



## Plan de pruebas – US438733-BotónDeHoy (Juan Vélez Velasco)

Los niveles de prueba que se van a aplicar son los siguientes:

- Pruebas de aceptación: las pruebas de aceptación se definirán siguiendo una estrategia basada en historias de usuario y serán ejecutadas de forma manual.
- Pruebas de integración: Estas pruebas verifican la interacción entre clases. Se llevarán a cabo utilizando los framework *Junit*, *Robolectric* y *Mockito*.
- Pruebas de interfaz: se prueban todos los elementos desde la interfaz, comprobando que la funcionalidad es correcta y que las salidas son las esperadas respecto a las entradas que se han introducido. Para ello, se ha usado el framework Espresso, que nos permite automatizar las pruebas.

**Nota:** Todos los niveles de prueba se ejecutarán sobre la fuente de datos disponible en:  
[https://personales.unican.es/rivasjm/resources/agenda\\_cultural.json](https://personales.unican.es/rivasjm/resources/agenda_cultural.json)

### PRUEBAS DE ACEPTACIÓN

Se muestran las pruebas en base a los criterios de aceptación acordados con el *Product Owner*.

#### *Prueba 00: Éxito*

1. El usuario pulsa el botón de eventos hoy.
2. El sistema abre una nueva pestaña.
3. El sistema muestra la lista de eventos en el día de hoy.
4. El usuario selecciona el botón de opciones para volver al menú principal.
5. El usuario pulsa la opción *menú principal*.
6. El sistema abre la pestaña anterior.

#### *Prueba 01: Día sin eventos*

1. El usuario pulsa el botón de eventos hoy.
2. El sistema abre una nueva pestaña.
3. El sistema muestra la lista de eventos vacía.



## PRUEBAS DE INTEGRACIÓN

Se aplica prueba de métodos, siendo los casos de prueba definidos para cada método los siguientes:

Método **List<Event>eventosHoy()** de la clase **TodayEventsPresenter.java**

Casos de prueba válidos:

Identificador	Entrada	Resultado
IT. 1A	<i>cachedEvents</i> con eventos.	Se retorna la lista de eventos <i>eventosHoy</i> con los eventos que hay en el día que se realiza la prueba.
IT. 1B	<i>cachedEvents</i> con eventos.	Se retorna la lista de eventos <i>eventosHoy</i> vacía porque no hay eventos en ese día.

Casos de prueba no válidos:

Identificador	Entrada	Resultado
IT. 1C	<i>cachedEvents</i> sin eventos.	Se retorna la lista de eventos <i>eventosHoy</i> vacía.



## PRUEBAS DE INTERFAZ DE USUARIO

- A. Pulsar botón de eventos hoy.
  - 1. Se pulsará el botón rojo situado en la parte inferior de interfaz.
  - 2. Se comprobará que se ha cambiado de pestaña.
  - 3. Se comprobará que la lista de eventos es correcta.
- B. Pulsar botón de eventos hoy y volver a la actividad principal.
  - 1. Se pulsará el botón rojo situado en la parte inferior de interfaz.
  - 2. Se comprobará que se ha cambiado de pestaña.
  - 3. Se comprobará que la lista de eventos es correcta.
  - 4. Se abrirá el menú situado en la parte posterior a la derecha que tiene un icono de 3 puntos.
  - 5. Se pulsará la opción 2, que indica que se volverá al menú principal.
  - 6. Se comprobará que se ha vuelto al menú principal.
  - 7. Se comprobará que la lista de eventos en la pestaña principal es la que estaba antes de entrar en la pestaña del botón de hoy.



## Informe de pruebas – US435132-FiltrarPorFecha

En este informe se describe el resultado de ejecutar las pruebas para la historia de usuario FiltrarPorFecha, así como los problemas encontrados durante la realización de las pruebas y las soluciones aportadas.

Autor del plan de pruebas: Adrián García Cubas

Implementador de las pruebas: Juan Vélez

### PRUEBA DE INTERFAZ DE USUARIO

La prueba de interfaz de usuario se ha realizado utilizando los frameworks JUnit y Espresso, se han implementado las pruebas IUT.A, IUT.B, IUT.C., IUT.D, IUT.E., IUT.F. Durante la implementación de dichas pruebas el ejecutor tuvo problemas a la hora de recoger los datos del pop-up implementado en la lógica y al intentar meter valores al calendario para insertar una fecha. El problema de recoger los datos del pop-up no ha sido solventado, el problema de meter valores al calendario para insertar una fecha, se ha solucionado tras una exhaustiva búsqueda por internet hasta dar con un método adecuado.

### PRUEBA DE INTEGRACIÓN

Las pruebas unitarias han sido realizadas sobre los métodos onFiltrarDate() y onEventClicked() de la clase EventsPresenter.java. En la realización de estas pruebas se han usado los frameworks Junit, Mockito y Robolectric.

Durante la realización de las pruebas el ejecutor no encontró ningún problema.