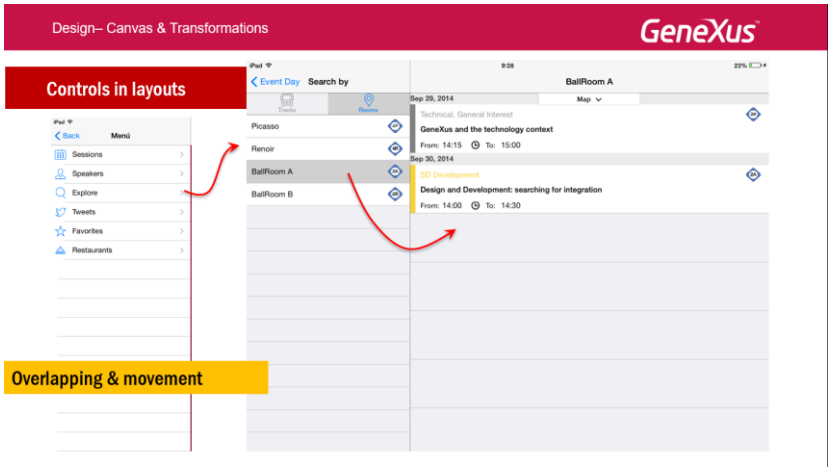
Canvas and Transformations (Part II)

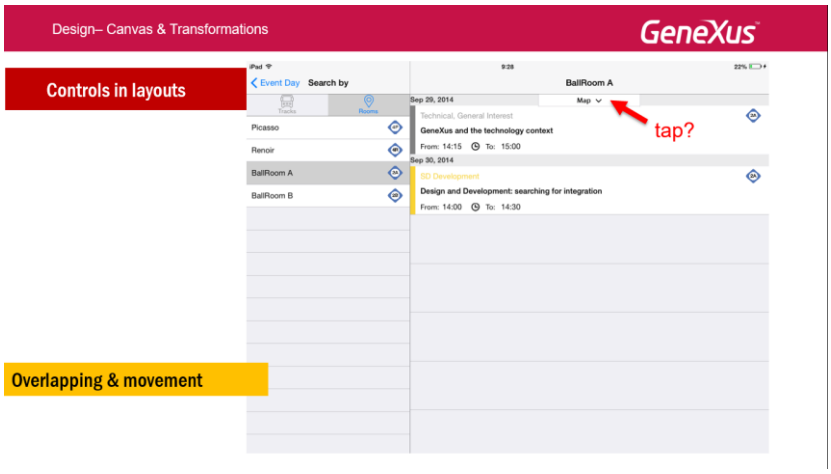


Para ver un caso que ya habíamos mostrado en la demo inicial, donde se va a dar solapamiento de controles con movimiento, para poder agregar entonces esta parte del movimiento.

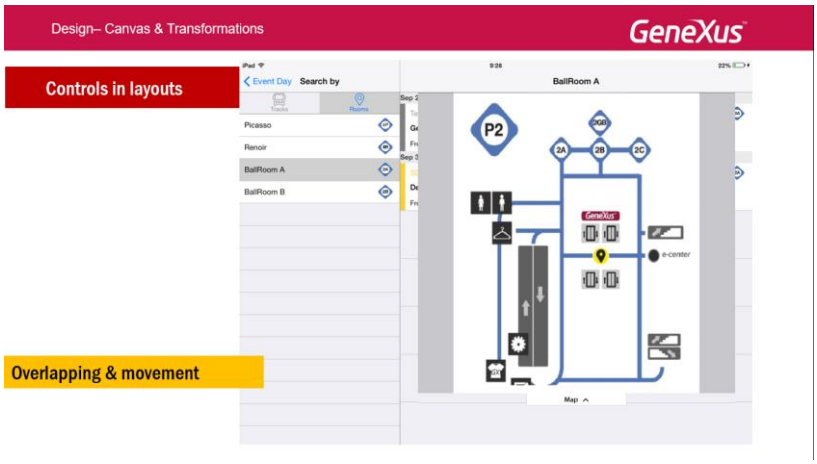
En este caso, desde el menú, esto era en un iPad, invocábamos al objeto EventExplore (un Work With for Smart Devices creado en forma manual, del que solo se utiliza su nodo Detail, con las dos secciones: Tracks y Rooms, que muestran las listas de tracks y las salas, respectivamente). Si desde la lista de salas se elige una –en el ejemplo que estamos viendo, BallRoom A–, se invoca al Detail del WorkWithDevicesRoom para ver los datos de esa sala elegida.

