

# Acceptance tests

## <Acme-Essay> <2.0>

---

### *Indexing data*

Designer	<i>Moises Llamas Garcia Grupo 12</i>
Tester	<i>Moises Llamas Garcia Grupo 12</i>
Notes	<i>Estas pruebas son realizadas al Grupo 32</i>

Grupo 12  
Moisés Llamas García  
Miguel Ángel Bellido Millán  
Valentín Barquero Sauci  
Ángel Núñez Cañete

## Functional requirement <01>

### Description

*Registrarse en el Sistema como un author.*

### Access

*Accediendo en la página principal, sin estar logueado en el sistema y hacienda click en la opción REGISTER y seleccionando as Author.*

### Checks

#### Check <#01>

Description	<i>No rellenar ningún campo.</i>
Expected	<i>Nos debe de indicar los errores correspondientes a cada campo del formulario.</i>
Outcome	<i>No podemos registrarnos en el Sistema y nos indica los campos necesarios que deben de rellenarse.</i>
Notes	<i>Es correcto</i>

#### Check <#02>

Description	<i>Comprobar que el botón cancelar funciona correctamente.</i>
Expected	<i>El formulario nos debe de redirigir a la página de inicio.</i>
Outcome	<i>El botón cancelar no redirecciona a otra página.</i>
Notes	<i>Necesita ser arreglado.</i>

#### Check <#03>

Description	<i>Rellenar el campo de forma errónea.</i>
Expected	<i>Al guardar nos debe de indicar el error del campo mal introducido.</i>
Outcome	<i>Informa del error sobre la fecha de nacimiento, y al introducir las contraseñas mal se detecta que no coinciden. Se prueba también la fecha en pasado y no deja dar de alta. La tarjeta de crédito también es validada correctamente. Se comprueba también que no permita hacer un registro si no se aceptan los términos de condiciones.</i>
Notes	<i>No existe un vínculo directo a la información de los términos de uso en el formulario. Todos los errores son informados mediante commit error. Podrían añadirse una aclaración sobre el tipo error.</i>

#### Check <#04>

Description	<i>Rellenar el campo de forma correcta.</i>
Expected	<i>Comprobar que el autor se ha dado de alta en el Sistema de forma correcta.</i>
Outcome	<i>Introduciendo los datos correctamente se comprueba que es posible almacenar el usuario.</i>
Notes	

### *Check <#05>*

Description	<i>Notificación de errores al producirse introducción de datos con fines maliciosos (Cross Scripting e inyección SQL)</i>
Expected	<i>Se deberá notificar al usuario en el formulario cuando este intente introducir fragmentos de SQL, Javascript,Html inseguro en aquellos campos que precisen de cadenas de caracteres.</i>
Outcome	<i>Se permite introducir HTML inseguro, en concreto valores del tipo &lt;h1&gt;nombre&lt;/h1&gt;, por lo que se puede introducir datos maliciosos.</i>
Notes	<i>No es válido.</i>

## Functional requirement <02>

### Description

*Registrarse en el Sistema como un organizador.*

### Access

*Accediendo en la pagina principal, sin estar logueado en el sistema y hacienda click en la opción REGISTER y seleccionando as Organiser.*

### Checks

#### Check <#01>

Description	<i>No rellenar ningún campo.</i>
Expected	<i>Nos debe de indicar los errores correspondientes a cada campo del formulario.</i>
Outcome	<i>No podemos registrarnos en el Sistema y nos indica los campos necesarios que deben de rellenarse.</i>
Notes	<i>Es correcto</i>

#### Check <#02>

Description	<i>Comprobar que el botón cancelar funciona correctamente.</i>
Expected	<i>El formulario nos debe de redirigir a la página de inicio.</i>
Outcome	<i>El botón cancelar no redirecciona a otra página.</i>
Notes	<i>Necesita ser arreglado.</i>

