

Cómo construir aplicaciones con más de una ventana (Screen)

Es muy probable que en muchas de las aplicaciones que hemos creado hasta ahora hayamos necesitado agregar más de una ventana para lograr el correcto funcionamiento de nuestra app. Por ejemplo, si queremos tener un menú inicial de bienvenida que nos permita dar opciones al usuario y de esa forma mostrar distintas funcionales al iniciar la app o bien para programar acciones a distintos botones contenidos en la aplicación en alguna parte específica. Como sea que fuese el caso, agregar más de una ventana nos simplifica la tarea de programar ya que tanto el diseño de la app como los bloques son independientes entre ventanas, por lo que muchas veces podrían considerarse como desarrollos independientes que se conectan por algún evento en común. La idea de este material es mostrar la manera ideal de trabajar las aplicaciones con más de una ventana para así evitar las complicaciones que surgen cuando nos excedemos en el límite máximo de ventanas recomendando que tiene App Inventor.

Antes de comenzar, es importante señalar que App Inventor nos permite agregar ventanas de una forma muy sencilla y es por lo mismo que debemos tener cuidado y no tentarnos en agregar ventana tras ventana. La forma de hacerlo es haciendo clic en el botón que se encuentra en la parte superior de la plataforma, a la derecha del espacio asignado para mostrar el título de la aplicación. Solo se debe asignar un nombre a la nueva ventana que hemos creado, tras lo cual App Inventor nos llevará inmediatamente al diseñador o editor de bloques para comenzar a trabajar.

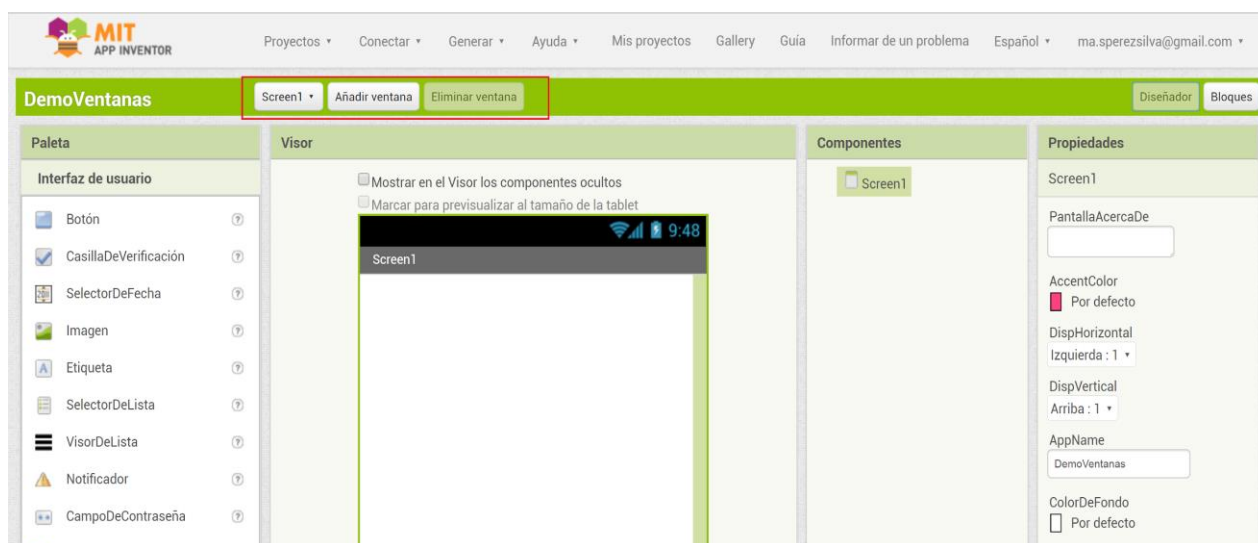


Imagen 1: Menú de opciones para trabajar con ventanas

La forma en la cual está diseñado App Inventor hace que sea bastante costoso en términos de recursos informáticos tener una aplicación con múltiples pantallas. Es por esto que, si creamos una aplicación con 15 pantallas, por ejemplo, es muy probable que surjan problemas al momento de trabajar en la plataforma ya que es fácil exceder las limitaciones de App Inventor. Como regla general, una aplicación no debería tener más de 10 ventanas, por lo mismo es que App Inventor alertará si es que intentamos exceder este límite.

