

Tutoriales para Justinmind Prototyper

Nivel básico

Este tutorial está recomendado para usuarios que se enfrentan al reto de usar Justinmind Prototyper por primera vez.

En los siguientes apartados se explica cómo usar Justinmind Prototyper para dibujar wireframes y hacerlos clicables. También se explica cómo dibujar mockups y como hacerlos interactivos. Finalmente, se describe cómo exportar esas pantallas en un HTML interactivo o en un documento.

Introducción

Justinmind Prototyper es una herramienta de prototipado rápido pensada para que un usuario sin conocimientos de programación pueda definir cómo quiere una cierta aplicación. Permite dibujar las pantallas de cualquier tipo de aplicación y darles interactividad. Esa interactividad se puede probar de manera instantánea con un solo clic de forma que se pueden probar los cambios de forma inmediata sin necesidad de escribir ni una sola línea de código. Estas pantallas se pueden acompañar de textos y diagramas para después generar un documento que sirva para definir cómo va a ser una cierta aplicación.

El prototipo se puede exportar a formato HTML para que cualquiera lo pueda simular y dar su opinión sobre la aplicación que se tiene pensado desarrollar. De esta forma, se pueden crear prototipos interactivos y todo lo detallados que se desee con el objetivo de capturar las opiniones de los futuros usuarios de esa aplicación y evitar cambios en fases críticas del proyecto.

Los prototipos hechos con Justinmind Prototyper se pueden publicar en Justinmind Usernote para que los usuarios puedan realizar anotaciones online y de forma colaborativa. De esta manera se pueden gestionar todas estas opiniones e ir refinando el prototipo hasta que todos tienen claro cómo va a quedar al final la aplicación.

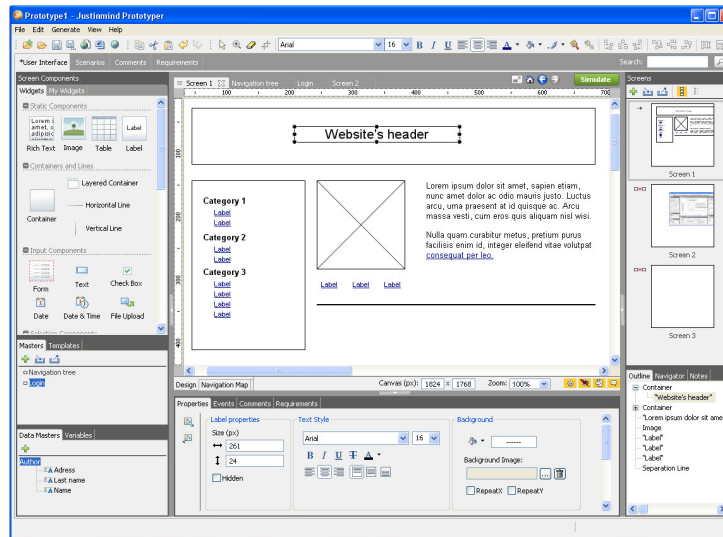
Justinmind Usernote también permite realizar tests de usuario al principio de un proyecto y se integra con las principales herramientas de test automático online del mercado.

En este tutorial y en los siguientes se enseña como usar Justinmind Prototyper para hacer wireframes, mockups y prototipos interactivos rápida y fácilmente.

Familiarizarse con el área de trabajo

La interfaz de Justinmind Prototyper está orientada a la creación de prototipos visuales e interactivos de una forma fácil y rápida. Contiene elementos comunes a otras

herramientas de dibujo para hacer aún más fácil su uso. A continuación veremos las principales secciones que trataremos en este tutorial.



A: Barra de menú, **B:** Barra de herramientas, **C:** Pestaña de Interfaz de usuario, **D:** Paleta de componentes, **E:** Lienzo o área de trabajo, **F:** Lista de pantallas, **G:** Propiedades del elemento seleccionado, **H:** Contenido de la pantalla actual, **I:** Pestañas de situación, **J:** Botón de simulación.

En este tutorial sólo se tienen en cuenta las secciones descritas en el párrafo anterior, el resto de secciones se explican en los tutoriales de nivel medio y avanzado.

Barra de menú

La barra de menú contiene menús para realizar tareas. Los menús están organizados por temas. Por ejemplo, el menú Generar contiene comandos para generar HTML, documentos o imágenes.

Barra de herramientas

La barra de herramientas contiene opciones de edición rápida del componente que esté seleccionado en el lienzo. También contiene accesos rápidos a las opciones más comunes y la selección de herramientas de edición.

Pestaña de Interfaz de Usuario

Esta pestaña engloba todo lo que hace referencia a la edición de pantallas y la interacción en el prototipo. El resto de pestañas se explican en otros tutoriales.

