Ejemplo de plan de pruebas de unidad.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **testagregarSoftwareCompleto** | | | | |
| Entradas | Condiciones de entrada | Salidas esperadas | Condiciones de salida esperadas | Resultado |
| idSoftware= “S001”  nombre = “sait”  origen = “comprado”  observaciones= “sin fallas bla bla bla”  fechaAdquisiscion= “2018-02-02”  tipoSoftware= “administrativo”  marca= “Sait”  requiereActualizacion = true  versión= 1.7  disponible=true  sistemaOperativo= “windows”  idioma= “español” | El idSoftware no se ha agregado previamente | True | Se registra en la base de datos el nuevo software. | satisfactorio |
| **testactualizarSoftwareCompleto** | | | | |
| idSoftware= “S001”  nombre = “word”  origen = “comprado”  observaciones= “sin fallas bla bla bla”  fechaAdquisiscion= “2018-02-02”  tipoSoftware= “administrativo”  marca= “office”  requiereActualizacion = true  versión= 7.0  disponible=true  sistemaOperativo= “windows”  idioma= “español” | El software si existe | true | Se guardan los cambios | denegado |
| testObtenerListaSoftware | | | | |
|  |  | Software con:  idSoftware= “S001”  nombre = “sait”  origen = “comprado”  observaciones= “sin fallas bla bla bla”  fechaAdquisiscion= “2018-02-02”  tipoSoftware= “administrativo”  marca= “Sait”  requiereActualizacion = true  versión= 1.7  disponible=true  sistemaOperativo= “windows”  idioma= “español” | Regresa los software guardados | satisfactorio |
| testBuscarSoftware | | | | |
| IDSOFTWARE\_ESPERADO = "S001"; | Existe el software | True = softwareEsperado, softwareObtenido | Regresa el software pedido | satisfactorio |
| testEliminarSoftwareCorrectamente | | | | |
| Software con id = S001 | El registro existe | true | Pone el software disponible = false | satisfactorio |
|  | | | | |
| Pathfile= ‘/tmp/pruebas/datasets/’  Archivo=’australian.arff’  Overwrite= True | El archivo no se ha agregado previamente | Resultado=  False, ‘File must be a .arff’ |  | satisfactorio |
| **Función strategy** | | | | |
| Entradas | Condiciones de entrada | Salidas esperadas | Condiciones de salida esperadas | Resultado |
| Pathfile= ‘/tmp/pruebas/’  Archivo=’prueba.zip’  Overwrite= False | El archivo no se ha agregado previamente | Resultado=  True, | El archivo se guarda en el Pathfile | satisfactorio |
| Pathfile= ‘/tmp/pruebas/’  Archivo=’australian.arff’  Overwrite= False | El archivo no se ha agregado previamente | Resultado=  False,’File must be a .zip’ |  | satisfactorio |
| Pathfile= ‘/tmp/pruebas/’  Archivo=’pruebaBad.zip’  Overwrite= False | El archivo no se ha agregado previamente | Resultado=  False,’The file is not valid’ |  | satisfactorio |
| Pathfile= ‘/tmp/pruebas/strategy/’  Archivo=’prueba.zip’  Overwrite= False | El archivo no se ha agregado previamente | Resultado=  False,  ’Compilation errors’ |  | satisfactorio |
| Pathfile= ‘/tmp/pruebas/strategy/’  Archivo=’australian.arff’  Overwrite= False | El archivo no se ha agregado previamente | Resultado=  False, ‘File must be a .zip’ |  | satisfactorio |
| Pathfile=‘/tmp/pruebas/’  Archivo=’prueba.zip’  Overwrite= False | El archivo se ha agregado previamente | Resultado=  False,’Existing file:prueba.zip’ |  | satisfactorio |
| Pathfile= ‘/tmp/pruebas/’  Archivo=’prueba.zip’  Overwrite= True | El archivo se ha agregado previamente | Resultado=  True, | El archivo se actualiza en el Pathfile | satisfactorio |