#### Check <#03>

Description	<i>Rellenar el campo de forma errónea.</i>
Expected	<i>Al guardar nos debe de indicar el error del campo mal introducido.</i>
Outcome	<i>Informa del error sobre la fecha de nacimiento, y al introducir las contraseñas mal se detecta que no coinciden. Se prueba también la fecha en pasado y no deja dar de alta. La tarjeta de crédito también es validada correctamente. Se comprueba también que no permita hacer un registro si no se aceptan los términos de condiciones.</i>
Notes	<i>No existe un vínculo directo a la información de los términos de uso en el formulario. Todos los errores son informados mediante commit error. Podrían añadirse una aclaración sobre el tipo error.</i>

#### Check <#04>

Description	<i>Rellenar el campo de forma correcta.</i>
Expected	<i>Comprobar que el autor se ha dado de alta en el Sistema de forma correcta.</i>
Outcome	<i>Introduciendo los datos correctamente se comprueba que es posible almacenar el usuario.</i>
Notes	

### *Check <#05>*

Description	<i>Notificación de errores al producirse introducción de datos con fines maliciosos (Cross Scripting e inyección SQL)</i>
Expected	<i>Se deberá notificar al usuario en el formulario cuando este intente introducir fragmentos de SQL, Javascript,Html inseguro en aquellos campos que precisen de cadenas de caracteres.</i>
Outcome	<i>Se permite introducir HTML inseguro, en concreto valores del tipo &lt;h1&gt;nombre&lt;/h1&gt;, por lo que se puede introducir datos maliciosos.</i>
Notes	<i>No es válido.</i>

## Functional requirement <03>

### Description

Listar los concursos que el Sistema tiene almacenado sin estar registrados.

### Access

Accediendo en la página principal, sin estar logueado en el sistema y desplegando en la opción CONTEST y seleccionando View All.

### Checks

#### Check <#01>

Description	Comprobar que se muestra la información esperada.
Expected	La tabla deberá de mostrar la siguiente información de cada concurso: nombre, descripción, resultado y las fechas de comienzo y fin.
Outcome	Se muestran los datos esperados
Notes	

#### Check <#02>

Description	Comprobar que la paginación de la tabla funciona correctamente.
Expected	Cuando la tabla contenga más de cinco registros, se deberán poder ver páginas posteriores sin que se produzcan errores.
Outcome	La paginación funciona correctamente.
Notes	

#### Check <#03>

Description	Comprobar el comportamiento de las entradas 'solitarias' en la última página de la tabla.
Expected	La estructura de la tabla deberá conservarse si una entrada presente en la última página de la tabla es editada y/o eliminada.
Outcome	Conserva la estructural eliminar la única entrada de la ultima página.
Notes	

## Functional requirement <04>

### Description

Enviar un ensayo para un determinado concurso estando registrado como autor.

### Access

Logueandonos como autor, desplegando la opción contest, seleccionado viewAll, y en la columna actions de la tabla clickando en el enlace de New Essay.

### Checks

#### Check <#01>

Description	No rellenar ningún campo.
Expected	Nos debe de indicar los errores correspondientes a cada campo del formulario.
Outcome	Nos indica los errores
Notes	

#### Check <#02>

Description	Comprobar que el botón cancelar funciona correctamente.
Expected	El formulario nos debe de redirigir a la página de inicio.
Outcome	Funciona correctamente, lleva hacia la lista de essay del usuario logueada.
Notes	

#### Check <#03>

Description	Rellenar el campo de forma errónea.
Expected	Al guardar nos debe de indicar el error del campo mal introducido.
Outcome	Funciona correctamente, al introducir un campo mal y nos avisa de los errores.
Notes	

#### Check <#04>

Description	Rellenar el campo de forma correcta.
Expected	Comprobar que el ensayo se ha creado correctamente y está asociado al concurso elegido
Outcome	El essay creado aparece en la lista de essays mandados.
Notes	

#### Check <#05>

Description	Notificación de errores al producirse introducción de datos con fines maliciosos (Cross Scripting e inyección SQL)
Expected	Se deberá notificar al usuario en el formulario cuando este intente introducir fragmentos de SQL o Javascript en aquellos campos que precisen de cadenas de caracteres.
Outcome	Se introduce en el título del essay <code>&lt;script&gt;&gt;window.alert("Hackeado")&lt;/script&gt;</code> y deja almacenarlo, permite la entrada de dos maliciosos.
Notes	Al listar los essays aparece una ventana emergente producto del intento anterior.

