

# Diseño de pruebas

Sergio Andrés Lozada Sánchez - A00345617

## Contenido

<b>Pruebas Unitarias</b> .....	3
TopicServiceTest - Create topic.....	3
<b>Prueba 1</b> .....	3
<b>Prueba 2</b> .....	3
<b>Prueba 3</b> .....	3
<b>Prueba 4</b> .....	4
TopicServiceTest - Update topic .....	4
<b>Prueba 5</b> .....	4
<b>Prueba 6</b> .....	4
<b>Prueba 7</b> .....	4
<b>Prueba 8</b> .....	5
<b>Prueba 9</b> .....	5
GameServiceTest - Create game .....	5
<b>Prueba 10</b> .....	5
<b>Prueba 11</b> .....	6
<b>Prueba 12</b> .....	6
<b>Prueba 13</b> .....	6
<b>Prueba 14</b> .....	7
<b>Prueba 15</b> .....	7
<b>Prueba 16 (Punto 5.d del taller)</b> .....	7
GameServiceTest - Update game.....	7
<b>Prueba 17</b> .....	7
<b>Prueba 18</b> .....	8
<b>Prueba 19</b> .....	8
<b>Prueba 20</b> .....	8
<b>Prueba 21</b> .....	9
<b>Prueba 22</b> .....	9
<b>Prueba 23</b> .....	9
StoryServiceTest – Create story .....	9
<b>Prueba 24</b> .....	9
<b>Prueba 25</b> .....	10
<b>Prueba 26</b> .....	10

<b>Prueba 27</b> .....	10
<b>Prueba 28</b> .....	11
StoryServiceTest – Update story .....	11
<b>Prueba 29</b> .....	11
<b>Prueba 30</b> .....	11
<b>Prueba 31</b> .....	12
<b>Prueba 32</b> .....	12
<b>Prueba 33</b> .....	12
<b>Prueba 34</b> .....	13

# Pruebas Unitarias

## TopicServiceTest - Create topic

<b>Prueba 1</b>	<b>Objetivo de la prueba:</b> Probar que al tratar de guardar un tema nulo se lance una excepción indicándolo. <b>Firma del método:</b> <code>public TsscTopic createTopic(TsscTopic topic) throws TopicServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
TopicServiceImp	<code>testCreateTopic0()</code>	Pasar por parámetro un tema nulo	<code>newTopic = null;</code>	Se lanza una excepción indicando que el tema a guardar no puede ser nulo

<b>Prueba 2</b>	<b>Objetivo de la prueba:</b> Probar que un tema es almacenado cuando cumple con las condiciones necesarias para ser guardado. <b>Firma del método:</b> <code>public TsscTopic createTopic(TsscTopic topic) throws TopicServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
TopicServiceImp	<code>testCreateTopic1()</code>	Pasar por parámetro un tema no nulo con número de grupos y sprints mayores a cero	<code>newTopic != null;</code>	El tema es guardado exitosamente.

<b>Prueba 3</b>	<b>Objetivo de la prueba:</b> Probar que se lance una excepción cuando el tema a almacenar tiene la cantidad de grupos iguales o menores a cero. <b>Firma del método:</b> <code>public TsscTopic createTopic(TsscTopic topic) throws TopicServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
TopicServiceImp	<code>testCreateTopic2()</code>	Pasar por parámetro un tema no nulo con número de grupos igual a cero y sprints mayores a cero	<code>newTopic != null;</code>	Se lanza una excepción indicando que el tema a guardar no puede tener grupos iguales o menores a cero

<b>Prueba 4</b>	<b>Objetivo de la prueba:</b> Probar que se lance una excepción cuando el tema a almacenar tiene la cantidad de sprints iguales o menores a cero.  <b>Firma del método:</b> <code>public TsscTopic createTopic(TsscTopic topic) throws TopicServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
TopicServiceImp	testCreateTopic3()	Pasar por parámetro un tema no nulo con número de grupos mayores a cero y sprints igual a cero.	<code>newTopic != null;</code>	Se lanza una excepción indicando que el tema a guardar no puede tener sprints iguales o menores a cero

