Formato de documentación de pruebas Taller 1

Pruebas Unitarias (en cada numeral se evalúa 20% diseño (documento Excel o Word), 80% implementación de las pruebas unitarias con todos los casos relevantes).

5. Realizar las pruebas definiendo los mocks necesarios en cada caso y la posterior implementación acorde al requerimiento en cada punto siguiendo TDD. Debe incluir la información inicial para las pruebas que requieren información de otra(s) entidad(es) de tal forma que esta(s) se puedan encontrar:

|  |  |  |
| --- | --- | --- |
| Nombre | Clase | Escenario |
| SetupPR | producto(id, productName, endDate, startDate, weight, size) | Product(1,00001, 28/3/2022, 27/3/2022,3, 10) |

Casos:

a. (0.6) Proveer servicios para guardar y editar un *producto* escogiendo la categoría y subcategoría (debe previamente crearlas y poder asociarlas al producto) garantizando que tenga un número de producto, una fecha de inicio de venta menor a la fecha de fin, y tamaño y peso mayores a cero.

Producto (número de producto, fecha de inicio, fecha de fin, tamaño, peso)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Guardar un producto exitosamente , ATRIBUTOS DEL PRODUCTO: Producto (número de producto, fecha de inicio (dd/mm/aaaa) , fecha de fin, peso(long)),tamaño(long) | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
| ServiceUnitTesting | productEditTestInvalidProductNumber |  | Product(null) | El producto no se guardo, porque es un producto vacío |
| ServiceUnitTesting | productEditTestInvalidProductNumber |  | Product(1,00002, 28/3/2022, 27/3/2022,1, 10) | El producto se guardó exitosamente |
| ServiceUnitTesting | productEditTestInvalidProductNumber |  | Product(1,00002, 28/3/2022, 27/3/2022,1, -1) | El producto no se guarda ya que tamaño es invalido |
| ServiceUnitTesting | testSaveProductWeigthInvalid |  | Product(1,1111, 28/3/2022, 27/3/2022,-1, 10) | El producto no se guarda ya que peso es invalido |
| ServiceUnitTesting | testSaveProductDate() |  | Product(1,1111, 27/3/2022, 28/3/2022,-1, 10) | El producto no se guarda ya que la fecha es inválida. |
| ServiceUnitTesting | testSaveProductNameNull |  | Product(1,” ”, 27/3/2022, 28/3/2022,-1, 10) | El producto no se guarda ya que el nombre del producto es invalido. |
| ServiceUnitTesting | testSaveProductInexistentSubCategory | setupPR |  | El producto no se guarda ya que no hay una subcategoría existente. |
| ServiceUnitTesting | testSaveProductInexistentCategory | setupPR |  | El producto no se guarda ya que no hay una categoría existente. |
| ServiceUnitTesting | testSaveProductIdequals | setupPR |  | El producto no se guarda ya que el id se duplicaría y no puede quedar con el mismo id |
| ServiceUnitTesting | testSaveProductcategoryEmptyt() | setupPR |  | El producto no se guarda , porque la categoría está vacía. |
| ServiceUnitTesting | testSaveProductSubcategoryEmptyt() | setupPR |  | El producto no se guarda , porque la subcategoría está vacía. |

Editar

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Editar un producto  Atributos: Producto (número de producto, fecha de inicio (dd/mm/aaaa) , fecha de fin, peso(long)),tamaño(long) | | | |
| Clase | Métodos | Escenario | Entrada | Resultado |
| ServiceUnitTesting | productEditTestInvalidProductNumber | setupPR | productNumber(“ ”) | El nombre de producto a modificar no se modifica correctamente, porque es un dato vacío. |
| ServiceUnitTesting | productEditTestInvalidDate | setupPR | starDate(27/3/2022) | El producto no se pudo modificar porque no tiene una fecha válida |
| ServiceUnitTesting | productEditTestInvalidsize | setupPR | size(-10) | El producto no se puede modificar ya que el tamaño es invalido. |
| ServiceUnitTesting | productEditTestInvalidWeigth | setupPR | weigth(-10) | El producto no se puede modificar ya que el peso es invalido. |
| ServiceUnitTesting | testEditProductcategoryInexist | setupPR |  | No se edita el producto , ya que la categoría es inexistente |
| ServiceUnitTesting | testEditProductsubcategoryInexist | setupPR |  | No se edita el producto , ya que la sub categoria es inexistente |
| ServiceUnitTesting | testEditProductNull | setupPR |  | No edita el producto, ya que es null |
| ServiceUnitTesting | testEditProductFull | setupPR |  | Edita el producto exitosamente |

b. (0.6) Proveer servicios para guardar y editar una *ubicación*. Se debe validar que el nombre tenga al menos cinco caracteres, la disponibilidad esté entre 1 y 10 y la tasa de costo entre 0 y 1.