## Functional requirement <05>

### Description

*Editar un ensayo previamente creado, registrado como autor.*

### Access

*Logueandonos como autor, desplegando la opción Essay, seleccionado My Essays , y en la columna actions de la tabla clickando en el enlace de Details.*

### Checks

#### Check <#01>

Description	<i>No rellenar ningún campo.</i>
Expected	<i>Nos debe de indicar los errores correspondientes a cada campo del formulario.</i>
Outcome	<i>Informa de que los campos no pueden ser vacíos.</i>
Notes	

#### Check <#02>

Description	<i>Comprobar que el botón cancelar funciona correctamente.</i>
Expected	<i>El formulario nos debe de redirigir a la página de la lista de los essays del usuario logueado .</i>
Outcome	<i>Funciona correctamente</i>
Notes	

#### Check <#03>

Description	<i>Rellenar el campo de forma errónea.</i>
Expected	<i>Al guardar nos debe de indicar el error del campo mal introducido.</i>
Outcome	<i>Informa sobre los errores de que no pueden ir en blanco los campos al enviar el formulario de edición.</i>
Notes	

#### Check <#04>

Description	<i>Rellenar el campo de forma correcta.</i>
Expected	<i>Comprobar que el ensayo se ha creado correctamente y está asociado al concurso elegido.</i>
Outcome	<i>Correcto</i>
Notes	



### *Check <#05>*

Description	<i>Notificación de errores al producirse introducción de datos con fines maliciosos (Cross Scripting e inyección SQL)</i>
Expected	<i>Se deberá notificar al usuario en el formulario cuando este intente introducir fragmentos de SQL o Javascript en aquellos campos que precisen de cadenas de caracteres.</i>
Outcome	<i>Se puede introducir datos maliciosos.</i>
Notes	

## Functional requirement <06>

### Description

Listar los ensayos creados por un determinado autor.

### Access

Indicar la navegacion seguida.

### Checks

#### Check <#01>

Description	Comprobar que se muestra la información esperada.
Expected	La tabla deberá mostrar la siguiente información de cada concurso: título, contenido y fecha de envío.
Outcome	Se muestra la información necesaria.
Notes	

#### Check <#02>

Description	Comprobar que la paginación de la tabla funciona correctamente.
Expected	Cuando la tabla contenga más de cinco registros, se deberán poder ver páginas posteriores sin que se produzcan errores.
Outcome	La paginación también funciona correctamente.
Notes	

## Functional requirement <07>

### Description

Crear, editar y borrar un concurso, estando logueado como organizador

### Access

Logueandonos como organiser, desplegando la opción Contest, seleccionado My Contest , y en la vista de la lista de sus contest clickando en el botón Create.

### Checks

#### Check <#01>

Description	No rellenar ningún campo.
Expected	Nos debe de indicar los errores correspondientes a cada campo del formulario.
Outcome	Lanza un "NullPointerException" al guardar el formulario con los campos vacíos mediante un Panic.
Notes	Se debería de especificar el error que se está cometiendo, para que el usuario sepa que error está introduciendo en el formulario.

#### Check <#02>

Description	Comprobar que el botón cancelar funciona correctamente.
Expected	El formulario nos debe de redirigir a la página de inicio.
Outcome	No hay botón de cancelar en el formulario para el contest
Notes	Esto debe de ser arreglado, para que el usuario pueda retroceder hacia la página de inicio.

