**Andrea Núñez Rodríguez – A00355453**

**Test cases design**

**Set up**

|  |  |  |
| --- | --- | --- |
| **Name** | **Class under test** | **Stage** |
| saveProductSetUp | ProdutService | One object of type Productcategory and Productsubcategory which are available in the productcategory and productsubcategory repository. |
| editProductSetUp | ProductService | One object of type Product which is available in the product repository. |
| saveTransactionHistorySetUp | TransactionhistoryService | One object of type Product which is available in the product repository. |
| editTransactionHistorySetUp | TransactionhistoryService | One object of type Transactionhistory which is available in the Transactionhistory repository. |
| saveDocumentSetUp | DocumentService | One object of type Productdocument and Product which are available in the Productdocument and Product repository. |
| editDocumentSetUp | DocumentService | One object of type Document which is available in the Document repository. |
| saveProductVendorSetUp | ProductVendorService | One object of type Product, Unit measure and Employee which are available in the Product, Unit measure and Employee repositories. |
| editProductVendorSetUp | ProductVendorService | One object of type ProductVendor which is available in the ProductVendor repository. |

**Test cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective:** To verify if the Product service correctly saves a product whit a product category and a product subcategory associated. | | | | |
| **Class** | **Method** | **Stage** | **Input** | **Output** |
| ProductService | save | saveProductSetUp | ProductcategoryId= 01  ProductsubcategoryId=01  number =” Number of product”  daystomanufacture=10  sellstart=”23/09/2020”  sellend= “23/10/2020” | The productcategory and productsubcategory objects are found.  The number of the product isn’t empty, the days to manufacture are greater than 0, and the sell start date is before of the sell end date.  They all are the same as the passed by parameter. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective:** To verify if the Product service correctly edits a product whit a product category and a product subcategory associated. | | | | |
| **Class** | **Method** | **Stage** | **Input** | **Output** |
| ProductService | edit | editProductSetUp | ProductId= 01  productNumber =””  daystomanufacture=0  sellstart=”23/10/2020”  sellend= “23/09/2020” | The number of the product is empty, the days to manufacture are equals than 0, and the sell start date isn’t before of the sell end date.  Therefore, the product cannot be edit successfully. |
| ProductService | edit | editProductSetUp | ProductId=01  number =” correct product number”  daystomanufacture=3  sellstart=”23/09/2020”  sellend= “23/10/2020” | The number of the product isn’t empty, the days to manufacture are greater than 0, and the sell start date is before of the sell end date. Therefore, the product is edited successfully. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective:** To verify if the Transaction history service correctly saves a transaction history whit a product associated. | | | | |
| **Class** | **Method** | **Stage** | **Input** | **Output** |
| Transaction-  HistoryService | save | saveTransaction-  HistorySetUp | ProductId= 01  TransactionId=01  quantity = 12  actualCost= 20  ahora= actual date | The product object is found.  The quantity is greater than 0, the actual cost is greater than 0 and the transaction date is the actual date.  They all are the same as the passed by parameter.  The transaction history is successfully saved. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective:** To verify if the Product service correctly edits a product whit a product category and a product subcategory associated. | | | | |
| **Class** | **Method** | **Stage** | **Input** | **Output** |
| Transaction-  HistoryService | edit | editTransaction-HistorySetUp | TransactionId=01  quantity = 0  actualCost= 0  ahora= null | The quantity is not greater than 0, the actual cost is not greater than 0 and the transaction date is not the actual date.  The transaction history is not successfully saved. |
| Transaction-  HistoryService | edit | editTransaction-HistorySetUp | TransactionId=01  quantity = 2  actualCost= 100  ahora= actual date | The quantity is greater than 0, the actual cost is greater than 0 and the transaction date is the actual date.  The transaction history is successfully saved. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective:** To verify if the Document service correctly saves a document whit a productDocument and a Product associated. | | | | |
| **Class** | **Method** | **Stage** | **Input** | **Output** |
| DocumentSerivce | save | saveDocument-  SetUp | ProductId= 01  DocumentId=01  productDocumentId=  fileName=”This is a file name”  fileExtension= “This is a test” | The product, productDocument and document objects are found.  The file name is greater than 4 characters.  The file extension is greater than 3 characters.  The Document object is saved successfully. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective:** To verify if the Document service correctly edits a document with a productDocument and a Product associated. | | | | |
| **Class** | **Method** | **Stage** | **Input** | **Output** |
| DocumentSerivce | edit | editProduct-  DocumentSetUp | correctFileName= null  correctExtensionFile=null | The file name is not greater than 4 characters.  The file extension is not greater than 3 characters.  The Document object is not edited successfully. |
| DocumentSerivce | edit | editProduct-  DocumentSetUp | correctFileName=”correct file name”  correctExtensionFile=”correct extension file” | The file name is greater than 4 characters.  The file extension is greater than 3 characters.  The Document object is edited successfully. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective:** To verify if the Product Vendor service correctly saves a product vendor whit a product, a Unit measure and an Employee associated. | | | | |
| **Class** | **Method** | **Stage** | **Input** | **Output** |
| ProductVendor-  Service | save | saveProductVendor-  SetUp | ProductId= 01  ProductVendorId=01  EmployeeId=01  maxorderqty=20  minotderqty=10  standardprice=”1000” | The product, productVendor and Employee objects are found.  The maximun value is greater than the minimun value.  The standard price is greater than 0.  The product vendor is saved successfully. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective:** To verify if the Product Vendor service correctly edits a product vendor whit a product, a Unit measure and an Employee associated. | | | | |
| **Class** | **Method** | **Stage** | **Input** | **Output** |
| ProductVendor-  Service | edit | editProductVendor-  SetUp | maxorderqty=5  minotderqty=10  standardprice=0 | The maximun value is not greater than the minimun value.  The standard price is not greater than 0.  The product vendor is not edited successfully. |
| ProductVendor-  Service | edit | editProductVendor-  SetUp | maxorderqty=10  minotderqty=5  standardprice=3 | The maximun value is greater than the minimun value.  The standard price is greater than 0.  The product vendor is edited successfully. |