Paleta de componentes

La paleta de componentes contiene todos los elementos que se pueden añadir a una pantalla. Los componentes se pueden añadir arrastrando y soltando al lienzo o pulsando con el ratón sobre el elemento y volviendo a pulsar en la posición del lienzo que se desee.

Lienzo o área de trabajo

El lienzo muestra el contenido de la pantalla activa y permite añadir y modificar componentes en esa pantalla.

Lista de pantallas

La lista de pantallas contiene todas las pantallas de las que está compuesto el prototipo. Para ver el contenido de una pantalla en el lienzo basta con hacer doble clic sobre ella en la lista.

Propiedades del elemento seleccionado

En esta sección se pueden editar las propiedades de estilo gráfico del componente seleccionado. También incluye un apartado donde añadir comentarios a ese mismo componente. El resto de apartados se explican en otros tutoriales.

Contenido de la pantalla actual

En este apartado se listan los componentes que forman parte de la pantalla activa en el lienzo. Permite seleccionarlos de manera individual y modificarlos igual que en el lienzo.

Pestañas de situación

Las pestañas de situación sirven para cambiar de forma rápida entre las últimas pantallas abiertas. La pestaña seleccionada indica la pantalla activa en el lienzo.

Botón de simulación

El botón de simulación permite probar la navegación del prototipo como si fuera un usuario final.

Definir el contenido de una pantalla

Arrastre y suelte elementos de la paleta de componentes al lienzo para crear contenido en una pantalla. Una vez introducidos en el lienzo, los componentes se pueden redimensionar y mover para organizarlos dentro de la pantalla. Para mover un componente basta con seleccionarlo y arrastrarlo a la zona del lienzo que desee. También puede mover los elementos con las teclas de dirección. Si mantiene la tecla “shift” pulsada mientras mueve un elemento con las teclas de dirección el elemento se moverá más deprisa.

Justinmind Prototyper incluye un sistema de alineación automática mediante unas guías de color azul que le ayudan a organizar el contenido de su pantalla. También se puede aplicar una retícula desde el menú “Ver-> Malla”. Asimismo también se pueden crear guías pulsando con el ratón sobre las reglas. Para quitar estas guías basta con seleccionarlas y pulsar la tecla de “Suprimir”.

Los elementos de una pantalla guardan un orden de profundidad entre sí de forma que algunos elementos pueden tapar a otros. Puede ver el orden de profundidad en la vista de Contenido. Para cambiar el orden de un elemento pulse con botón derecho ese elemento o selecciónelo en la vista de contenido y seleccione la opción del menú “Ordenar” que prefiera.

El componente de Imagen le permite indicar en su wireframe las zonas de la pantalla que van a ser ocupadas por imágenes. Puede sustituir estas representaciones por imágenes reales seleccionando un fichero desde la vista de propiedades. Desde esta misma vista también puede cambiar las propiedades de estilo (como por ejemplo el color, el borde o la fuente de texto) del componente seleccionado.

Para editar el contenido de una pantalla diferente a la pantalla activa ha de hacer doble clic sobre la pantalla que desee editar en la lista de pantallas. Tras hacer doble clic el contenido de esa pantalla se cargará en el lienzo. Si desea crear más pantallas en el prototipo pulse el botón de “+” situado encima de la lista de pantallas. Para cambiar el nombre a una pantalla seleccione la pantalla de la lista que desee, pulse con el botón derecho del ratón y seleccione la opción “Propiedades” del menú contextual. Si lo que desea es eliminar una de las pantallas de la lista selecciónela con el botón derecho del ratón y seleccione la opción “Borrar”.

Nota importante: toda acción que realice la puede deshacer o volver a hacerla usando las opciones de “deshacer” y “rehacer” del menú de “Edición”.

Entre la parte inferior del lienzo y la vista de propiedades se sitúa una barra de herramientas relacionadas con el área de trabajo. Por un lado hay dos campos de texto que indican la resolución actual del área de trabajo. Puede aumentar o disminuir esa resolución editando directamente esos valores. A la derecha de la resolución del lienzo se encuentra el indicador de zoom. Puede indicar el nivel de zoom mediante este desplegable.

También puede copiar y pegar componentes de una pantalla en la misma pantalla o en otras usando las opciones del menú Edición o los atajos de teclado CTRL+C o CTRL+V. También puede duplicar elementos de pantalla arrastrándolos mientras mantiene la tecla CTRL pulsada.

Importar contenido externo al Prototyper

Justinmind Prototyper proporciona una serie de utilidades para incluir contenido existente fuera de Justinmind Prototyper a un prototipo. Se pueden incluir textos, imágenes y contenido web como vídeos, flash y mapas de Google maps.