#### Check <#03>

Description	Comprobar que el botón eliminar funciona correctamente.
Expected	Debe de borrarse dicho concurso.
Outcome	Los contests con essays no tienen botón de borrado algo que es correcto. Los contests sin essays publicadas son borrados sin problemas y se muestra un botón en el formulario para ello.
Notes	El test es correcto

#### Check <#04>

Description	Rellenar el campo de forma errónea.
Expected	Al guardar nos debe de indicar el error del campo mal introducido.
Outcome	Se prueba a introducir el deadline posterior al holding date del contest y se informa de un error commit error. El comportamiento es correcto aunque se podría a portar más información sobre el error. Fechas en el pasado son admitidas a la hora de almacenar o guardar un contest por lo que este comportamiento no es válido.
Notes	Debe de informarse de los errores específicos que se cometen en cada momento, no solo debe de mostrarse "commit error"

#### *Check <#05>*

Description	<i>Rellenar el campo de forma correcta.</i>
Expected	<i>Comprobar que el concurso se ha creado correctamente.</i>
Outcome	<i>Funciona, permite almacenar un concurso.</i>
Notes	

#### *Check <#06>*

Description	<i>Notificación de errores al producirse introducción de datos con fines maliciosos (Cross Scripting e inyección SQL)</i>
Expected	<i>Se deberá notificar al usuario en el formulario cuando este intente introducir fragmentos de SQL o Javascript en aquellos campos que precisen de cadenas de caracteres.</i>
Outcome	<i>Permite la introducción de datos maliciosos.</i>
Notes	

## Functional requirement <08>

### Description

Registrar el estado para un concurso, estando logueado como organizador

### Access

Logueandonos como organiser, desplegando la opción Contest, seleccionado My Contest , y en la vista de la lista de sus contes, clickando en la columna actions Edit Contest.Se habilita la opción de añadir un resultado cuando el contest ha finalizado.

### Checks

#### Check <#01>

Description	Comprobar que el botón cancelar funciona correctamente.
Expected	El formulario nos debe de redirigir a la página de inicio.
Outcome	El botón cancelar nos redirige a la página anterior.
Notes	Este test es correcto.

#### Check <#02>

Description	Rellenar el campo de forma correcta.
Expected	Comprobar que el concurso se ha actualizado correctamente.
Outcome	El resultado del concurso se almacena correctamente.
Notes	Esta opción se habilita en el formulario de edición una vez el contest ha finalizado.

#### Check <#03>

Description	Notificación de errores al producirse introducción de datos con fines maliciosos (Cross Scripting e inyección SQL)
Expected	Se deberá notificar al usuario en el formulario cuando este intente introducir fragmentos de SQL o Javascript en aquellos campos que precisen de cadenas de caracteres.
Outcome	Se puede introducir datos del tipo <h1>algo</h1> en el resultado.
Notes	

## Functional requirement <09>

### Description

Listar los concursos creados por un determinado organizador.

### Access

Logueandonos como organiser, desplegando la opción Contest, seleccionado My Contests.

### Checks

#### Check <#01>

Description	Comprobar que se muestra la información esperada.
Expected	La tabla deberá mostrar la siguiente información de cada concurso: título, contenido, estado y fecha de envío.
Outcome	Muestra la información correctamente
Notes	

#### Check <#02>

Description	Comprobar que la paginación de la tabla funciona correctamente.
Expected	Cuando la tabla contenga más de cinco registros, se deberán poder ver páginas posteriores sin que se produzcan errores.
Outcome	La paginación funciona correctamente.
Notes	

#### Check <#03>

Description	Comprobar el comportamiento de las entradas 'solitarias' en la última página de la tabla.
Expected	La estructura de la tabla deberá conservarse si una entrada presente en la última página de la tabla es editada y/o eliminada.
Outcome	Devuelve a la página 1 de la lista. El comportamiento es el esperado.
Notes	

## Functional requirement <10>

### Description

Registrar a un administrador, estando logueado como administrador.

### Access

Logueandonos como administrador, desplegando la opción ADMINISTRATOR, seleccionado New Administrator.