## TopicServiceTest - Update topic

<b>Prueba 5</b>	<b>Objetivo de la prueba:</b> Probar que al tratar de actualizar un tema nulo se lance una excepción indicándolo.  <b>Firma del método:</b> <code>public TsscTopic updateTopic(TsscTopic topic) throws TopicServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
TopicServiceImp	testUpdateTopic0()	Pasar por parámetro un tema nulo	<code>newTopic = null;</code>	Se lanza una excepción indicando que el tema a actualizar no puede ser nulo

<b>Prueba 6</b>	<b>Objetivo de la prueba:</b> Probar que al tratar de actualizar un tema que no ha sido guardado previamente se lance una excepción.  <b>Firma del método:</b> <code>public TsscTopic updateTopic(TsscTopic topic) throws TopicServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
TopicServiceImp	testUpdateTopic1()	Pasar por parámetro un tema no nulo pero que no ha sido guardado previamente	<code>newTopic != null;</code>	Se lanza una excepción indicando que el tema a actualizar no se encuentra guardado

<b>Prueba 7</b>	<b>Objetivo de la prueba:</b> Probar que un tema es actualizado cuando cumple con las condiciones necesarias para ser actualizado.  <b>Firma del método:</b> <code>public TsscTopic updateTopic(TsscTopic topic) throws TopicServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>

TopicServiceImp	testUpdateTopic2()	Pasar por parámetro un tema no nulo que ha sido guardado previamente y que la cantidad de grupos y sprints son mayores a cero.	newTopic != null;	El tema ha sido actualizado exitosamente.
-----------------	--------------------	--	-------------------	---

<b>Prueba 8</b>	<b>Objetivo de la prueba:</b> Probar que se lance una excepción cuando el tema a actualizar tiene la cantidad de grupos iguales o menores a cero.  <b>Firma del método:</b> public TsscTopic updateTopic(TsscTopic topic) throws TopicServiceException			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
TopicServiceImp	testUpdateTopic3()	Pasar por parámetro un tema no nulo con número de grupos igual a cero y sprints mayores a cero	newTopic != null;	Se lanza una excepción indicando que el tema a actualizar no puede tener grupos iguales o menores a cero

<b>Prueba 9</b>	<b>Objetivo de la prueba:</b> Probar que se lance una excepción cuando el tema a actualizar tiene la cantidad de sprints iguales o menores a cero.  <b>Firma del método:</b> public TsscTopic updateTopic(TsscTopic topic) throws TopicServiceException			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
TopicServiceImp	testUpdateTopic4()	Pasar por parámetro un tema no nulo con número de grupos mayores a cero y sprints igual a cero.	newTopic != null;	Se lanza una excepción indicando que el tema a actualizar no puede tener sprints iguales o menores a cero

## GameServiceTest - Create game

<b>Prueba 10</b>	<b>Objetivo de la prueba:</b> Probar que al tratar de guardar un juego nulo o tema nulo se lance una excepción indicándolo.  <b>Firma del método:</b> public boolean createGame(Object game) throws GameServiceException			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>

GameServiceImp	testCreateGame0()	Pasar por parámetro un juego o tema nulo	<pre>newGame = null; newTopic = null;</pre>	Se lanza una excepción indicando que el juego o tema a guardar no puede ser nulo
----------------	-------------------	--	---	--

<b>Prueba 11</b>	<b>Objetivo de la prueba:</b> Probar que un juego es almacenado cuando cumple con las condiciones necesarias para ser guardado.  <b>Firma del método:</b> <pre>public boolean createGame(Object game) throws GameServiceException</pre>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImp	testCreateGame1()	Pasar por parámetro un juego no nulo con cantidad de grupos y sprints mayores a cero.	<pre>newGame != null;</pre>	El juego es guardado exitosamente.