Para añadir una imagen desde el sistema de ficheros es suficiente con arrastrar el fichero de imagen al lienzo. También se puede importar esa imagen desde la vista de propiedades de un elemento Imagen arrastrado desde la paleta de componentes. Otra manera de incluir imágenes en un prototipo es arrastrando y soltando directamente desde otras herramientas o incluso desde navegadores de Internet. Por ejemplo se puede copiar una selección desde el PhotoShop y pegarla como imagen en Prototyper.

También se pueden importar textos desde otras aplicaciones arrastrando y soltando o usando las funciones de copiar y pegar. Dependiendo de la aplicación, Justinmind Prototyper importará los estilos de texto o no.

Visualizar contenido de Internet en un prototipo es muy fácil usando los “Widgets avanzados” (están situados en el último grupo de widgets de la paleta de componentes”. Por ejemplo se puede reservar un espacio de la pantalla activa del prototipo para que se visualice una cierta dirección de Internet. Para hacer eso arrastre el componente URL al lienzo e indique una dirección de Internet en la vista de propiedades. También puede visualizar un video de Youtube o una localización de Google Maps copiando el HTML de estos sites en las propiedades del componente HTML.

Componentes de pantalla interactivos

La mayoría de componentes de la paleta son interactivos. Eso significa que una vez introducidos en el lienzo pueden interactuar con el usuario si se pulsa el botón de simular. Por ejemplo es el caso del componente Texto de formulario. Si arrastra este componente al lienzo y pulsa el botón de “Simular” entrará en modo simulación, en este modo todos los componentes de pantalla que sean interactivos se activan y, en este caso, si el usuario pulsa sobre el campo de texto podrá introducir un texto como si fuera una aplicación real. Otros componentes interactivos son los desplegables. Los valores de un desplegable se indican desde la vista de propiedades.

Existen más componentes interactivos pero hay dos que son especiales: el menú y el árbol de navegación. Para crear un menú arrastre el componente de menú al lienzo e indique el tamaño que va a ocupar la barra de menú. Después puede arrastrar elementos de menú y colocarlos en la posición que desee dentro del menú. El componente de árbol funciona muy parecido pero en este caso hay que arrastrar nodos de árbol. Los dos componentes se pueden personalizar mediante la vista de propiedades.

Todos estos componentes tienen interactividad implícita y se activan al pulsar el botón de “Simular”.

Wireframes clicables

Si se ha entendido lo explicado en párrafos anteriores se está preparado para construir wireframes clicables fácilmente con Justinmind Prototyper. Primero dibuje los contenidos de su wireframe usando las figuras de rectángulo, texto, label e imagen. Seguidamente marque con un estilo especial (típicamente azul y subrayado) los textos que vayan a ser clicables. Por último arrastre los elementos de pantalla que desee sobre la pantalla que quiera que se muestre al pulsar sobre ese elemento y se creará automáticamente un link. Llegados a este punto, si pulsa el botón de “Simular” no sólo podrá interactuar con los elementos de la pantalla sino que además podrá simular la navegación entre las diferentes pantallas del prototipo.

Truco: si quiere indicar que solo un trozo de un componente de texto rico es clicable sitúe un componente de “Mapa de imágenes” sobre la zona que ocupe ese trozo de texto y arrástrelo sobre la pantalla que desee mostrar. Los “Mapas de imágenes” no

se muestran al pulsar el botón de simular pero sí conservan las interacciones en el área que están ocupando.

A medida que defina la navegación entre las diferentes pantallas de su prototipo se puede dar el caso de no saber cómo está quedando esa navegación. Justinmind Prototyper ofrece varias opciones para moverse dentro de un prototipo. Las primeras son las utilidades de “navegador” formadas por las opciones de “ver pantalla inicial”, “anterior” y “siguiente” que simulan el comportamiento de un navegador de Internet y están situadas a la izquierda del botón de “Simular”. Llegados a este punto conviene aclarar el concepto de “pantalla inicial”. En todo prototipo hay una pantalla inicial que sirve para introducir al usuario que está probando el prototipo al resto de pantallas. Ésta pantalla inicial siempre sale la primera en la lista de pantallas y está remarcada con una flecha. Puede cambiar la pantalla inicial por otra con la opción de “Marcar como pantalla inicial” del menú contextual de la lista de pantallas.

Otra manera de ver la estructura de un prototipo es cambiando la vista de la pantalla actual de “Diseño” a “Navegación”. Para cambiar a modo “Navegación” pulse la pestaña “Navegación” situada entre la parte inferior del lienzo y la vista de propiedades en la parte izquierda de la pantalla. En este modo se muestran las pantallas a las que se puede navegar desde la pantalla activa. Si se pulsa sobre la pantalla activa se despliega el primer nivel de navegación de esa pantalla. Si se pulsa sobre una de las pantallas de ese nivel se despliega otro nivel de navegación indicando a qué pantallas se puede navegar desde la pantalla seleccionada. Este mapa de navegación se actualiza cada vez que se crea una navegación en la vista de diseño.