Aplicaciones con múltiples ventanas, cómo proceder:

Es importante mencionar que App Inventor no impedirá que una aplicación exceda el límite sugerido de ventanas, así que, si queremos tener una aplicación con 20 ventanas, vamos a poder hacerlo. El problema es que si procedemos de esa forma existe el peligro de sobrepasar las limitaciones que existen actualmente tanto de procesamiento como de memoria disponibles en la mayoría de los dispositivos Android, inclusive, del navegador donde estemos trabajando. Los límites precisos dependen de varios factores diferentes, por ejemplo, el tamaño total que tendrá la aplicación, el almacenamiento ocupado por los sonidos e imágenes que hayamos agregado al proyecto, la velocidad de internet disponible al momento de trabajar, la configuración que tengamos en el navegador web, entre otros. Mientras más ventanas agreguemos a la aplicación nos vamos a encontrar con que uno o más de estos factores inciden en el correcto desarrollo que podamos tener, generando consecuencias que pueden ser muy molestas. Ejemplo de ello es que el Editor de Bloques podría ejecutarse cada vez más lento o incluso que App Inventor sea incapaz de construir el proyecto. En los casos más extremos se puede perder la capacidad de App Inventor para guardar o cargar el proyecto, lo que nos impediría hacer modificaciones futuras al proyecto.

Si encontramos que es imposible que nuestra aplicación tenga menos de 10 ventanas, lo recomendable es guardar el proyecto en el computador. Cada vez que realicemos un cambio importante lo ideal es exportar el proyecto en formato Aia y mantener una copia en algún dispositivo local. Una buena idea sería guardar distintas versiones del proyecto ya que ante cualquier problema que pueda surgir con App Inventor, existe un historial de versiones guardado en el computador por lo cual solo habría que importar alguna versión que sabemos funciona sin problemas. Como recomendación general, nunca está demás tener respaldos extras del trabajo que estamos realizando, sobre todo si son proyectos grandes y complejos en los cuales se han invertido varias horas de trabajo.

Cómo diseñar una aplicación para limitar el número de ventanas

A veces tenemos la sensación de que la aplicación que estamos creando va a necesitar una gran cantidad de ventanas. Si este es el caso, podemos rediseñarla para no usar tantas. Supongamos que queremos hacer una aplicación que tenga distintos botones que al ser presionados muestre distintos tipos de imágenes. La funcionalidad que tienen los botones es la misma para todos, lo único que cambia es que las imágenes que se van a mostrar serán distintas según el tipo de botón que sea presionado. Una forma válida de codificar esto sería tener una ventana principal que tendrá cada uno de los botones y por cada uno de estos botones crear una nueva ventana que permita mostrar la imagen asociada al botón que fue presionado. Este método obliga a tener tantas ventanas como botones e imágenes vayamos a usar. ¿Qué pasaría si tengo 15 botones? Otra forma de hacer este mismo proceso podría usando 1 o 2 ventanas solamente y comenzar a trabajar cambiando las propiedades de la imagen que vamos a mostrar según cual sea el botón presionado. Esta es una de las distintas soluciones que podríamos usar trabajando con muy pocas ventanas. Entre un usar 2 ventanas en vez de 15 hay una gran diferencia. Esta misma idea puede aplicarse a otro tipo de elementos. Supongamos que queremos crear una aplicación que en cada ventana muestre una etiqueta con un mensaje propio, en lugar de crear varias ventanas y que en cada una de ellas agregamos la etiqueta con el mensaje, se podría tener solo una ventana con una etiqueta e ir cambiando el mensaje según cuál sea la ventana que estemos abriendo. El esquema muestra cómo abordar este ejercicio desde ambas miradas:

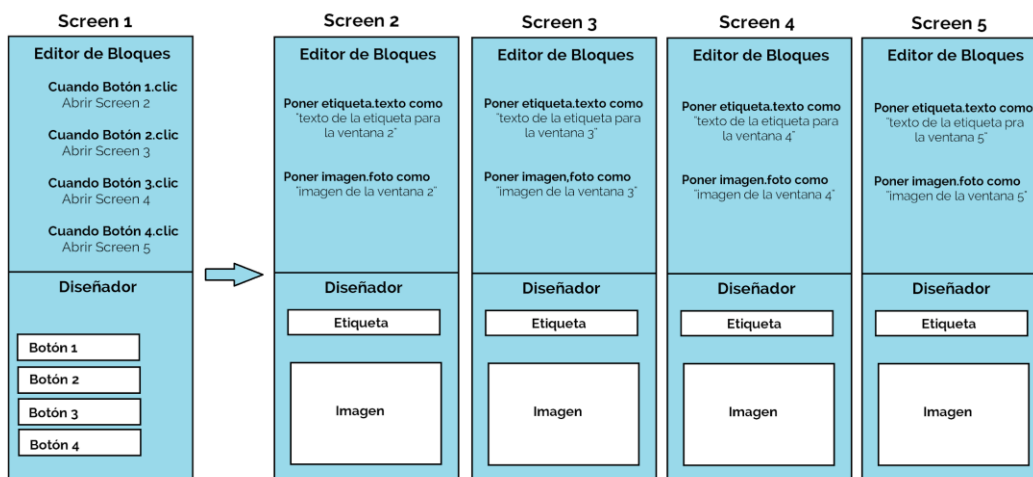


Imagen 2: Diseño tradicional con múltiples ventanas

Como podemos ver en este esquema, nuestra aplicación consta de 5 ventanas, la primera de ellas (Screen1) es utilizada para mostrar un menú con 4 opciones distintas. Al momento de presionar alguno de estos botones, la aplicación nos llevará a una nueva ventana (screens 1, 2, 3 y 4) que será la encargada de mostrar el contenido asociado a esa opción. Como se mencionó anteriormente, este tipo de diseño implica que, si tenemos 20 botones mostrando 20 opciones distintas, la aplicación será bastante costosa a nivel de recursos. ¡Podemos hacerlo de otra forma! El siguiente esquema muestra un diseño diferente que tiene la misma funcionalidad que la aplicación representada en el esquema anterior, con la gran diferencia que esta vez solo estaremos usando 2 ventanas.

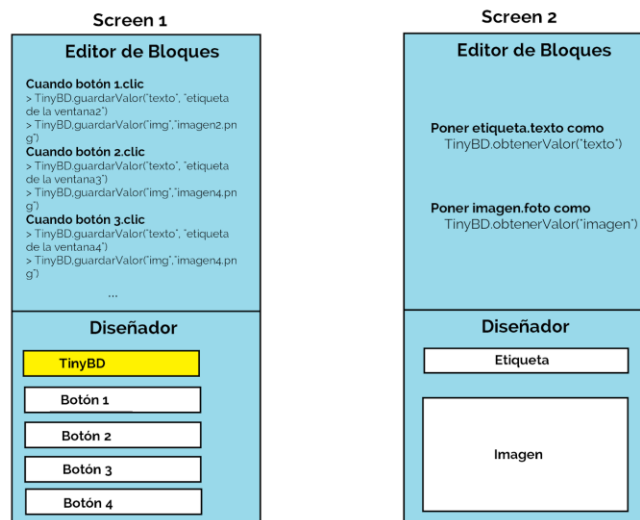


Imagen 3: Diseño simplificado utilizando base de datos local.

La principal diferencia en este tipo de diseño es que necesitamos apoyarnos de recursos de almacenamiento para saber cuál es el contenido correcto a mostrar dependiendo de la opción que indicamos en la ventana que contiene el menú principal (screen 1). Si el valor a enviar a la segunda ventana es uno solo, podemos apoyarnos del bloque "Abrir otra pantalla con un valor inicial" que es el bloque que se muestra a continuación:

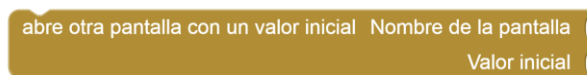


Imagen 4: Bloque para abrir otra ventana enviando información a la ventana que queremos abrir

El funcionamiento de este bloque es sencillo. Al momento de abrir una nueva ventana podemos pasarle información que queramos compartir entre ventanas. De esta forma cuando estemos trabajando en la nueva ventana tendremos un valor que podremos rescatar y utilizarlo para definir el funcionamiento de la ventana. Por ejemplo, si estamos realizando un juego podríamos enviar el puntaje.

En el esquema que se muestra en la imagen 3 podemos ver que la información que vamos a compartir entre ventanas corresponde al nombre de la imagen que queremos mostrar y el texto que vamos a mostrar en la etiqueta. Es por eso que resulta mucho más conveniente utilizar un base de datos local y guardar el valor de los datos que queremos mostrar. Si en el menú principal hacemos clic en el botón 3, escribimos en la base de datos el nombre de la foto que se debería mostrar al hacer clic en ese botón, así como el texto a mostrar en las etiquetas. Nuestra segunda ventana lo único que tendría que hacer es recuperar los valores almacenados en la base de datos y mostrarlos en la respectiva etiqueta e imagen. Como vemos, hemos utilizado tan solo 2 ventanas en vez de las 5 que teníamos en la idea propuesta originalmente.