### Checks

#### Check <#01>

Description	No rellenar ningún campo.
Expected	Nos debe de indicar los errores correspondientes a cada campo del formulario.
Outcome	No podemos registrarnos en el Sistema y nos indica los campos necesarios que deben de rellenarse junto con las restricciones para poder registrar un administrador.
Notes	

#### Check <#02>

Description	Comprobar que el botón cancelar funciona correctamente.
Expected	El formulario nos debe de redirigir a la página de inicio.
Outcome	El botón cancelar lleva a una página a la que no tiene acceso.
Notes	Debe de redirigirse a una página a la que se tenga acceso.

#### Check <#03>

Description	Rellenar el campo de forma errónea.
Expected	Al guardar nos debe de indicar el error del campo mal introducido.
Outcome	Al introducir los datos mal se nos avisa del tipo de error. Al introducir contraseñas distintas o si no aceptamos los términos salta un mensaje de "Could not commit this operation". Comportamiento esperado.
Notes	Debería de especificarse el error que se está cometiendo en cada momento para facilitar al usuario e indicar que errores está cometiendo en el formulario.

#### Check <#04>

Description	Rellenar el campo de forma correcta.
Expected	Comprobar que el administrador se ha dado de alta en el Sistema de forma correcta.
Outcome	El administrador se almacena correctamente introduciendo los campos bien.
Notes	

### *Check <#05>*

Description	<i>Notificación de errores al producirse introducción de datos con fines maliciosos (Cross Scripting e inyección SQL)</i>
Expected	<i>Se deberá notificar al usuario en el formulario cuando este intente introducir fragmentos de SQL o Javascript en aquellos campos que precisen de cadenas de caracteres.</i>
Outcome	<i>Se puede introducir datos maliciosos.</i>
Notes	



## Functional requirement <11>

### Description

Listar los autores y organizadores registrados en el Sistema, estando logueado como administrador .

### Access

Logueandonos como administrator, desplegando la opción CUSTOMERS, seleccionado View All.

### Checks

#### Check <#01>

Description	Comprobar que se muestra la información esperada.
Expected	Las tablas deberán de mostrar la información pertinente respecto a los autores y organizadores: nombre, apellidos, email, teléfono de contacto, fecha de nacimiento y nacionalidad.
Outcome	Se muestra la información correcta con las columnas correspondientes.
Notes	

#### Check <#02>

Description	Comprobar que la paginación de la tabla funciona correctamente.
Expected	Cuando la tabla contenga más de cinco registros, se deberán poder ver páginas posteriores sin que se produzcan errores.
Outcome	La paginación funciona correctamente.
Notes	

#### Check <#03>

Description	Comprobar el comportamiento de las entradas 'solitarias' en la última página de la tabla.
Expected	La estructura de la tabla deberá conservarse si una entrada presente en la última página de la tabla es editada y/o eliminada.
Outcome	Devuelve a la página 1 de la lista. El comportamiento es el esperado.
Notes	

## Functional requirement <12>

### Description

Ver la lista de ensayos que hay publicados para un determinado concurso y ver sus detalles.

### Access

Accediendo en la página principal, sin estar logueado en el sistema y desplegando en la opción *CONTEST*, seleccionando *View All* y haciendo click en *Pubilshed*.

### Checks

#### Check <#01>

Description	Comprobar que se muestra la información esperada.
Expected	La tabla deberá mostrar la siguiente información de cada concurso: título, texto, fecha de envío y si están publicados.
Outcome	Se muestran los datos esperados
Notes	

#### Check <#02>

Description	Comprobar que la paginación de la tabla funciona correctamente.
Expected	Cuando la tabla contenga más de cinco registros, se deberán poder ver páginas posteriores sin que se produzcan errores.
Outcome	La paginación funciona correctamente.
Notes	

#### Check <#03>

Description	Comprobar el comportamiento de las entradas 'solitarias' en la última página de la tabla.
Expected	La estructura de la tabla deberá conservarse si una entrada presente en la última página de la tabla es editada y/o eliminada.
Outcome	Conserva la estructura al eliminar la única entrada de la última página.
Notes	