Los elementos de menú y los nodos de un árbol también pueden navegar a otras pantallas. Lo único que hay que hacer es arrastrarlos sobre la pantalla que se quiera mostrar al hacer clic sobre ellos. También se puede arrastrar la pantalla sobre el elemento y el efecto es el mismo. En realidad, cualquier elemento de pantalla se puede arrastrar a una pantalla de la lista para que sea clicable.

Mockups Interactivos

Dentro de todo proceso de definición de una aplicación se incluye una fase de diseño gráfico. En esa fase se proponen varios estilos y diseños de las interfaces gráficas de la aplicación aunque hay casos en los que ese diseño ya viene marcado por guías de estilo corporativas. Los diseños se suelen hacer con herramientas de diseño gráfico especializadas como por ejemplo Photoshop y se suelen presentar varias de esas imágenes representando una serie de pantallas tipo. El problema de estas imágenes es que son estáticas, es decir, no se puede interactuar con ellas, y se pueden perder de vista la navegación entre esas pantallas. A estas propuestas gráficas se les suele denominar mockups.

Justinmind Prototyper ofrece una forma muy sencilla de hacer que estas imágenes sean interactivas. Primero añade las imágenes de las pantallas en pantallas del prototipo diferentes (recuerde que puede arrastrar directamente los ficheros de

imagen o que puede copiar y pegar contenido desde otros programas de diseño). Después arrastre el componente de mapa de imágenes sobre las zonas de la imagen que representen botones o cualquier elemento que provoque un cambio de pantalla al pulsar encima de él. Arrastre los mapas de imágenes de la pantalla sobre las pantallas de lista a las que quiera que navegue. Haga lo mismo con el resto de pantallas y ya podrá simular la navegación entre esas imágenes. También puede colocar campos de formulario sobre los campos de formulario dibujados en la imagen, quitarles el borde y ajustar el tamaño y la fuente de forma que “cobren vida” al darle al botón de simular.

En los siguientes tutoriales se explica cómo definir otro tipo de interacciones diferentes a la de “clic-> muestro otra pantalla” y también se enseña cómo mostrar contenido dinámico dentro de la misma pantalla. Recuerde que también se pueden usar imágenes en esos casos.

Compartir tu prototipo

El prototipo es un vehículo de comunicación entre diferentes personas para definir una aplicación. Cuanto más parecido sea a la aplicación final más claro quedará a todas las personas implicadas en un proyecto cómo va a ser esa aplicación cuando esté acabada de construir. Al ser un modo de explicar cómo va a ser una aplicación, interesa que el prototipo lo vean la mayor cantidad de personas clave dentro de un proyecto para que también puedan dar su opinión. Lo importante de un prototipo es, por un lado, que refleje lo mejor posible cómo va a ser la aplicación final para que todos lo sepan y por otro detectar cambios en la definición antes de que se empiece a construir la aplicación. Justinmind Prototyper ofrece varias formas de enseñar o distribuir el prototipo entre personas clave de un proyecto.

La primera forma de enseñar un prototipo es desde el propio Justinmind Prototyper. Usando el botón de Simular (F5) y el de pantalla completa (F11) podemos revisar junto a los usuarios el prototipo. La principal ventaja de esto es que se pueden aplicar los cambios que va diciendo el usuario en ese mismo momento y hacer que los valide inmediatamente. La desventaja es que hay que ir haciendo reuniones con esos usuarios.

Otra manera de enseñar un prototipo es exportarlo a formato HTML. Este formato se puede abrir desde cualquier navegador de Internet así que se puede distribuir entre los usuarios clave, que éstos lo ejecuten y lo prueben y que nos manden sus opiniones. La ventaja de este sistema es que los usuarios pueden probar el prototipo y realizar sus aportaciones cuando quieran. La desventaja es que, por un lado, se ha de preparar el prototipo para guiar al usuario en lo que ha de ir haciendo y, por otro, que hay muchas opciones de que el filtro de spam de los correos no permita recibir correos que tienen código HTML adjunto.

Aún existe otra forma más de que los usuarios revisen y prueben un prototipo: Justinmind Usernote. Este servicio permite publicar un prototipo accesible desde Internet e invitar a usuarios clave a que lo vean usando sus navegadores de Internet.

Los usuarios también pueden realizar anotaciones sobre el prototipo incluso de forma colaborativa y así todos los comentarios quedan centralizados. Otra opción es integrar ese prototipo accesible desde Internet con herramientas de test remoto y realizar tests de usuario sobre el prototipo.

Además se puede exportar toda la información del prototipo en un documento de especificaciones para integrarlo en los diferentes procesos documentales de una organización o simplemente para hacer un documento que luego se pueda firmar.