Si tenemos esta aplicación instalada entonces como decíamos en un iPad, vamos a ver que se nos despliega la lista de conferencias agendadas en esa sala. Pero además, vemos arriba lo que parece un botón para expandir, ese botón Map que estamos viendo acá.

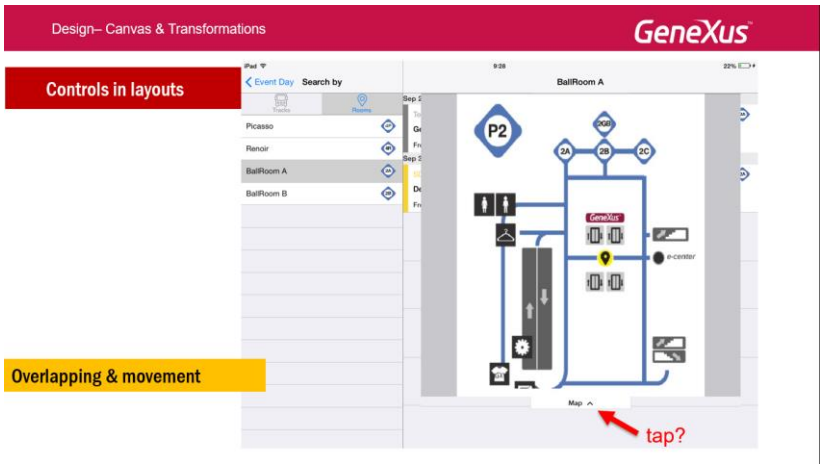
Si hacemos tap sobre ese botón, ¿qué es lo que va a pasar en ejecución?



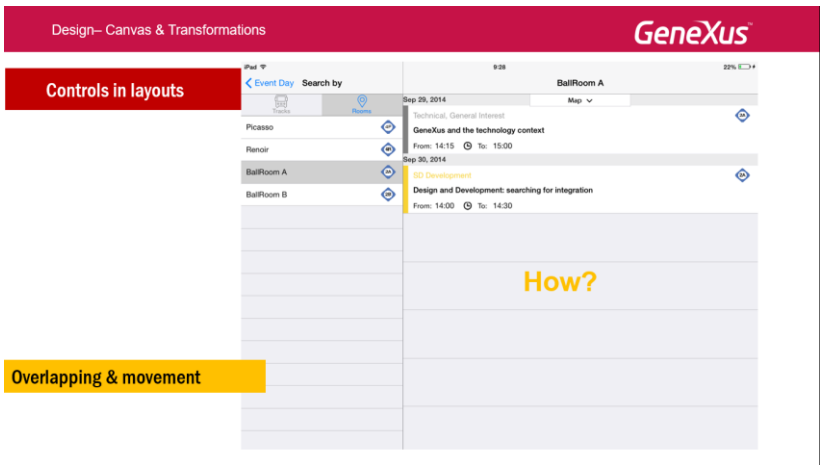
Vamos a ver que se despliega el mapa que muestra dónde se encuentra la sala en el plano del piso, del edificio donde se está desarrollando el evento.



Si ahora volvemos a hacer tap en el botón, que vemos que cambió la orientación de la flechita… ¿qué es lo que sucede?