<b>Prueba 12</b>	<b>Objetivo de la prueba:</b> Probar que se lance una excepción cuando el juego a guardar tiene la cantidad de grupos iguales o menores a cero.  <b>Firma del método:</b> <pre>public boolean createGame(Object game) throws GameServiceException</pre>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImp	testCreateGame2()	Pasar por parámetro un juego no nulo con número de grupos igual a cero y sprints mayores a cero	<pre>newGame != null;</pre>	Se lanza una excepción indicando que el juego a guardar no puede tener grupos iguales o menores a cero

<b>Prueba 13</b>	<b>Objetivo de la prueba:</b> Probar que se lance una excepción cuando el juego a guardar tiene la cantidad de sprints iguales o menores a cero.  <b>Firma del método:</b> <pre>public boolean createGame(Object game) throws GameServiceException</pre>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImp	testCreateGame3()	Pasar por parámetro un juego no nulo con número de grupos mayores a cero y sprints igual a cero.	<pre>newGame != null;</pre>	Se lanza una excepción indicando que el juego a guardar no puede tener sprints iguales o menores a cero

<b>Prueba 14</b>	<b>Objetivo de la prueba:</b> Probar que al tratar de guardar un tema que no ha sido almacenado previamente se lance una excepción indicándolo.  <b>Firma del método:</b> <code>public boolean createGame(Object game) throws GameServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImpl	<code>testCreateGame4()</code>	Pasar por parámetro un tema no nulo y que previamente no ha sido almacenado	<code>newTopic = !null;</code>	Se lanza una excepción indicando que el tema a guardar no se encuentra almacenado

<b>Prueba 15</b>	<b>Objetivo de la prueba:</b> Probar que un tema ha sido almacenado por que cumple con las condiciones para ser guardado.  <b>Firma del método:</b> <code>public boolean createGame(Object game) throws GameServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImpl	<code>testCreateGame5()</code>	Pasar por parámetro un tema no nulo y que previamente ha sido almacenado	<code>newTopic != null;</code>	El tema es guardado exitosamente.

<b>Prueba 16 (Punto 5.d del taller)</b>	<b>Objetivo de la prueba:</b> Probar que un tema ha sido almacenado por que cumple con las condiciones para ser guardado y que se crea un juego con las historias y cronogramas del tema.  <b>Firma del método:</b> <code>public boolean createGame(Object game) throws GameServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImpl	<code>testCreateGame6()</code>	Pasar por parámetro un tema no nulo y que previamente ha sido almacenado.	<code>newTopic != null;</code>	El tema es guardado exitosamente y se ha creado el juego con las historias y cronogramas del juego.

## GameServiceTest - Update game

<b>Prueba 17</b>	<b>Objetivo de la prueba:</b> Probar que al tratar de actualizar un juego nulo o tema nulo se lance una excepción indicándolo.  <b>Firma del método:</b> <code>public boolean updateGame(Object game) throws GameServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>

GameServiceImp	testUpdateGame0()	Pasar por parámetro un juego o tema nulo	<pre>newGame = null; newTopic = null;</pre>	Se lanza una excepción indicando que el juego o tema a actualizar no puede ser nulo
----------------	-------------------	--	---	---

<b>Prueba 18</b>	<b>Objetivo de la prueba:</b> Probar que se lance una excepción cuando el juego a actualizar tiene la cantidad de grupos iguales o menores a cero.  <b>Firma del método:</b> <pre>public boolean updateGame(Object game) throws GameServiceException</pre>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImp	testUpdateGame1()	Pasar por parámetro un juego no nulo con número de grupos igual a cero y sprints mayores a cero	<pre>newGame != null;</pre>	Se lanza una excepción indicando que el juego a actualizar no puede tener grupos iguales o menores a cero

<b>Prueba 19</b>	<b>Objetivo de la prueba:</b> Probar que se lance una excepción cuando el juego a actualizar tiene la cantidad de sprints iguales o menores a cero.  <b>Firma del método:</b> <pre>public boolean updateGame(Object game) throws GameServiceException</pre>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImp	testUpdateGame2()	Pasar por parámetro un juego no nulo con número de grupos mayores a cero y sprints igual a cero.	<pre>newGame != null;</pre>	Se lanza una excepción indicando que el juego a actualizar no puede tener sprints iguales o menores a cero

<b>Prueba 20</b>	<b>Objetivo de la prueba:</b> Probar que se lance una excepción cuando el juego a actualizar no se ha guardado previamente.  <b>Firma del método:</b> <pre>public boolean updateGame(Object game) throws GameServiceException</pre>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImp	testUpdateGame3()	Pasar por parámetro un juego no nulo que no se ha guardado previamente.	<pre>newGame != null;</pre>	Se lanza una excepción indicando que el juego a actualizar no ha sido creado previamente



<b>Prueba 21</b>	<b>Objetivo de la prueba:</b> Probar que un juego es actualizado cuando cumple con las condiciones necesarias para actualizar.  <b>Firma del método:</b> <code>public boolean updateGame(Object game) throws GameServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImpl	testUpdateGame4()	Pasar por parámetro un juego no nulo que se ha guardado previamente y con cantidad de grupos y sprints mayores a cero.	<code>newGame != null;</code>	Se actualiza el juego correctamente.

