

## PRUEBAS UNITARIAS

**correctTopic:** Se crea con el fin de testear la provisión de servicios para guardar y editar un tema sin ninguna asociación, pero garantizando que la cantidad de Sprints y grupos sea mayor a cero.

**correctGame:** Se crea con el fin de testear la provisión de servicios para guardar y editar un juego sin ninguna asociación, pero garantizando que la cantidad de Sprints y grupos sea mayor a cero. En caso de recibir un tema, se debe validar que este exista antes de guardarlo (del tema se recibe el identificador y se busca).

**correctStory:** Se crea con el fin de testear la provisión de servicios para guardar y editar una Historia sin ninguna asociación adicional pero asociada a un juego. Debe garantizar que el valor de negocio, el sprint inicial y la prioridad sean mayores que cero. Se debe validar que este exista el juego antes de guardarlo (del juego se recibe el identificador y se busca).

### TopicAUnitTest:

Método	Escenario	Entrada	Salida
<b>saveNullTopic()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	Null.	No se realizó el guardado con éxito ya que es un objeto nulo.
<b>saveValidTopic()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "CorrectTopic" como un escenario ideal donde el tema existe y sus demás parámetros son mayores a cero.	"CorrectTopic" se ha guardado con éxito.

<b>saveTopicWithoutGroups()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	"modifiedTopic" como ejemplo de la modificación de un tema sin grupos.	No se realizó la operación con éxito. No se ha guardado el objeto.
<b>saveTopicWithoutSprints()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	"modifiedTopic" como ejemplo de la modificación de un tema sin sprints.	No se realizó la operación con éxito. No se ha guardado el objeto.
<b>saveTopicWithoutGroupsAndSprints()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	"modifiedTopic" como ejemplo de la modificación de un tema sin sprints ni grupos.	No se realizó la operación con éxito. No se ha guardado el objeto.
<b>modifyTopic()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "CorrectTopic" como un escenario ideal para posteriormente ser modificado.	Se ha modificado a "Modified" con éxito.
<b>modifyExistingTopicWithWrongValues()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "CorrectTopic" como un escenario ideal, sin embargo, este no posee grupos ni sprints	No se realizó la operación con éxito. No se modificó el objeto.

<b>modifyNonExistingTopic()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "CorrectTopic" como un escenario ideal, sin embargo, este no posee la id correspondiente a la BD.	No se realizó la operación con éxito. No se modificó el objeto.
---------------------------------	---	--	---

#### GameBUnitTest:

Método	Escenario	Entrada	Salida
<b>saveNullGame()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	Null.	No se realizó el guardado con éxito ya que es un objeto nulo.
<b>saveGameWithoutTopic()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "CorrectGame" como un escenario ideal donde el tema existe y sus demás parámetros son mayores a cero. No tiene asociaciones con temas.	Se realizó la operación con éxito. El objeto ha sido guardado.
<b>saveGameWithExistingTopic()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "CorrectGame" como un escenario ideal donde el tema existe y sus demás parámetros son mayores a cero. Posee a "CorrectTopic" como asociación.	El objeto se ha guardado con éxito realizando las verificaciones correspondientes para validar la operación.
<b>saveGameWithNonExistingTopic()</b>	Se realiza un llamado al	Se utiliza a "CorrectGame" como	No se ha guardado con

	servicio “save” teniendo en cuenta la entrada para el caso específico.	un escenario ideal, sin embargo, posee a “CorrectTopic” como asociación errónea.	éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>saveInvalidGameWithoutTopic()</b>	Se realiza un llamado al servicio “save” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “CorrectGame” como un escenario ideal, sin embargo, el id no corresponde a la BD. No posee temas asociados.	No se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>saveInvalidGameWithExistingTopic()</b>	Se realiza un llamado al servicio “save” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “CorrectGame” como un escenario ideal, sin embargo, el id no corresponde a la BD. Posee a “correctTopic” como asociado.	No se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>saveInvalidGameWithNonExistingTopic()</b>	Se realiza un llamado al servicio “save” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “CorrectGame” como un escenario ideal, sin embargo, el id no corresponde a la BD. Posee a “correctTopic” como asociado erróneo.	No se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>modifyExistingGames()</b>	Se realiza un llamado al servicio “modify” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctGame” como un escenario ideal para posteriormente ser editado.	Se ha modificado a “modified” con éxito.
<b>modifyGameWithWrongValues()</b>	Se realiza un llamado al	Se utiliza a “correctGame” con	No se ha modificado con

