es posterior a la actual. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-1 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso válido en el que se cumplan con todas las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_ValidCase() |  | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis();  size = “30”  weight = BigDecimal(“30.0”  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-2 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde el número del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidProductNumber() |  | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis();  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-3 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde las fechas del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidDate() |  | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = 100  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-4 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde el peso del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidWeigth() |  | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“-30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-5 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde el tamaño del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidSize() |  | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-6 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde la subcategoría del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidSubcategory() |  | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = 78  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-7 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde la categoría del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidCategory() |  | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = 809  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-8 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde la unidad de medida 1 del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidUnitMeasure1() |  | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = 897  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-9 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde la unidad de medida 2 del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidUnitMeasure2() |  | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = 876  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-10 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editProduct frente a un caso inválido donde el id del producto no cumple con la verificación | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +editProduct\_noValidId() |  | name = “ProductN”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = psop2.getProductsucategoryid()  productCategoryId = pcop2.getProductcategoryid()  measureunit1id = un1op.getUnitmeasurecode()  measureunit2id = un2op.getUnitmeasurecode()  id = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-11 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso válido donde se cumplen todas las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +saveProduct\_validCase() |  | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** U-12 | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido en el que el número del producto no cumple con las verificaciones | | | | | | | | |
| **Clase** | **Método** | | | **Stage** | | **Input** | | **Output** |
| ProductServiceTest | +saveProduct\_noValidProductNumber() | | |  | | name = “Product1”  productNumber = “”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |
| **Identificador:** U-13 | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la fechas no cumplen con las verificaciones | | | | | | | | |
| **Clase** | | **Método** | **Stage** | | **Input** | | **Output** | |
| ProductServiceTest | | +saveProduct\_noValidDate() |  | | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = 200  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones | |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** U-14 | | | | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde el peso no cumplen con las verificaciones | | | | | | | | | | | |
| **Clase** | **Método** | | | | | **Stage** | | **Input** | | | **Output** |
| ProductServiceTest | +saveProduct\_noValidWeigth() | | | | |  | | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“-30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |
| **Identificador:** U-15 | | | | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde el tamaño no cumplen con las verificaciones | | | | | | | | | | | |
| **Clase** | | **Método** | | | **Stage** | | **Input** | | **Output** | | |
| ProductServiceTest | | +saveProduct\_noValidSize() | | |  | | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “-30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones | | |
| **Identificador:** U-16 | | | | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la subcategoría no cumplen con las verificaciones | | | | | | | | | | | |
| **Clase** | | | **Método** | **Stage** | | | **Input** | | | **Output** | |
| ProductServiceTest | | | +saveProduct\_noValidSubcategory() |  | | | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = 3242  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-17 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la categoría no cumplen con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +saveProduct\_noValidCategory() |  | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = 324  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = un2.getUnitmeasurecode() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-18 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la unidad de medida 1 no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +saveProduct\_noValidUnitMeasure1() |  | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = 434  measureunit2id = un2.getUnitmeasurecode() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-19 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveProduct frente a un caso inválido donde la unidad de medida 2 no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductServiceTest | +saveProduct\_noValidUnitMeasure2() |  | name = “Product1”  productNumber = “100”  sellStartDate = System.currentTimeMillis()  sellEndDate = System.currentTimeMillis()  size = “30”  weight = BigDecimal(“30.0”)  productSubcategoryId = ps.getProductsucategoryid()  productCategoryId = pc.getProductcategoryid()  measureunit1id = un1.getUnitmeasurecode()  measureunit2id = 234 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-1 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveSubcategory frente a un caso válido donde no deberían lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +saveSubcategory\_ValidCase() |  | ModifiedDate = System.currentTimeMillis()  name = “Subcategory”  rowGuid = 0  productCategoryId = pc.getProductCategoryid() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-2 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveSubcategory frente a un caso inválido en que la categoría no cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +saveSubcategory\_noCategory() |  | ModifiedDate = System.currentTimeMillis()  name = “Subcategory”  rowGuid = 0  productCategoryId = 212 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-3 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveSubcategory frente a un caso inválido en que el nombno cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +saveSubcategory\_noValidName() |  | ModifiedDate = System.currentTimeMillis()  name = “Sub”  rowGuid = 0  productCategoryId = pc.getProductCategoryid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-4 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editSubcategory frente a un caso válido donde no deberían lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +editSubcategory\_ValidCaseAllChange() |  | Id = sc.getProductsubcategoryid()  ModifiedDate = System.currentTimeMillis()  name = “Subcategory1”  rowGuid = 4  productCategoryId = pcop2.getProductCategoryid() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-5 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido donde la categoría no cumple con las validaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +editSubcategory\_noCategory() |  | Id = sc.getProductsubcategoryid()  ModifiedDate = System.currentTimeMillis()  name = “Subcategory1”  rowGuid = 4  productCategoryId = 24 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-6 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido donde el nombre no cumple con las validaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +editSubcategory\_noValidName() |  | Id = sc.getProductsubcategoryid()  ModifiedDate = System.currentTimeMillis()  name = “sc”  rowGuid = 4  productCategoryId = pcop2.getProductCategoryid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-7 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editCategory frente a un caso inválido donde la id no cumple con las validaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| ProductSubcategoryServiceTest | +editSubcategory\_noValidId() |  | Id = 423  ModifiedDate = System.currentTimeMillis()  name = “Subcategory1”  rowGuid = 4  productCategoryId = pcop2.getProductCategoryid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-1 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso válido donde no deberían lanzarse excepciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_ValidCase() |  | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = 1  scrappedqty = 2  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-2 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido donde el producto no cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noProduct() |  | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = 1  scrappedqty = 2  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = 322 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-3 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido donde screapreason no cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noScrapreason() |  | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = 1  scrappedqty = 2  startdate = 1000  scrapreasonId = 523  productid = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** U-4 | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido donde orderQty no cumple con las verificaciones. | | | | | | | | |
| **Clase** | **Método** | | | **Stage** | | **Input** | | **Output** |
| WorkorderService | +saveWorkorder\_orderQtySmaller0() | | |  | | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = -1  scrappedqty = 2  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = p.getProductid() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |
| **Identificador:** U-5 | | | | | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido donde el scrappedqty no cumple con las verificaciones. | | | | | | | | |
| **Clase** | | **Método** | **Stage** | | **Input** | | **Output** | |
| WorkorderService | | +saveWorkorder\_scrappedqtySmaller0() |  | | duedate = 100  enddate = 10000  modifieddate = 100  orderqty = 1  scrappedqty = -22  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = p.getProductid() | | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-6 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método saveWorkorder frente a un caso inválido en el que las fechas no cumplen con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_startDateGreaterEndDate() |  | duedate = 100  enddate = 10  modifieddate = 100  orderqty = 1  scrappedqty = 2  startdate = 1000  scrapreasonId = sr.getScrapreasonid()  productid = p.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-7 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso válido en el que no deberían lanzarse excepciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +editWorkorder\_ValidCaseAllChange() |  | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | Se guarda la nueva instancia de tarea en el repositorio. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-8 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el producto no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noProduct() |  | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = 234 | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-9 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el scrapreason no cumple con las verificaciones. | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noScrapreason() |  | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = 324  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-10 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el orderQty no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_orderQtySmaller0() |  | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = -5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-11 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el scrappedqty no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_scrappedqtySmaller0() |  | Id = wo.getWorkorderid()  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = -6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-12 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que las fechas no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_startDateGreaterEndDate() |  | Id = wo.getWorkorderid()  duedate = 200  enddate = 2  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identificador:** U-13 | | | | |
| **Objetivo de la prueba:** Verificar el funcionamiento del método editWorkorder frente a un caso inválido en el que el id no cumple con las verificaciones | | | | |
| **Clase** | **Método** | **Stage** | **Input** | **Output** |
| WorkorderService | +saveWorkorder\_noValidID() |  | Id = 3243  duedate = 200  enddate = 20000  modifieddate = 200  orderqty = 5  scrappedqty = 6  startdate = 4000  scrapreasonId = srop2.getScrapreasonid()  productid = pop2.getProductid() | El método lanza una excepción debido a que no se cumplió con alguna de las verificaciones |