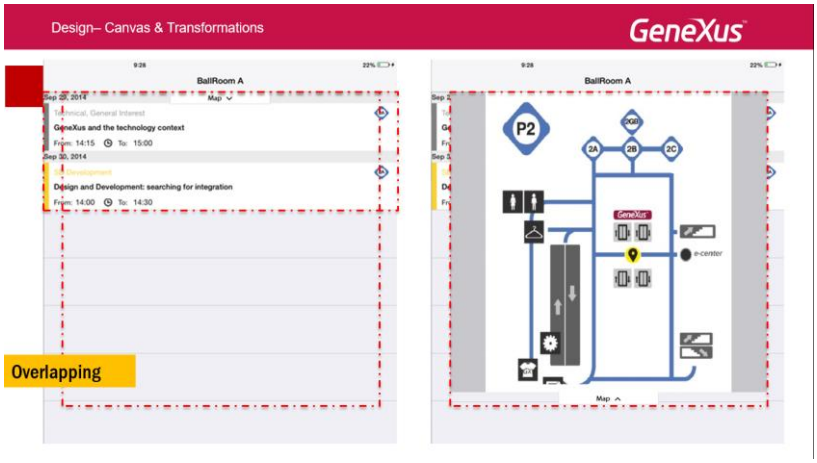
Lo que vamos a ver es que se colapsa el mapa, y estamos viendo solamente el botón.



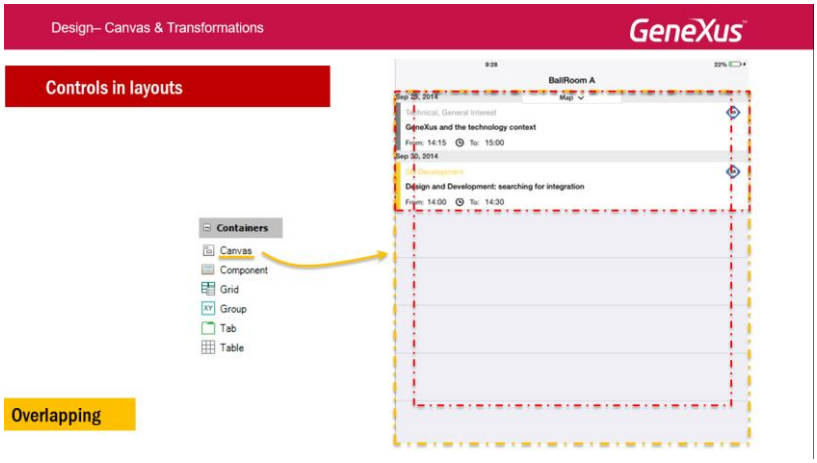
¿Cómo hacemos esto? ¿Cómo lo implementamos?

Bueno, la parte del solapamiento ya la sabemos, la acabamos de ver, y es con un control Canvas. En este layout tenemos, en una misma área de pantalla, la superposición de un grid,

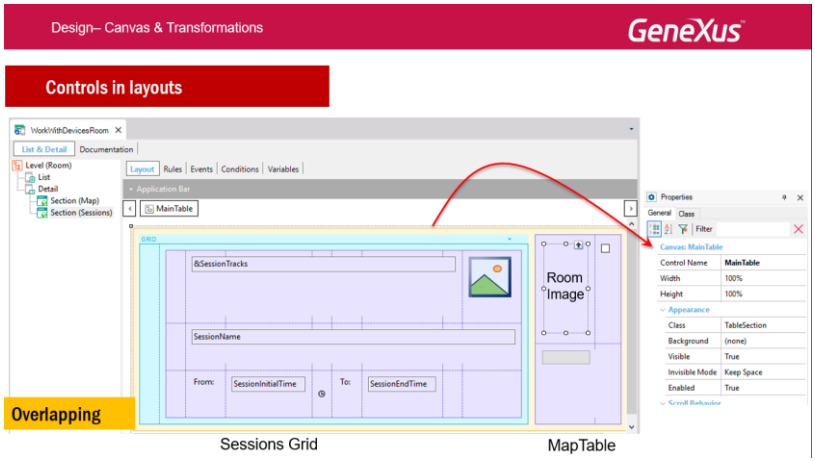
que es el que muestra las conferencias, con una tabla que contiene una imagen con el mapa y un botón; entonces está ese solapamiento.



