# INFORME PRÁCTICA 4: PRUEBAS UNITARIAS DE SOFTWARE

## Introducción

<Un párrafo describiendo el objetivo del documento.>

<Breve descripción del proceso de pruebas general que se ha aplicado: fases, técnicas de prueba aplicadas, criterios de cobertura elegidos, herramientas utilizadas para cada tipo de prueba, etc.>

En las siguientes secciones (2, 3 y 4), explicar el detalle de la aplicación de las técnicas de prueba aplicadas para cada una de las clases involucradas y los resultados obtenidos.

* En el caso de pruebas de caja negra, incluir una tabla con las clases de equivalencia identificadas, los valores interesantes seleccionados y los casos de prueba definidos a partir de ellos (tanto válidos como no válidos).
* Para pruebas de caja blanca, criterios de cobertura aplicados y cobertura conseguida. Si ha sido necesario ampliar el conjunto de casos de prueba definidos en el paso anterior, indicar cuáles son los casos añadidos.
* Para pruebas de integración, especificación de los casos de prueba definidos (si se ha aplicado clases de equivalencia, exponerlas) y enumeración de los que se han implementado.
* En la medida de lo posible, listar los errores que se han detectado en el proceso de pruebas.

## Proceso de pruebas unitarias de la clase Seguro

* Método precio()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Clases válidas** | **Valores válidos** | **Clases no válidas** | **Valores no válidos** |
| *Fecha*  *Último*  *Siniestro* | (actual – 1 año, actual]  (actual – 3 años, actual – 1 año]  (-inf, actual – 3 años] | Actual  Actual – 3 meses  Actual – 364 días  Actual – 1 año  Actual – 2 años  Actual – 2 años y 364 días  Actual – 3 años  Actual – 4 años | > actual  Null | Actual + 1 día  Actual + 1 año  Null |
| *potenciaCV* | [0,90)  [90,110]  [111, inf) | 0  45  89  90  100  110  111  200 | < 0 | -1  -30 |
| *Cobertura* | TERCEROS  TODORIESGO  TERCEROSLUNAS | TERCEROS  TODORIESGO  TERCEROSLUNAS | No {TERCEROS,  TODORIESGO,  TERCEROSLUNAS}  Null | No {TERCEROS,  TODORIESGO,  TERCEROSLUNAS}  Null |
| *Minusvalía* | Verdadero  Falso | Verdadero  Falso | - | - |

|  |  |
| --- | --- |
| **Casos de prueba válidos** | **Casos de prueba no válidos** |
| * Fecha = Actual; Potencia = 0; Cobertura = TERCEROS; Minusvalía = Verdadero; Resultado esperado = 450 * Fecha = Actual – 3 meses; Potencia = 45; Cobertura=TODORIESGO; Minusvalía=falso; Resultado esperado = 1200 * Fecha = Actual – 364 días; Potencia = 89; Cobertura = TERCEROSLUNA; Minusvalía=falso; Resultado esperado = 800 * Fecha = Actual – 1 año; Potencia = 90; Cobertura = TERCEROS; Minusvalía=falso; Resultado esperado = 470 * Fecha = Actual – 2 años; Potencia = 100; Cobertura = TODORIESGO; Minusvalía = verdadero; Resultado esperado = 825 * Fecha = Actual – 2 años y 364 días; Potencia = 110; Cobertura = TODORIESGO; Minusvalía = verdadero; Resultado esperado = 825 * Fecha = Actual – 3 años; Potencia = 111; Cobertura = TERCEROS; Minusvalía = verdadero; Resultado esperado = 360 * Fecha = Actual – 4 años; Potencia = 200; Cobertura=TERCEROSLUNA; Minusvalía= verdadero; Resultado esperado = 540 | * Fecha= Actual +1 día; Potencia=0; Cobertura=TODORIESGO; Minusvalía=falso; Resultado esperado = DatoIncorrectoException * Fecha = Actual +1 año; Potencia = 45; Cobertura= TERCEROS; Minusvalía=verdadero; Resultado esperado = DatoIncorrectoException * Fecha = Null; Potencia = 45; Cobertura= TODORIESGO; Minusvalía = verdadero; Resultado esperado = DatoIncorrectoException * Fecha = Actual; Potencia = -1; Cobertura = TERCEROS; Minusvalía = falso; Resultado esperado = DatoIncorrectoException * Fecha = Actual; Potencia = -30; Cobertura = TERCEROS; Minusvalía = falso; Resultado esperado = DatoIncorrectoException * Fecha = Actual; Potencia = 45; Cobertura = Null; Minusvalía = falso; Resultado esperado = DatoIncorrectoException |