Nivel Medio

Este tutorial está orientado a usuarios que ya han usado alguna vez Justinmind Prototyper y pretenden profundizar en cómo hacer prototipos con alta interactividad. Para aquellos que nunca han usado Justinmind Prototyper se recomienda que lean primero el tutorial de Nivel Básico.

Contenido dinámico

En este apartado se profundiza sobre las capacidades de Justinmind Prototyper para simular contenido dinámico dentro de una pantalla. Por contenido dinámico se entiende todo aquel contenido que cambia en función de las acciones que haga el usuario sin cambiar de pantalla. A continuación se explican una serie de casos como ejemplo usando principalmente el componente de Paneles Dinámicos. La idea que se sigue en estos ejemplos es fácilmente extrapolable a otros casos que se puedan dar durante la creación de un prototipo.

Pestañas

Una de las maneras más usadas de organizar gran cantidad de información en una misma pantalla es agrupar esa información en diferentes “pestañas”. Cada pestaña muestra un contenido diferente cuando el usuario pulsa sobre ella. El componente de Paneles Dinámicos es especialmente útil para este caso. Este componente permite agrupar contenido en varios paneles o capas y decidir qué panel se está viendo en cada momento.

Para crear un componente de pestañas primero debemos arrastrar el componente de Paneles Dinámicos al lienzo. Una vez en el lienzo el componente muestra el área reservada para el panel activo en ese momento, el nombre del panel o capa y un icono de “+” para añadir más paneles. Pulsaremos el icono tantas veces como pestañas queramos simular reservando así un panel para el contenido de cada pestaña, por ejemplo tres paneles. En cada panel se coloca el contenido de las respectivas pestañas (nota: para seleccionar un panel primero hay que seleccionar el componente de paneles dinámicos y después la etiqueta gris que corresponda al panel o capa que se desee) arrastrando componentes de pantalla directamente encima del panel. Seguidamente dibujaremos las pestañas, en este caso de ejemplo usaremos textos y les daremos forma de pestaña.

Para dar forma de pestaña a un texto primero lo alinearemos al centro en vertical y horizontal. Después le indicaremos que tiene un borde y personalizaremos ese borde

indicando que el borde inferior no es visible. Finalmente podemos redondear un poco los bordes para darle más aspecto de pestaña. Una vez hecha la primera pestaña podemos copiarla y crear las dos que faltan. Hasta el momento tenemos preparadas las pestañas por un lado y el contenido que ha de mostrar cada una por otro.

Ahora vamos a indicar que al pulsar cada una de las pestañas se va a mostrar el contenido que corresponde a esa pestaña. Primero seleccionamos una de las pestañas que hemos dibujado y seleccionamos la vista de Eventos situada al lado de la de Propiedades. En esta vista es donde se indica toda la interacción que podemos definir entre el usuario y el elemento seleccionado cuando se ejecute la simulación. Si pulsamos en el botón de Añadir interacción se abrirá un diálogo para especificar qué va a pasar cuando el usuario pulse sobre esa pestaña. Lo que queremos que pase es que se muestre el panel con el contenido de esa pestaña así que seleccionamos la opción de Mostrar/Esconder, seleccionamos el panel que queremos mostrar e indicamos que lo que queremos hacer es mostrarlo (nota: no hace falta indicar que los demás paneles se han de ocultar porque sólo puede haber un panel visible). Hacemos lo mismo con las otras dos pestañas indicando los paneles que han de mostrar.

Finalmente si pulsamos sobre el botón de Simular veremos que se muestra un contenido diferente en función de la pestaña seleccionada.

Truco: si además de cambiar el panel queremos que se remarque qué pestaña está seleccionada en cada momento podemos mover las pestañas dentro de los paneles y cambiar el estilo de la que representa cada panel.

Formularios dinámicos

En algunos casos los formularios de entrada de datos de una aplicación pueden llegar a tener muchos campos. Hay varias maneras de organizar los campos de forma que no inunden de información al usuario. Una de esas maneras es mostrar sólo los campos imprescindibles y dar la opción al usuario de mostrar los campos que son opcionales. A continuación veremos cómo simular este tipo de formularios con Justinmind Prototyper.

Usaremos la misma idea que con las pestañas pero más simplificada. Primero dibujaremos algunos campos que consideremos imprescindibles. Después arrastraremos al lienzo el componente de Paneles Dinámicos y crearemos dos paneles o capas. En el primer panel pondremos una etiqueta que se titule “Mostrar campos opcionales”, en el segundo panel otra etiqueta que sea “Ocultar campos opcionales” y debajo de ésta añadiremos unos cuantos campos de entrada de formulario.

Definiremos un evento en la etiqueta “Mostrar Campos opcionales” de forma que al clicar el usuario encima de esa etiqueta se muestre el panel que contiene los campos opcionales. En ese panel definiremos un evento sobre la etiqueta “Ocultar campos opcionales” que muestre el otro panel. Así de sencillo.