Entonces por lo tanto lo que vamos a tener que hacer para implementar esto es insertar una tabla Canvas…

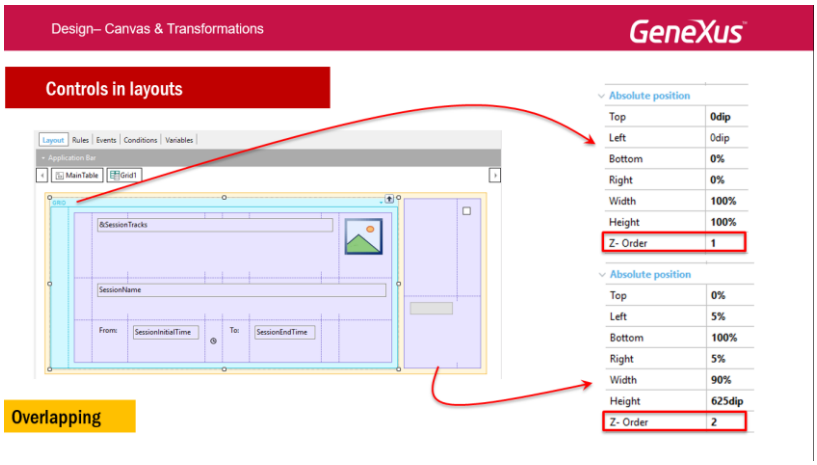


…dentro de la cuál ubicaremos al grid y a la tabla que contiene el mapa y el botón. A esa tabla que contiene el mapa y el botón le vamos a llamar MapTable, ahora vamos a ver para qué la necesitamos.

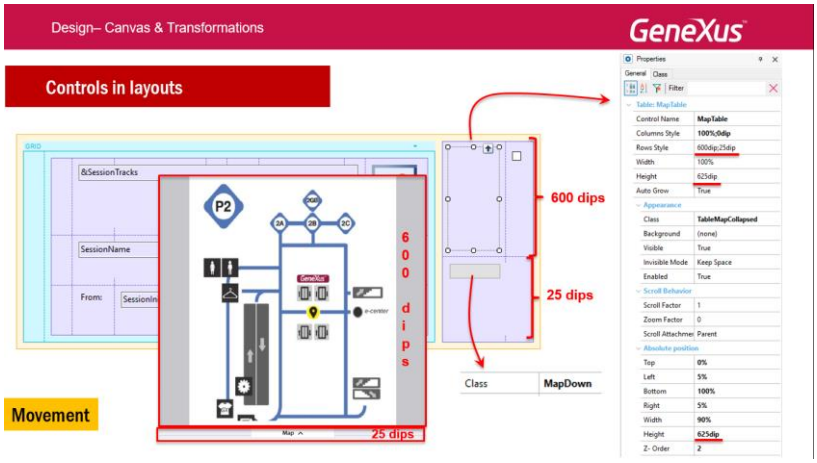


En diseño, no vemos los controles solapados, no es un editor WYSIWYG, justamente para poder trabajar con cada uno en diseño. Pero bueno, como vimos en ejecución, sí aparece como tiene que aparecer, y tenemos Live Editing entonces para ir probando, así que no es necesario tenerlos superpuestos aquí.

Bien. Entonces, además de brindar las posiciones absolutas de grid y tabla, con las propiedades del grupo Absolute Position, es con la propiedad Z-Order que indicamos que el grid estará abajo de la tabla, como vemos aquí. Le pusimos Z-order = 1 al grid, y Z-order = 2 a la tabla.