<b>Prueba 22</b>	<b>Objetivo de la prueba:</b> Probar que al tratar de actualizar un tema que no ha sido almacenado previamente se lance una excepción indicándolo.  <b>Firma del método:</b> <code>public boolean updateGame(Object game) throws GameServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImpl	testUpdateGame5()	Pasar por parámetro un tema no nulo y que previamente no ha sido almacenado	<code>newTopic = !null;</code>	Se lanza una excepción indicando que el tema a actualizar no se encuentra almacenado

<b>Prueba 23</b>	<b>Objetivo de la prueba:</b> Probar que un tema ha sido almacenado por que cumple con las condiciones para ser guardado.  <b>Firma del método:</b> <code>public boolean updateGame(Object game) throws GameServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
GameServiceImpl	testUpdateGame6()	Pasar por parámetro un tema no nulo y que previamente ha sido almacenado	<code>newTopic != null;</code>	El tema es actualizado exitosamente.

StoryServiceTest – Create story

<b>Prueba 24</b>	<b>Objetivo de la prueba:</b> Probar que al tratar de guardar una historia nula se lance una excepción indicándolo.  <b>Firma del método:</b> <code>public TsscStory createStoryService(TsscStory story) throws StoryServiceException</code>			
<b>Clase por probar</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>

StoryServiceImp	testCreateStory0()	Pasar por parámetro una historia nula	newStory = null;	Se lanza una excepción indicando que la historia a guardar no puede ser nula
-----------------	--------------------	---------------------------------------	------------------	--

<b>Prueba 25</b> <b>Objetivo de la prueba:</b> Probar que al tratar de guardar una historia no nula pero sin un juego relacionado se lance una excepción indicándolo. <b>Firma del método:</b> public TsscStory createStoryService(TsscStory story) throws StoryServiceException				
Clase por probar	Método prueba	Escenario	Valor de entrada	Resultado
StoryServiceImp	testCreateStory1()	Pasar por parámetro una historia no nula sin ningún juego relacionado	newStory != null;	Se lanza una excepción indicando que la historia a guardar debe tener un juego relacionado

<b>Prueba 26</b> <b>Objetivo de la prueba:</b> Probar que al tratar de guardar una historia no nula con un juego no almacenado previamente se lance una excepción indicándolo. <b>Firma del método:</b> public TsscStory createStoryService(TsscStory story) throws StoryServiceException				
Clase por probar	Método prueba	Escenario	Valor de entrada	Resultado
StoryServiceImp	testCreateStory2()	Pasar por parámetro una historia no nula con un juego relacionado que no ha sido guardado previamente	newStory != null;	Se lanza una excepción indicando que la historia a guardar tiene un juego relacionado que debe estar almacenado previamente.

<b>Prueba 27</b> <b>Objetivo de la prueba:</b> Probar que al tratar de guardar una historia no nula que no cumple con lo necesario para ser almacenada, se lance una excepción. <b>Firma del método:</b> public TsscStory createStoryService(TsscStory story) throws StoryServiceException				
Clase por probar	Método prueba	Escenario	Valor de entrada	Resultado
StoryServiceImp	testCreateStory3()	Pasar por parámetro una historia no nula con un juego relacionado que ha sido guardado previamente, pero	newStory != null;	Se lanza una excepción indicando que la historia a guardar tiene un juego relacionado que debe estar

		con valor del negocio, sprint inicial y prioridad igual a cero.		almacenado previamente.
--	--	---	--	-------------------------

<b>Prueba 28</b>		<b>Objetivo de la prueba:</b> Probar que se guardar una historia no nula que cumple con lo necesario para ser almacenada.  <b>Firma del método:</b> <code>public TsscStory createStoryService(TsscStory story) throws StoryServiceException</code>			
<b>Clase probar</b>	<b>por</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
StoryServiceImp		<code>testCreateStory4()</code>	Pasar por parámetro una historia no nula con un juego relacionado que ha sido guardado previamente, pero con valor del negocio, sprint inicial y prioridad mayor a cero.	<code>newStory != null;</code>	Se guarda exitosamente la historia.