## Functional requirement <13>

### Description

*Decidir si un ensayo se publica o no, estando logueado como organizador.*

### Access

*Accediendo en la página principal, estando logueados como organiser y desplegando en la opción CONTEST, seleccionando View All, haciendo click en Submitted y luego haciendo click en Publish para publicar dicho ensayo.*

### Checks

#### Check <#01>

Description	<i>Comprobar que el enlace funciona correctamente.</i>
Expected	<i>El enlace para publicar debe de funcionar correctamente.</i>
Outcome	<i>El enlace para publicar funciona correctamente.</i>
Notes	

#### Check <#02>

Description	<i>Comprobar que el botón cancelar funciona correctamente.</i>
Expected	<i>El formulario no debe de redirigir a la vista anterior.</i>
Outcome	<i>El botón de cancelar funciona correctamente</i>
Notes	

## Functional requirement <14>

### Description

Listar los ensayos que un autor ha enviado a un determinado concurso, incluyendo la información de la sesión pública, logueado como un autor.

### Access

Haciendo login como autor, en el menú debeos de seleccionar CONTEST -> View All -> My Essay -> Public Session.

### Checks

#### Check <#01>

Description	Comprobar que se muestra la información esperada.
Expected	La tabla deberá mostrar la siguiente información de cada ensayo: título, contenido, fecha de envío y la información pertinente acerca de la sesión pública.
Outcome	La lista devuelve los datos correctos y los enlaces funcionan correctamente.
Notes	

#### Check <#02>

Description	Comprobar que la paginación de la tabla funciona correctamente.
Expected	Cuando la tabla contenga más de cinco registros, se deberán poder ver páginas posteriores sin que se produzcan errores.
Outcome	La paginación funciona correctamente.
Notes	

#### Check <#03>

Description	Comprobar el comportamiento de las entradas 'solitarias' en la última página de la tabla.
Expected	La estructura de la tabla deberá conservarse si una entrada presente en la última página de la tabla es editada y/o eliminada.
Outcome	Conserva la estructura al eliminar la única entrada de la última página.
Notes	

## Functional requirement <15>

### Description

Añadir un organizador a un determinado concurso, estando logueado como organizador.

### Access

Logueandonos como organizador, debemos de dirigirnos al menú y seguir la siguiente navegación: CONTEST->My Contest y en la columna actions, darle al link Add Organiser.

### Checks

#### Check <#01>

Description	No rellenar ningún campo.
Expected	Nos debe de indicar los errores correspondientes a cada campo del formulario.
Outcome	No podemos añadir un organizador que sea nulo.
Notes	

#### Check <#02>

Description	Comprobar que el botón cancelar funciona correctamente.
Expected	El formulario nos debe de redirigir a la lista de nuestros concursos.
Outcome	El botón cancelar nos redirige a la vista anterior, por lo que funciona correctamente.
Notes	

#### Check <#03>

Description	Rellenar el campo de forma correcta.
Expected	Comprobar que se ha añadido de forma correcta el organizador al concurso.
Outcome	Podemos observar en la lista de organizadores que se añaden correctamente.
Notes	

## Functional requirement <16>

### Description

Crear y editar una sesión pública para un concurso que el organiza, estando logueado como organizador.

### Access

Para crear una sesión pública, debemos de loguearnos como organizador, debemos de dirigirnos al menú y seguir la siguiente navegación: CONTEST->My Contest y en la columna actions, darle al link Create Sessions.

Para editar una sesión pública, debemos de loguearnos como organizador, debemos de dirigirnos al menú y seguir la siguiente navegacion: ABOUT SESSIONS->My Sessions y en la columna actions, darle al link edit.

### Checks

#### Check <#01>

Description	No rellenar ningún campo.
Expected	Nos debe de indicar los errores correspondientes a cada campo del formulario.
Outcome	Nos indica que no se puede realizar la operación, pero no especifica que errores se producen.
Notes	

#### Check <#02>

Description	Comprobar que el botón cancelar funciona correctamente.
Expected	El formulario nos debe de redirigir a la lista de nuestros concursos.
Outcome	El botón cancelar nos redirige a la vista esperada.
Notes	

#### Check <#03>

Description	Rellenar el campo de forma errónea.
Expected	Al guardar nos debe de indicar el error del campo mal introducido.
Outcome	Al intentar poner en capacidad texto plano, nos muestra un error del convertidor. Al introducir fechas erróneas, tanto la de comienzo como la de fin, nos muestra una página de error.
Notes	Debería de corregirse estos errores y mostrar un error que a nivel de usuario se pueda entender.