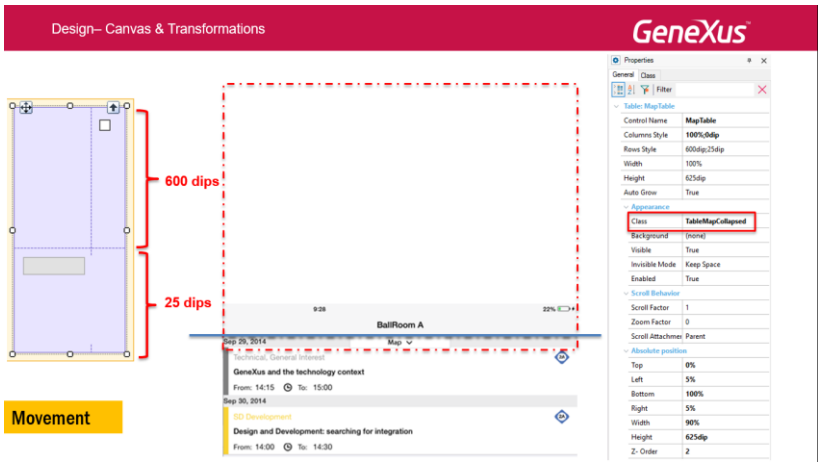


La tabla superpuesta tiene dos filas: una de 600 dips, donde se encuentra el atributo RoomMap que contiene la imagen del mapa de la sala y… otra de 25 dips, donde se encuentra el botón, que tiene asignada una clase del theme, MapDown, que es la que tiene configurada la imagen que se muestra en ejecución.



¿Por qué vemos solamente el botón de expand (Map) cuando invocamos al objeto por primera vez, y lo vemos arriba, como si toda la tabla se hubiese movido 600 dips verticalmente hacia arriba, dejando afuera del espacio del layout a la imagen? En esta línea azul estamos indicando la posición Top del Canvas.

Este comportamiento que acabamos de nombrar lo tiene especificado la clase que le hemos asignado a la tabla: TableMapCollapsed (clase que creamos nosotros, los desarrolladores, dentro del Theme para iPad).

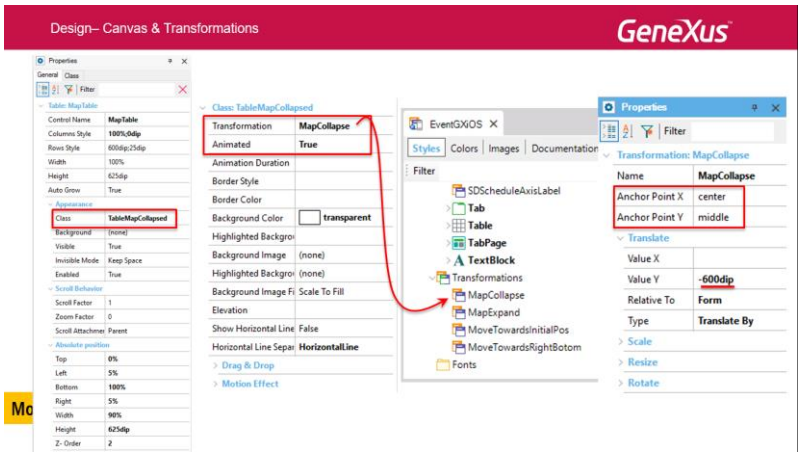


Entonces vemos que esta clase tiene asignada una “transformación”. Esa “transformación” es la que la va a definir el desarrollador dentro del mismo Theme, y es donde se especifica el comportamiento que vemos en ejecución, de **traslación**.

El control que tenga asociada la clase que tiene esta transformación, se va a mover **desde** (Translate by, lo vemos aquí abajo) el punto X, Y indicado (Anchor Point X, Anchor Point Y), en el eje Y, 600 dips hacia arriba (Value Y = -600dip bajo el grupo Translate), tomando como origen de coordenadas (Relative to) el Form (el punto (0, 0) del form equivale al vértice superior izquierdo).

En nuestro caso, la tabla se moverá hacia arriba, 600 dips. Como está al borde de la pantalla, ese movimiento va a dejar afuera a la imagen, como veíamos en la slide anterior. Eso es lo que hace que desaparezca la imagen del mapa de la sala.