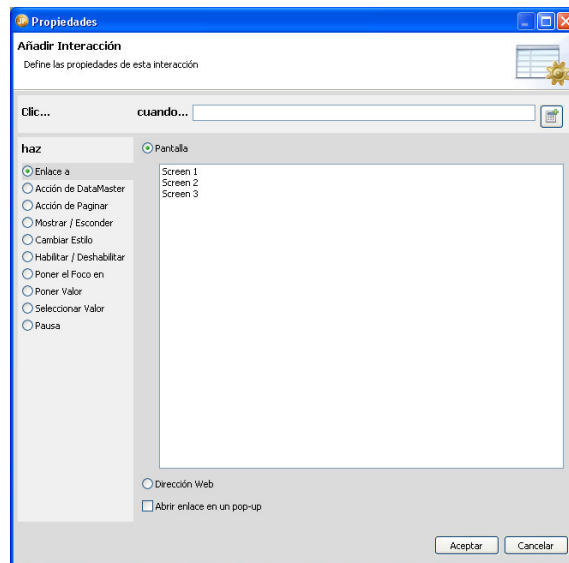
En el tutorial de nivel avanzado se explica cómo hacer que el contenido dinámico no pise el resto del contenido usando la propiedad de layout de los paneles dinámicos.

Interacciones ricas

Ya hemos visto en este tutorial y en el anterior alguna de las interacciones que podemos simular con Justinmind Prototyper. A continuación entraremos más en detalle en qué eventos y acciones en concreto podemos simular.

La definición de eventos se concentra en la pestaña de Eventos situada al lado de la de Propiedades. Para crear un evento se necesita, por un lado, elegir qué acción del usuario va a provocar el comportamiento que queremos, y por otro elegir qué queremos que pase. Para elegir qué acción de usuario queremos se dispone de un menú en la parte izquierda. Es importante saber que un mismo componente puede reaccionar a diferentes acciones del usuario, así que se puede definir que, por ejemplo, un botón reaccione ante un clic y también mientras se pasa el cursor por encima. El menú de acciones del usuario muestra, además, el número de acciones que hay definidas para cada disparador.

Una vez escogida la acción del usuario podemos crear una interacción pulsando el botón de “Añadir interacción”. Al pulsarlo aparece un cuadro de diálogo en el que podremos configurar qué pasará cuando el usuario realice la acción escogida. En la parte izquierda de este diálogo hay una lista de las diferentes opciones que podemos simular.



- **Enlace a:** cambia la pantalla que se esté viendo por la pantalla seleccionada. También se puede hacer que se muestre una URL (dirección de Internet). Además se puede configurar para que la nueva pantalla se vea dentro de un pop up.
- **Acción de Data Master:** son acciones relacionadas con la simulación de listas de datos dinámicas y formularios funcionales. Se explican en detalle en el tutorial de nivel avanzado.
- **Acción de Pagar:** También está relacionado con la simulación de listas de datos dinámicas y queda fuera del alcance de este tutorial.
- **Mostrar / Esconder:** esconde o muestra contenido existente en la pantalla actual.

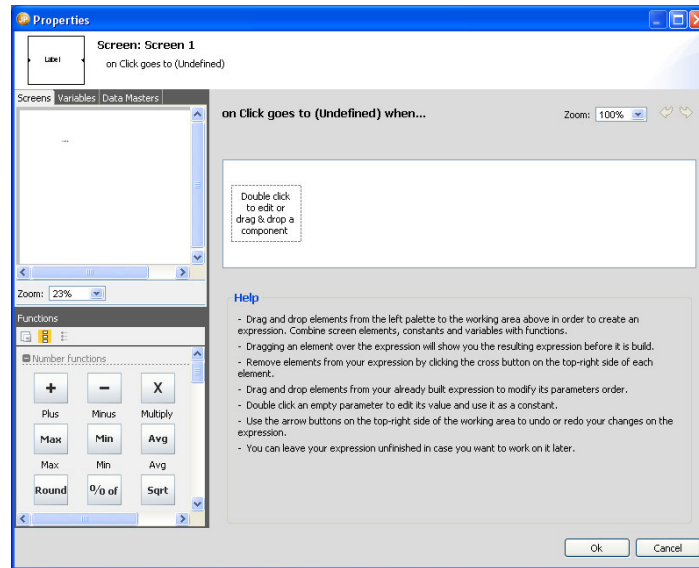
- **Cambiar Estilo:** cambia las propiedades de color, borde, fondo, etc. de un componente de la pantalla actual.
- **Habilitar / Deshabilitar:** convierte un campo de entrada de formulario en editable (si no lo era) o lo convierte en no editable.
- **Poner el foco en:** centra la pantalla en el elemento seleccionado y lo activa si es un campo de entrada de formulario.
- **Poner valor:** cambia el valor de una etiqueta, campo de formulario o variable por el que indique el usuario.
- **Selecciona valor:** sólo es válido sobre campos de entrada de formulario de tipo selección (desplegables, listas, grupos de checkboxes, etc) y selecciona los valores introducidos por el usuario.
- **Pausa:** se espera el tiempo que indique el usuario antes de ejecutar cualquier otra opción. Esta acción se puede combinar con otras para simular animaciones como, por ejemplo, una barra de progreso.