| Pathfile= ‘/tmp/pruebas/’  Archivo=’australian.arff’  Overwrite= True | El archivo no se ha agregado previamente | Resultado=  False,’File must be a .zip’ |  | satisfactorio |
| Pathfile= ‘/tmp/pruebas/’  Archivo=’pruebaBad.zip’  Overwrite= True | El archivo no se ha agregado previamente | Resultado=  False,’The file is not valid’ |  | satisfactorio |
| Pathfile= ‘/tmp/pruebas/strategy/’  Archivo=’prueba.zip’  Overwrite= True | El archivo no se ha agregado previamente | Resultado=  False,  ’Compilation errors’ |  | satisfactorio |
| Pathfile= ‘/tmp/pruebas/strategy/’  Archivo=’australian.arff’  Overwrite= True | El archivo no se ha agregado previamente | Resultado=  False, ‘File must be a .zip’ |  | satisfactorio |
| **Función saveArtifact** | | | | |
| Entradas | Condiciones de entrada | Salidas esperadas | Condiciones de salida esperadas | Resultado |
| Pathfile= ‘/tmp/pruebas/’  Archivo=’artefacto.zip’ | El archivo no se ha agregado previamente | Resultado=  True, | El archivo se guarda en el Pathfile | satisfactorio |
| Pathfile= ‘/tmp/pruebas/’  Archivo=’australian.arff’ | El archivo no se ha agregado previamente | Resultado=  False, ‘File errors’ |  | satisfactorio |
| Pathfile= ‘/tmp/pruebas/artifact/’  Archivo=’artefacto.zip’ | El archivo no se ha agregado previamente | Resultado=  False, ‘Compilation errors’ |  | satisfactorio |
| Pathfile= ‘/tmp/pruebas/artifact/’  Archivo=’australian.arff’ | El archivo no se ha agregado previamente | Resultado=  False, ‘File errors’ |  | satisfactorio |
| **Función xmlStrategy** | | | | |
| Entradas | Condiciones de entrada | Salidas esperadas | Condiciones de salida esperadas | Resultado |
| Pathfile= ‘/home/chise/Documentos/JaaDDM-1/JacaDDMInterfaz/pruebas/Unidad/’ | Existe un solo archivo XML. | Resultado=  True, ParallelCounterGPU, agentes, parámetros | El archivo se lee y devuelve los Agentes y parámetros definidos en el archivo XML de la estrategia. | satisfactorio |
| Pathfile= ‘/tmp/strategia/ejemplo/pruebas/’ | El Pathfile es erróneo porque no existe. | Resultado=  False, ‘Compilation erros’ |  | satisfactorio |
| Pathfile= ‘Pruebas/Unidad/parallelCounterGP/’ | El Pathfile es erróneo porque no existe ningún archivo XML en ese directorio. | Resultado=  False, ‘strategy errors’ |  | satisfactorio |
| **Función xmlExperiment** | | | | |
| Entradas | Condiciones de entrada | Salidas esperadas | Condiciones de salida esperadas | Resultado |
| Pathfile= ‘/home/chise/Documentos/JaaDDM-1/JacaDDMInterfaz/pruebas/Unidad/a0.xml’ | El pathfile es un archivo XML que existe y es correcto. | Resultado=  True, nodes, strategy, typeEvaluation, folds, repetitions, evaluation, count, strategia1 | El archivo se lee y devuelve los parámetros definidos en el archivo XML del experimento. | satisfactorio |
| Pathfile= ‘/home/chise/Documentos/JaaDDM-1/JacaDDMInterfaz/pruebas/Unidad/a1.xml’ | El pathfile es un archivo XML que no existe. | Resultado=  False, ‘Experiment errors’ |  | satisfactorio |
| Pathfile= ‘/home/chise/Documentos/JaaDDM-1/JacaDDMInterfaz/pruebas/Unidad/’ | El pathfile no es un archivo, es un directorio. | Resultado=  False, ‘Compilation erros’ |  | satisfactorio |
| Pathfile= ‘/home/chise/Documentos/JaaDDM-1/JacaDDMInterfaz/pruebas/Unidad/a2.xml’ | El pathfile es un archivo XML que existe pero contiene el path de la estrategia incorrecto. | Resultado=  False, ‘Files errors’ |  | satisfactorio |