#### StoryServiceTest – Update story

Prueba 29		<b>Objetivo de la prueba:</b> Probar que al tratar de actualizar una historia nula se lance una excepción indicándolo.  <b>Firma del método:</b> <code>public TsscStory updateStoryService(TsscStory story) throws StoryServiceException</code>			
Clase probar	por	Método prueba	Escenario	Valor de entrada	Resultado
StoryServiceImp		<code>testUpdateStory0()</code>	Pasar por parámetro una historia nula	<code>newStory = null;</code>	Se lanza una excepción indicando que la historia a actualizar no puede ser nula
Prueba 30		<b>Objetivo de la prueba:</b> Probar que al tratar de actualizar una historia no nula y no almacenada previamente se lance una excepción.  <b>Firma del método:</b> <code>public TsscStory updateStoryService(TsscStory story) throws StoryServiceException</code>			
Clase probar	por	Método prueba	Escenario	Valor de entrada	Resultado

StoryServiceImp	testUpdateStory1()	Pasar por parámetro una historia no nula	newStory != null;	Se lanza una excepción indicando que la historia a actualizar no se ha guardado previamente
-----------------	--------------------	--	-------------------	---

Prueba 31		<b>Objetivo de la prueba:</b> Probar que al tratar de actualizar una historia no nula pero sin un juego relacionado se lance una excepción indicándolo.  <b>Firma del método:</b> <code>public TsscStory updateStoryService(TsscStory story) throws StoryServiceException</code>			
Clase probar	por	Método prueba	Escenario	Valor de entrada	Resultado
StoryServiceImp		<code>testUpdateStory2()</code>	Pasar por parámetro una historia no nula sin ningún juego relacionado	<code>newStory != null;</code>	Se lanza una excepción indicando que la historia a actualizar debe tener un juego relacionado

Prueba 32		<b>Objetivo de la prueba:</b> Probar que al tratar de actualizar una historia no nula con un juego no almacenado previamente se lance una excepción indicándolo.			
		<b>Firma del método:</b> <code>public TsscStory updateStoryService(TsscStory story) throws StoryServiceException</code>			
Clase probar	por	Método prueba	Escenario	Valor de entrada	Resultado
StoryServiceImp		<code>testUpdateStory3()</code>	Pasar por parámetro una historia no nula con un juego relacionado que no ha sido guardado previamente	<code>newStory != null;</code>	Se lanza una excepción indicando que la historia a actualizar tiene un juego relacionado que debe estar almacenado previamente.

<b>Prueba 33</b>		<b>Objetivo de la prueba:</b> Probar que al tratar de actualizar una historia no nula que no cumple con lo necesario para ser almacenada, se lance una excepción.			
		<b>Firma del método:</b> <code>public TsscStory updateStoryService(TsscStory story) throws StoryServiceException</code>			
<b>Clase probar</b>	<b>por</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
StoryServiceImp		<code>testUpdateStory4()</code>	Pasar por parámetro una historia no nula con un juego relacionado que ha sido guardado previamente, pero	<code>newStory != null;</code>	Se lanza una excepción indicando que la historia a actualizar tiene un juego relacionado que debe estar

		con valor del negocio, sprint inicial y prioridad igual a cero.		almacenado previamente.
--	--	---	--	-------------------------

<b>Prueba 34</b>		<b>Objetivo de la prueba:</b> Probar que se actualiza una historia no nula que cumple con lo necesario para ser almacenada.  <b>Firma del método:</b> <code>public TsscStory updateStoryService(TsscStory story) throws StoryServiceException</code>			
<b>Clase probar</b>	<b>por</b>	<b>Método prueba</b>	<b>Escenario</b>	<b>Valor de entrada</b>	<b>Resultado</b>
StoryServiceImp		<code>testUpdateStory5()</code>	Pasar por parámetro una historia no nula con un juego relacionado que ha sido guardado previamente, pero con valor del negocio, sprint inicial y prioridad mayor a cero.	<code>newStory != null;</code>	Se ha actualizado exitosamente la historia.