Este cuadro de diálogo también da la opción de añadir una condición pero esa parte se explicará en el siguiente apartado. Una vez definida la acción que queremos realizar pulsaremos el botón de "OK". En ese momento aparecerá la acción seleccionada dentro de la pestaña de Eventos. Una misma acción de usuario puede desencadenar varias acciones en forma secuencial. Por ejemplo podemos añadir otra acción que se ejecute después de la que acabamos de definir pulsando el botón de la flecha al lado de la palabra "haz". También se pueden añadir más casos pulsando otra vez el botón de "Añadir interacción". Los casos se ejecutan de arriba abajo y las acciones de izquierda a derecha. Se volverá sobre este tema en el tutorial de nivel avanzado cuando se hable de cómo simular animaciones.

Condiciones

Justinmind Prototyper permite añadir condiciones a los eventos para que sólo se ejecuten en ciertos momentos de la simulación. Por ejemplo podríamos querer simular un formulario de "log in" que sólo deje navegar a la pantalla siguiente si el usuario y la contraseña introducidos son correctos y si no mostrar un mensaje de error.

Para crear este formulario de "log in" primero lo dibujaremos en una pantalla arrastrando etiquetas, campos de formulario y botones. Una vez dibujado arrastramos el botón de "Log in" a la pantalla de bienvenida. En la pestaña de eventos se habrá creado de forma automática un evento de navegación. Si nos fijamos veremos que encima de la navegación hay la opción de "Añadir condición". Si pulsamos sobre esta opción se abre un cuadro de diálogo especial para construir condiciones.



Este editor de condiciones está compuesto, por un lado, por un área de edición marcada en color blanco en la que construir la condición y por otro por los elementos que se pueden usar para construirla. Los elementos se dividen en componentes y en funciones. Los componentes pueden ir desde elementos de formulario o textos de las pantallas a variables, data masters o campos de data master. Todos estos componentes se pueden arrastrar y soltar en los huecos del área de edición. Las funciones son las operaciones que se pueden realizar sobre esos componentes para construir la condición.

Así pues, siguiendo con el ejemplo del log in, construiremos una condición que compruebe que el campo “user” es igual a “admin” y el campo “password” es igual a “1234”. Primero arrastramos la función “&” hacia el área de edición. En el primer hueco de esta función arrastramos otra función de “=” y hacemos lo mismo con el otro hueco de la función “&”. Arrastramos el campo “user” de la pantalla al primer hueco de la primera función “=” (nota: si no se ven bien los campos de la pantalla se puede usar la opción de zoom). En el segundo hueco de la primera función escribimos la palabra “admin”. Arrastramos el campo password al primer hueco de la segunda función y en el último hueco escribimos “1234”. Si pulsamos el botón de “ok” veremos la condición que acabamos de construir en la pestaña de eventos.

Tras crear la condición podemos ver que debajo del evento de navegar ha aparecido una opción “si no” para indicar qué pasa si no se cumple esa condición. A continuación vamos a simular que si no se cumple la condición se va a mostrar un mensaje de error en el formulario de log in. Primero tenemos que crear el mensaje de error usando el componente de “Etiqueta”. Arrastramos ese componente a la pantalla y escribimos el mensaje de error que queramos. Después marcamos ese componente de texto como “oculto” usando el panel de propiedades. Seguidamente seleccionamos el botón de log in y pulsamos sobre la opción de “si no” en la pestaña de eventos. Seleccionamos la opción de “Mostrar/Ocultar”, seleccionamos el texto de error y escogemos la opción de “Mostrar”. Para finalizar pulsamos el botón de “ok” y pulsamos el botón de simular para comprobar que hemos construido bien la simulación del formulario de log in.

Se pueden hacer las condiciones todo lo complicado que se quieran y se pueden ir encadenando casos de “si ‘x’ haz ‘y’ si no” también que se deseen. Las condiciones también se pueden copiar y pegar en otros eventos incluso en eventos de otros componentes.

Variables

El sistema de variables permite guardar datos durante la simulación de un prototipo para usarlos en otras partes del prototipo. Parte de la idea de “variables globales” y vienen a ser como unos contenedores en los que se puede almacenar un dato en una pantalla y usarlo en otra.