#### *Check <#04>*

Description	<i>Rellenar el campo de forma correcta.</i>
Expected	<i>Comprobar que la sesión publica se ha creado correctamente.</i>
Outcome	<i>Editamos un campo de la sesión pública y comprobamos que se edita de forma correcta.</i>
Notes	

## Functional requirement <17>

### Description

Listar los concursos que él ha creado incluyendo las sesiones públicas, estando logueado como organizador.

### Access

Debemos de loguearnos como organizador, debemos de dirigirnos al menú y seguir la siguiente navegacion: CONTEST->My Contest y en la columna actions, darle al link Sessions.

### Checks

#### Check <#01>

Description	Comprobar que se muestra la información esperada.
Expected	La tabla deberá mostrar la siguiente información de cada concurso: nombre, descripción, resultado, fechas de inicio y fin y la información pertinente acerca de la sesión pública.
Outcome	La lista muestra la información correcta y además, el enlace que nos dirige a la sesión pública funciona correctamente.
Notes	

#### Check <#02>

Description	Comprobar que la paginación de la tabla funciona correctamente.
Expected	Cuando la tabla contenga más de cinco registros, se deberán poder ver páginas posteriores sin que se produzcan errores.
Outcome	La paginación funciona correctamente.
Notes	

#### Check <#03>

Description	Comprobar el comportamiento de las entradas 'solitarias' en la última página de la tabla.
Expected	La estructura de la tabla deberá conservarse si una entrada presente en la última página de la tabla es editada y/o eliminada.
Outcome	Conserva la estructura al eliminar la única entrada de la última página.
Notes	



## Functional requirement <18>

### Description

Añadir un presidente la sesión pública, estando logueado como organizador.

### Access

Debemos de loguearnos como organizador, debemos de dirigirnos al menú y seguir la siguiente navegacion: ABOUT SESSIONS->My Sessions y en la columna actions, darle al link edit.

.

### Checks

#### Check <#01>

Description	No rellenar ningún campo.
Expected	Nos debe de indicar los errores correspondientes a cada campo del formulario.
Outcome	No podemos mandar un organizador nulo, nos devuelve una página de error, "org.hibernate.LazyInitializationException".
Notes	Debería de indicar un error que sea entendible a nivel de usuario.

#### Check <#02>

Description	Comprobar que el botón cancelar funciona correctamente.
Expected	El formulario nos debe de redirigir a la página de inicio.
Outcome	Nos redirige a la lista de nuestros concursos.
Notes	

#### Check <#03>

Description	Rellenar el campo de forma correcta.
Expected	Comprobar que el organizador se ha modificado correctamente.
Outcome	Tras modificar el organizador, listamos las sesiones públicas y comprobamos que se ha modificado correctamente.
Notes	

## Additional checks

### *Check <#001>*

Description	<i>El sistema cumple la ley de protección de datos, lo que supone que para registrarse en el sistema debe aceptar los términos y condiciones.</i>
Expected	<i>No registrarse en el sistema sin haber aceptado los términos y condiciones.</i>
Outcome	<i>Efectivamente, no permite registrarse sin aceptar los términos, por tanto, obliga a aceptar dichos términos.</i>
Notes	<i>Falta añadir un enlace indicando términos y condiciones, las cuales estamos aceptando</i>

### *Check <#002>*

Description	<i>Comprobar que la pagina informa sobre las cookies que usa.</i>
Expected	<i>Se muestran las cookies que se usan.</i>
Outcome	<i>Debe de mostrarse las cookies en ambos idiomas, tanto en inglés como en español.</i>
Notes	