	servicio “modify” teniendo en cuenta la entrada para el caso específico.	sus parámetros modificados establecidos en 0.	éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>modifyGameWithExistingTopic()</b>	Se realiza un llamado al servicio “modify” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctGame” como escenario ideal. Se encuentra asociado con “correctTopic”	Se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>modifyGameWithNonExistingTopic()</b>	Se realiza un llamado al servicio “modify” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctGame” como escenario ideal. Se encuentra asociado con una edición errónea de “correctTopic”	No se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>modifyNonExistingGame()</b>	Se realiza un llamado al servicio “modify” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctGame” como escenario ideal, sin embargo, posee una id no perteneciente a la BD.	No se ha modificado a “The game edited doesnt exist” con éxito tras realizar las verificaciones correspondientes para validar la operación.

### StoryCUnitTest:

Método	Escenario	Entrada	Salida
<b>saveNullStory()</b>	Se realiza un llamado al	Null	No se realizó el guardado con

	servicio “save” teniendo en cuenta la entrada para el caso específico.		éxito ya que es un objeto nulo.
<b>saveStoryWithExistingGame()</b>	Se realiza un llamado al servicio “save” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctStory” como escenario ideal con “correctGame” como asociación.	Se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>saveStoryWithNonExistingGame()</b>	Se realiza un llamado al servicio “save” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctStory” como escenario ideal con “correctGame” como asociación errónea.	No se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>saveInvalidStoryWithExistingGame()</b>	Se realiza un llamado al servicio “save” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctStory” como escenario ideal, sin embargo, no tiene sprints. Posee “correctGame” como asociación.	No se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>saveInvalidStoryWithNonExistingGame()</b>	Se realiza un llamado al servicio “save” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctStory”, sin sprints. Posee “correctGame” como asociación errónea.	No se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>modifyNonExistingStory()</b>	Se realiza un llamado al servicio	Se utiliza a “correctStory”, como escenario ideal, con	No se ha modificado con éxito a “modified”

	“modify” teniendo en cuenta la entrada para el caso específico.	una asociacion a “correctGame”	tras realizar las verificaciones correspondientes para validar la operación.
<b>modifyExistingStoryWithWrongValues()</b>	Se realiza un llamado al servicio “modify” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctStory”, como escenario ideal, con un valor de negocio de 0 y una asociación a “correctGame”	No se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>modifyExistingStory()</b>	Se realiza un llamado al servicio “modify” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctStory”, como escenario ideal, con sus parametros correctos y una asociación a “correctGame”	Se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>modifyStoryWithExistingGame()</b>	Se realiza un llamado al servicio “modify” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctStory”, como escenario ideal, con sus parametros correctos y una asociación a “correctGame”	Se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.
<b>modifyStoryWithNonExistingGame()</b>	Se realiza un llamado al servicio “modify” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctStory”, como escenario ideal, con sus parametros correctos y una asociación a “correctGame” erronea	No se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación.

## SaveGameDUnitTest:

Método	Escenario	Entrada	Salida
<b>guardarJuego2()</b>		<ul style="list-style-type: none"><li>• Se utiliza a "correctGame" como una asociación con "correctTopic"</li><li>• "correctTopic" posee con "correctStory"</li><li>• "correctStory" posee con "time1" y "time2"</li></ul>	Se ha guardado con éxito el juego tras realizar las verificaciones correspondientes para validar la operación, teniendo en cuenta las copias asociadas producto del funcionamiento de los repositorios como fuente de verificación.

## PRUEBAS DE INTEGRACIÓN

**correctTopic:** Se crea con el fin de testear la provisión de servicios para guardar y editar un tema sin ninguna asociación, pero garantizando que la cantidad de Sprints y grupos sea mayor a cero.

**correctGame:** Se crea con el fin de testear la provisión de servicios para guardar y editar un juego sin ninguna asociación, pero garantizando que la cantidad de Sprints y grupos sea mayor a cero. En caso de recibir un tema, se debe validar que este exista antes de guardarlo (del tema se recibe el identificador y se busca).

**correctStory:** Se crea con el fin de testear la provisión de servicios para guardar y editar una Historia sin ninguna asociación adicional pero asociada a un juego. Debe garantizar que el valor de negocio, el sprint inicial y la prioridad sean mayores que cero. Se debe validar que este exista el juego antes de guardarlo (del juego se recibe el identificador y se busca).