|  |  |  |
| --- | --- | --- |
| Nombre | Clase | Escenario |
| SetupLOC | Location | (1,12345,10,1) |
| LocationSetup2 | Location | (1,Cali0007,10,1) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Guardar una localización exitosamente , ATRIBUTOS DEL LOCATION: Ubicación(nombre(5 caracteres o más), disponibilidad (1,10),( tasa de costo entre 0 y 1) | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
| ServiceUnitTesting | testSaveLocationNull |  | Locationnull) | No se guarda la ubicación, debido a que es nula |
| ServiceUnitTesting | testSaveLocationFull |  | (1,00005,10,1) | Se guarda una ubicación exitosamente |
| ServiceUnitTesting | testSaveLocationInvalidName |  | (1,1234,1,1) | No se guarda una ubicación, debido a que el nombre es menor a 5 caracteres |
| ServiceUnitTesting | testSaveLocationInvalidAvailability |  | (2,12345,-1,1) | No se guarda, porque la disponibilidad es menor a 1. |
| ServiceUnitTesting | testSaveLocationInvalidAvailabilityMajor1 |  | (2,12345,11,1) | No se guarda, porque la disponibilidad es mayor a 10. |
| ServiceUnitTesting | testSaveLocationNullAvailability |  | (2,12345,null,1) | No se guarda, porque la disponibilidad es nula |
| ServiceUnitTesting | testSaveLocationInvalidCostRate |  | (2,12345,10,3) | No se guarda, porque el cost rate es mayor a 1 |
| ServiceUnitTesting | testSaveLocationInvalidCostRateMinor |  | (2,12345,10,-1) | No se guarda, porque el cost rate es menor a 0 |
| ServiceUnitTesting | testSaveLocationInvalidCostRateNull |  | (2,12345,10,null) | No se guarda, porque el cost rate es nulo. |
| ServiceUnitTesting | testSaveduplicateLocationid | Setup1 | (1,12345,10,1) | No se guarda, porque el id está duplicado. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Editar una localización exitosamente , ATRIBUTOS DEL LOCATION: Ubicación(nombre(5 caracteres o más), disponibilidad (1,10),( tasa de costo entre 0 y 1) | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
| ServiceUnitTesting | testEditLocationNull | LocationSetup2 | Locationnull) | No se modifica la ubicación, debido a que es nula |
| ServiceUnitTesting | testEditLocationFull | LocationSetup2 | (1,Cali0007,10,1) | Se modifica una ubicación exitosamente |
| ServiceUnitTesting | tesEditLocationInvalidName | LocationSetup2 | (1,1234,10,1) | No se gmodifica una ubicación, debido a que el nombre es menor a 5 caracteres |
| ServiceUnitTesting | testEditLocationInvalidAvailability | LocationSetup2 | (1,Cali0007,-1,1) | No se modifica, porque la disponibilidad es menor a 1. |
| ServiceUnitTesting | testEditLocationInvalidAvailabilityMajor1 | LocationSetup2 | (1,Cali0007,11,1) | No se modifica, porque la disponibilidad es mayor a 10. |
| ServiceUnitTesting | testEditLocationNullAvailability | LocationSetup2 | (1,Cali0007,null,1) | No se modifica, porque la disponibilidad es nula |
| ServiceUnitTesting | testEditLocationInvalidCostRate | LocationSetup2 | (1,Cali0007,10,3) | No se modifica, porque el cost rate es mayor a 1 |
| ServiceUnitTesting | testEditLocationInvalidCostRateMinor | LocationSetup2 | (1,Cali0007,10,-1) | No se modifica, porque el cost rate es menor a 0 |
| ServiceUnitTesting | testEditLocationInvalidCostRateNull | LocationSetup2 | (1,Cali0007,10,nullll) | No se modifica , porque el cost rate es nulo. |
| ServiceUnitTesting | testEditLocationemptyid | LocationSetup2 | (,Cali0007,10,1) | No se modifica , porque el id esta vacio, |
| ServiceUnitTesting | ttestEditLocationnullid( | LocationSetup2 | (null,Cali0007,10,1) | No se guarda, porque el id es nulo |

c. (0.6) Proveer servicios para guardar y editar *el costo histórico del producto* asociado a un producto. Se debe validar que la fecha de finalización no sea mayor a la actual y el costo estándar no sea negativo. Se debe validar que exista el producto (se pasa el identificador y se busca).