* Método Constructor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Clases válidas** | **Valores válidos** | **Clases no válidas** | **Valores no válidos** |
| *Fecha*  *Último*  *Siniestro* | <= Actual | Actual – 1 mes | > Actual  Null | Actual + 1 día  Null |
| *potenciaCV* | >= 0 | 0  30 | < 0 | -1  -30 |
| *Cobertura* | TERCEROS  TODORIESGO  TERCEROSLUNAS | TERCEROS  TODORIESGO  TERCEROSLUNAS | No {TERCEROS,  TODORIESGO,  TERCEROSLUNAS}  Null | No {TERCEROS,  TODORIESGO,  TERCEROSLUNAS}  Null |
| *Minusvalía* | Verdadero  Falso | Verdadero  Falso | - | - |

## Proceso de pruebas de integración de la clase SegurosGUI con la clase Seguro

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Clases válidas** | **Valores válidos** | **Clases no válidas** | **Valores no válidos** |
| *Fecha*  *Último*  *Siniestro* | <= Actual | Actual – 1 mes | > Actual  Fecha con formato incorrecto | Actual + 1 día  dd/mm/aaaa |
| *potenciaCV* | >= 0 | 0  30 | < 0  Potencia con formato incorrecto | -1  75a |
| *Cobertura* | TERCEROS  TODORIESGO  TERCEROSLUNAS | TERCEROS  TODORIESGO  TERCEROSLUNAS | - | - |
| *Minusvalía* | Verdadero  Falso | Verdadero  Falso | - | - |

* Fecha = 20/01/2000; Potencia = 60; Cobertura = TERCEROS; Minusvalía = Falso; Resultado esperado = 400
* Fecha = Actual + 1 año; Potencia = 60; Cobertura = TERCEROS; Minusvalía = Falso; Resultado esperado = ¡Dato de entrada erróneo!
* Fecha = 02/mm/2000; Potencia = 60; Cobertura = TERCEROS; Minusvalía = Falso; Resultado esperado = La fecha no se pudo parsear
* Fecha = 20/02/2000; Potencia = 60a; Cobertura = TERCEROS; Minusvalía = Falso; Resultado esperado = ¡Dato de entrada erróneo!
* Fecha = 20/02/2000; Potencia = -60; Cobertura = TERCEROS; Minusvalía = Falso; Resultado esperado = ¡Dato de entrada erróneo!

### Errores

* Se ha corregido el error de selección de tipo de seguro, esto se debía a que el combo box asociado carecía de nombre
* Se encontró un error de FEST en el que el carácter “/” no se puede presionar, se ha optado por utilizar en carácter “-” en vez de este dentro de la aplicación
* Se ha detectado un error en donde se lanzaba una excepción no capturada al dar un valor de entrada con formato no válido a potencia, en vez de mostrar el mensaje de error “¡Dato de entrada erróneo!”

## Proceso de pruebas de la clase ListaOrdenada

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Clases válidas** | **Valores válidos** | **Clases no válidas** | **Valores no válidos** |
| *Get* | Índice válido en lista con elementos  Índice válido en lista con un elemento | (1) en [1, 2, 3]  (0) en [1] | Índice no válido  Índice en lista sin elementos | (-6) en [1, 2, 3]  (6) en [1, 2, 3]  (1) en [] |
| *Add* | Añade en lista vacía  Añade en lista con un elemento  Añade en lista con elementos  Añade elemento en medio de la lista | (1) en []  (2) en [1]  (3) en [1, 2]  (2) en [1, 3] | - | - |
| *Remove* | Índice válido en lista con elementos  Índice válido en lista con un elemento | (2) en [1, 2, 3]  (0) en [2] | Índice no válido  Índice en lista sin elementos | (-6) en [1, 2, 3]  (6) en [1, 2, 3]  (1) en [] |
| *size* | Lista vacía  Lista no vacía | []  [1, 2] | - | - |
| *clear* | Lista vacía  Lista no vacía | []  [1, 2] | - | - |

### Errores tras la ejecución con el jar

* Se ha detectado un error en el método remove cuando se intenta eliminar un valor en una lista con un elemento
* Se ha detectado un error en el método clear cuando la lista no está vacía

Luego de la ejecución de las pruebas con el jar, se procedió a ejecutar las pruebas con el código proporcionado por el docente, dejando comentada la dependencia Maven del jar.