Observemos que la clase de la tabla tiene prendida la propiedad Animated, como vemos aquí, en True, y esto es lo que va a hacer que el desplazamiento se vea como tal cuando la tabla esté extendida y se colapse.



Como vemos aquí, es decir cuando está extendida, se hace Tap sobre el botón, y vamos a ver cómo se desplaza de esa manera de forma animada.

Entonces, ¿cómo se programa todo esto?

Cuando la tabla está trasladada hacia arriba (es decir colapsada, lo que equivale a decir **no expandida**), vamos a querer que cuando el usuario presione el botón Map, ésta “se expanda”, lo que no es otra cosa que decir: que aparezca tal cual la vemos en diseño (pero de forma animada, dando la sensación de movimiento).

De la misma manera, el botón debe cambiar su imagen en un caso y el otro. El botón, tiene configurado en diseño la clase MapDown del theme, que podemos ver que es la que que muestra la imagen de esta manera, pero existe otra clase, MapUp, que tiene configurada la otra imagen.

Entonces será en el evento “expand”, asociado al botón, que llamamos también expand, donde se va a programar entonces el comportamiento en un caso y otro (cuando está expandida la información y cuando está colapsada). ¿Cómo? Simplemente cambiándoles las clases a la tabla y al botón, de acuerdo al caso de que se trate.

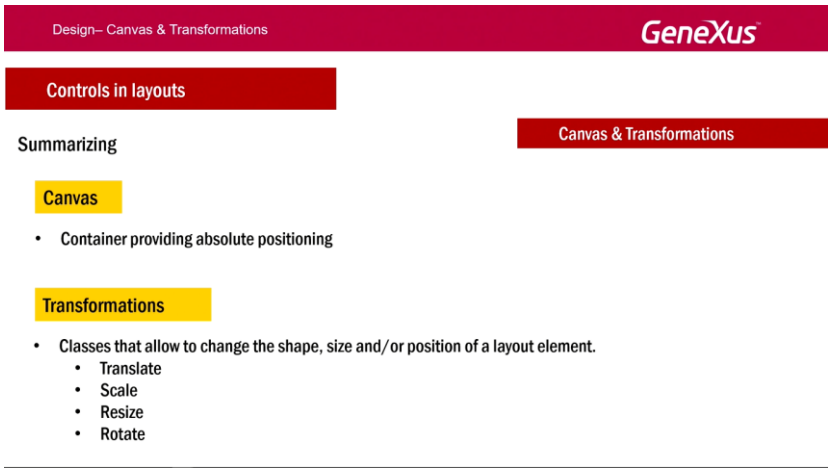
Por ejemplo, cuando la tabla está expandida y se presiona el botón, se va a entrar en este if, y lo que vemos es que se cambia la clase de la tabla que contiene a la imagen de la sala y al botón por TableMapCollapsed, que es la que tiene la transformación de traslación que vimos antes. Y allí es donde se va a dar el efecto buscado, es decir que va a parecer que se traslada hacia arriba, que sube y desaparece la imagen de la sala.

También observemos que se cambia la clase del botón para que salga el que tiene la flechita hacia abajo. Por otro lado, se va a utilizar una variable invisible en la tabla como switch, para prender y apagar, para indicar cuando está colapsada y cuando está expandida.

Para dar el movimiento de expansión de la tabla, que es el inverso, el que veníamos viendo, también vamos a tener que definir una transformación, en sentido inverso, y es análogo.

En resumen, el control Canvas es un contendor de controles, que permite que todo lo contenido pueda asumir posicionamiento absoluto, y de esa manera, eso nos va a permitir, entre otras cosas, superponer controles.

Por otro lado, las transformaciones permiten que un control del Layout pueda ser trasladado, escalado, redimensionado, o incluso rotado, en runtime. Para ello se definen clases especiales bajo el nodo Transformations del theme.



Con esto terminamos con lo que queríamos ver de los controles en los Layouts. Pasemos al siguiente tema.