|  |  |  |
| --- | --- | --- |
| Nombre | Clase | Escenario |
| SetupPHC | HistoricCost | (1,1,10,28/03/2022) |
| Setup 2 | HistoricCost | (1,1,10,28/03/2022) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Guardar un costo histórico  Atributos: PHC\_ID, PR\_ID, COSTSTANDAR, ENDDATE | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
| ServiceUnitTesting | testSaveProductCostHistoryNull |  | null | No se guarda porque tiene como entrada datos nulos. |
| ServiceUnitTesting | testSaveProductCostHistoryfull | SetupPHC | (3,3,30,28/03/2022) | Se guarda correctamente el product cost historic |
| ServiceUnitTesting | testSaveProductCostHistoryCostEstandarMinor | SetupPHC | (1,1,-10,28/03/2022) | No se guarda correctamente porque el coste estandar es menor a 0. |
| ServiceUnitTesting | testSaveProductCostHistoryCostEstandarNull | SetupPHC | (1,1,null,28/03/2022) | No se guarda correctamente porque el coste estándar es nulo. |
| ServiceUnitTesting | estSaveProductCostHistoryDate | SetupPHC | (1,1,null,27/03/2022) | No se guarda correctamente porque la fecha de fin es menor a la de inicio |
| ServiceUnitTesting | testSaveProductCostHistoryDuplicatedId | SetupPHC | (1,1,10,28/03/2022) | No se guarda porque el id está duplicado |
| ServiceUnitTesting | testSaveProductCostHistoryProductIDempty | SetupPHC | (1,” ”,10,28/03/2022) | No se guarda porque porque el id de producto está vacío |
| ServiceUnitTesting | testSaveProductCostHistoryProductIDnull | SetupPHC | (1,null,10,28/03/2022) | No se guarda porque porque el id de producto está nulo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Editar un costo histórico  Atributos: PHC\_ID, PR\_ID, COSTSTANDAR, ENDDATE | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
| ServiceUnitTesting | testEditProductCostHistoryNull |  | null | No se modifica, ya que le entran datos nulos. |
| ServiceUnitTesting | testEditProductCostHistoryfull | SetupPHC |  | Se modifica y guarda correctamente el product cost historic |
| ServiceUnitTesting | testEditProductCostHistoryCostEstandarMinor | SetupPHC | (1,1,-10,28/03/2022) | No se modifica correctamente porque el coste estándar es menor a 0. |
| ServiceUnitTesting | testEditProductCostHistoryCostEstandarNull | SetupPHC | (1,1,null,28/03/2022) | No se modifica correctamente porque el coste estándar es nulo. |
| ServiceUnitTesting | testEditProductCostHistoryDate | SetupPHC | (1,1,null,27/03/2022) | No se modifica correctamente porque la fecha de fin es menor a la de inicio |
| ServiceUnitTesting | testEditProductCostHistoryDuplicatedId | SetupPHC | (1,1,10,28/03/2022) | No se modifica porque el id está duplicado |
| ServiceUnitTesting | testEditProductCostHistoryProductIDempty( | SetupPHC | (1,” ”,10,28/03/2022) | No se modifica porque porque el id de producto está vacío |
| ServiceUnitTesting | testEditProductCostHistoryProductIDnull | SetupPHC | (1,null,10,28/03/2022) | No se modifica porque porque el id de producto está nulo |

d. (0.7) Proveer servicios para guardar y editar *un inventario de producto* asociado a una ubicación y un producto. Se debe validar que existan las dos llaves foráneas de guardarlo (se reciben los identificadores y se buscan). Se debe garantizar que la cantidad no sea inferior a cero.