### TopicAIntegrationTest:

Método	Escenario	Entrada	Salida
<b>saveTopic()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctTopic", como escenario ideal, con sus parámetros correctos	Se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación
<b>modifyExistingTopic()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctTopic", como escenario ideal, con sus parámetros correctos	Se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación
<b>modifyNonExistingTopic()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctTopic", como escenario ideal, con su id no correspondiente a la BD.	No se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación

### GameBIntegrationTest:

Método	Escenario	Entrada	Salida
<b>saveGameWithoutTopic()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la	Se utiliza a "correctGame", como escenario	Se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes

	entrada para el caso específico.	ideal, sin asociaciones.	para validar la operación
<b>saveGameWithExistingTopic()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctGame", como escenario ideal, con "correctTopic" como asociación.	Se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación
<b>saveGameWithNotExistingTopic()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctGame", como escenario ideal, con "correctTopic" como asociación errónea.	No se ha guardado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación
<b>modifyExistingGame()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctGame", como escenario ideal, con sus valores diferentes de cero.	Se ha modificado con éxito a "Modified" tras realizar las verificaciones correspondientes para validar la operación
<b>modifyGameWithExistingTopic()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctGame", como escenario ideal, con sus valores diferentes de cero y una asociación con "correctTopic"	Se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación
<b>modifyGameWithNonExistingTopic()</b>	Se realiza un llamado al	Se utiliza a "correctGame",	No se ha modificado con

	servicio “modify” teniendo en cuenta la entrada para el caso específico.	como escenario ideal, con sus valores diferentes de cero y una asociación con “correctTopic” erróneo.	éxito el objeto tras realizar las verificaciones correspondientes para validar la operación
<b>modifyNonExistingGame()</b>	Se realiza un llamado al servicio “modify” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctGame”, como escenario ideal, con sus valores diferentes de cero pero con un id no perteneciente a la BD.	No se ha modificado con éxito a “Modified” tras realizar las verificaciones correspondientes para validar la operación

#### StoryCIntegrationTest:

Método	Escenario	Entrada	Salida
<b>saveStoryWithExistingGame()</b>	Se realiza un llamado al servicio “save” teniendo en cuenta la entrada para el caso específico.	Se utiliza a “correctStory”, como escenario ideal, con sus valores diferentes de cero y una asociación con “correctGame”.	Se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación
<b>saveStoryWithNonExistingGame()</b>	Se realiza un llamado al servicio “save” teniendo en cuenta la entrada para el	Se utiliza a “correctStory”, como escenario ideal, con sus valores	No se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes

	caso específico.	diferentes de cero y una asociación con "correctGame" erróneo.	para validar la operación
<b>modifyNonExistingStory()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctStory", como escenario ideal	No se ha modificado con éxito a "Modified doesnt exist" tras realizar las verificaciones correspondientes para validar la operación
<b>modifyExistingStory()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctStory", como escenario ideal, con sus valores diferentes de cero y una asociación con "correctGame".	Se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación
<b>modifyExistingStoryWithExistingGame()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el caso específico.	Se utiliza a "correctStory", como escenario ideal, con sus valores diferentes de cero y una asociación con "correctGame"	Se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes para validar la operación
<b>modifyExistingStoryWithNonExistingGame()</b>	Se realiza un llamado al servicio "modify" teniendo en cuenta la entrada para el	Se utiliza a "correctStory", como escenario ideal, con sus valores diferentes de	No se ha modificado con éxito el objeto tras realizar las verificaciones correspondientes

	caso específico.	cero y una asociación con "correctGame" errónea.	para validar la operación
--	------------------	--	---------------------------

### SaveGameDIntegrationTest:

Método	Escenario	Entrada	Salida
<b>guardarJuego2()</b>	Se realiza un llamado al servicio "save" teniendo en cuenta la entrada para el caso específico.	<ul style="list-style-type: none"> <li>• Se utiliza a "correctGame" como una asociación con "correctTopic"</li> <li>• "correctTopic" posee con "correctStory"</li> <li>• "correctStory" posee con "time1" y "time2"</li> </ul>	Se ha guardado con éxito el juego tras realizar las verificaciones correspondientes para validar la operación, teniendo en cuenta las copias asociadas producto del funcionamiento de los repositorios como fuente de verificación.