|  |  |  |
| --- | --- | --- |
| Nombre | Clase | Escenario |
| SetupPIR | ProductInventory | (1,1,100) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Guardar un product inventory  Atributos: PIR\_ID,QUANTITY | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
| ServiceUnitTesting | testSaveProductCostHistoryNull |  | null | No se guarda correctamente ya que es un valor nulo. |
| ServiceUnitTesting | testSaveProductInventoryfull | SetupPIR | (1,1,100) | Se guarda correctamente el product cost historic |
| ServiceUnitTesting | productInventorySaveTestNullLocationId | SetupPIR | (1,null, 100) | No se guarda correctamente porque el id ubicación es null |
| ServiceUnitTesting | productInventorySaveTestNullProductId | SetupPIR | (null,1, 100) | No se guarda correctamente porque el id de product Inventory es null |
| ServiceUnitTesting | testSaveProductInventoryDuplicateId | SetupPIR | (1,1,100) | No se guarda correctamente porque el id está en uso. |
| ServiceUnitTesting | testSaveProductInventoryQuantityMinor | SetupPIR | (1,1,-1) | No se guarda porque la cantidad es menor a 0 |
| ServiceUnitTesting | testSaveProductInventoryQuantitynull | SetupPIR | 1,1,-1) | No se guarda ya que la cantidad es null |
| ServiceUnitTesting | testSavePIRProductinexist | SetupPIR | (,1,-1) | No se guarda porque porque el id de producto Inventory no existe |
| ServiceUnitTesting | testSavePIRLocationnexist | SetupPIR | (1,””,-1) | No se guarda porque porque el id de location no existe |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Editar un product inventory  Atributos: PIR\_ID,QUANTITY | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
|  | testEditProductInventoryNull |  | null | No se edita correctamente porque esta nulo |
|  | testEditProductInventoryFull | SetupPIR | (1,1,100) | El product inventory se edita correctamente. |
|  | testEditproductInventoryNullLocationId | SetupPIR | (1,1,100) | No se guarda correctamente porque el id ubicación es null |
|  | testEditproductInventoryNullProductId | SetupPIR | (1,1,100) | No se guarda correctamente porque el id de product Inventory es null |
|  | testEditProductInventoryDuplicateId | SetupPIR | (1,1,100) | No se modifica correctamente porque el id está en uso. |
|  | testEditPIRProductinexist | SetupPIR | (,1,100) | No se modifica porque porque el id de producto Inventory no existe |
|  | testEditPIRLocationnexist() | SetupPIR | (1,100) | No se modifica porque porque el id de location no existe |
|  | testEditProductInventoryQuantityMinor | SetupPIR | (1,1,-1) | No se modifica correctamente porque la cantidad es negativa. |
|  | testEditProductInventoryQuantitynull( | SetupPIR | (1,1,null) | No se modifica correctamente porque la cantidad es null. |

Pruebas de Integración (en cada numeral se evalúa 10% diseño (documento Excel o Word), 50% implementación de las pruebas de integración con todos los casos relevantes, 40% implementación del servicio solicitado).

1. Realizar las pruebas de integración, y la posterior implementación acorde a los requerimientos del numeral anterior. Debe conservar las pruebas unitarias, puede realizar una copia de la clase de pruebas unitarias para realizar las pruebas de integración, sin embargo, no van a ser las mismas pruebas en ambos casos.
2. a. (0.6) producto
3. b. (0.6) ubicación
4. c. (0.6) costo histórico de producto
5. d. (0.7) inventario de producto

|  |  |  |
| --- | --- | --- |
| Nombre | Clase | Escenario |
| SetupPR | producto(id, productName, endDate, startDate, weight, size) | Product(1,00001, 28/3/2022, 27/3/2022,3, 10) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Pruebas de integración Guardar un producto | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
|  | Test Save Integration Product |  | Product(1,00002, 28/3/2022, 27/3/2022,1, 10) | El producto se guarda correctamente verificando sus valores, tanto categoría, como subcategoria. |
|  | Test Edit Integration Product | SetupPR | Product(1,00002, 28/3/2022, 27/3/2022,1, 20) | Se edita el size de producto y se guarda correctamente |

|  |  |  |
| --- | --- | --- |
| Nombre | Clase | Escenario |
| SetupLOC | Location | (1,12345,10,1) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Pruebas de integración de localización para guardar y editar un producto | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
|  | Test Save Integration Location |  | (1,00005,10,1) | La ubicación se guarda correctamente, con los datos validos. |
|  | Test Edit Integration Location | SetupLOC | (1,00005,10,0) | Se edita el cost rate de producto y se guarda correctamente |

|  |  |  |
| --- | --- | --- |
| Nombre | Clase | Escenario |
| SetupPHC | HistoricCost | (1,1,10,28/03/2022) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Pruebas de integración de product cost historic guardado y editar | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
|  | Test Save Integration PCH |  | (1,1,10,28/03/2022) | El producto cost historic se guarda correctamente, validando el id de producto. |
|  | Test Edit Integration PCH | SetupPHC | (1,1,20,28/03/2022) | El producto cost historic se edita correctamente. |

|  |  |  |
| --- | --- | --- |
| Nombre | Clase | Escenario |
| SetupPIR | ProductInventory | (1,1,100) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Objetivo | Pruebas de integración de product inventory | | | |
| Clase/nombre | Métodos | Escenario | Entrada | Resultado |
|  | Test Save Integration PIR |  | (1,1,100) | El product inventory se guarda correctamente teniendo en cuenta el producto y location. |
|  | Test Edit Integration PIR | SetupPIR | (1,1,200) | La cantidad del producto se modifica correctamente |