Por ejemplo podríamos crear un formulario de registro que nos lleve a una pantalla de bienvenida y, en esta pantalla, mostrar el nombre de usuario que se ha introducido en el campo “usuario”. Primero dibujaremos el formulario de registro usando los componentes de la paleta. Después crearemos una variable desde la pestaña de “Variables” situada debajo de la paleta de componentes. Arrastraremos el campo de entrada donde se introduce el nombre del usuario a la variable indicando así que al salir de esa pantalla se guardará en la variable lo que el usuario haya escrito en ese campo. Seguidamente abriremos otra pantalla e insertaremos dos etiquetas de texto. En una escribiremos “Bienvenido” y arrastraremos la variable encima de la otra. De esta manera indicamos que al cargar la pantalla se sustituya el texto de esa etiqueta por el valor que tenga la variable en ese momento. Si pulsamos el botón de simular desde la primera pantalla veremos como se pasa el dato de una pantalla a otra.

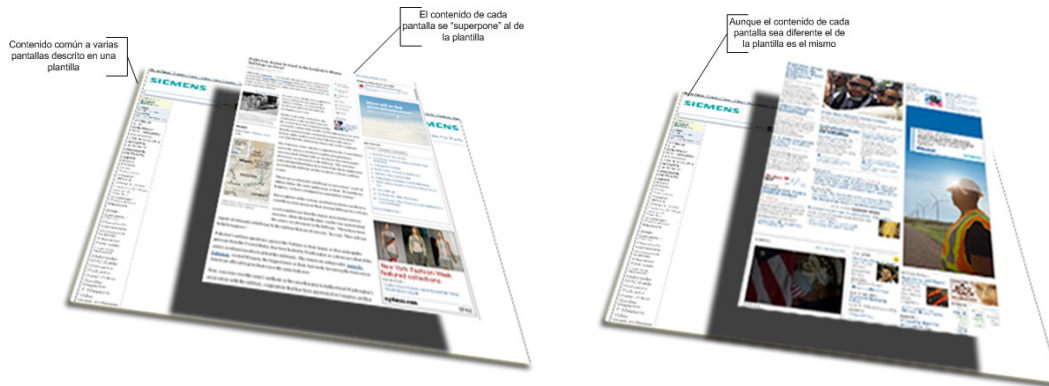
Al arrastrar el campo de “usuario” y soltarlo encima de la variable el programa ha creado de forma automática un evento. Si nos fijamos más en detalle en ese evento veremos que es de tipo “Salir de pantalla” y que lo que hace es cambiar el valor de la variable por el contenido de ese campo. Ésta no es la única manera de inicializar una variable, cualquier acción de “Poner Valor” asociada a un evento puede cambiar el valor de una variable. Para ello, en el diálogo de eventos hay que seleccionar la pestaña “Variables” situada al lado de la pestaña “Pantalla” y seleccionar la variable a la que queremos cambiar el valor.

Las variables también se pueden usar como un elemento más en las condiciones. Si abrimos el diálogo de expresiones veremos que, además de poder arrastrar elementos de la pantalla, también podemos arrastrar variables como literales de las condiciones.

Plantillas

El sistema de plantillas es una herramienta muy útil para hacer prototipos rápidamente y, sobre todo, para hacer cambios de forma global. Por un lado permiten definir el contenido y las interacciones comunes a varias pantallas en un solo sitio y, por otro, definir estilos (colores, tipos de borde, tipos de letra, etc.) por defecto de los elementos de pantalla en las pantallas que usen esa plantilla.

La idea que subyace en el concepto de “plantilla” es la de una pantalla que se pone como “fondo común” entre diferentes pantallas, de forma que las pantallas se superponen al contenido de la plantilla que usan. El siguiente dibujo ilustra lo que acabamos de explicar:




El dibujo representa una web en la que se observa que la cabecera y la barra de navegación lateral son comunes a todas las pantallas, en cambio el contenido central de cada pantalla varía. Si quisiéramos prototipar esta web usando Justinmind Prototyper, lo mejor sería colocar la cabecera y la barra de navegación en una plantilla y dibujar el contenido central en las diferentes pantallas del prototipo.

La pestaña “Plantillas” está situada debajo de la paleta de componentes de pantalla y agrupa todas las plantillas disponibles en un prototipo. Por defecto esta agrupación está compuesta de una sola plantilla. Si queremos ver su contenido debemos hacer doble-clic sobre el nombre de dicha plantilla. Se abrirá entonces el contenido en el área de trabajo igual que si fuera una pantalla normal del prototipo.

El contenido de una plantilla se define de la misma manera que el de una pantalla, arrastrando componentes de pantalla al lienzo. Por ejemplo podemos arrastrar una imagen en la parte superior izquierda de la plantilla con un logo. Si abrimos las pantallas vemos como ese logo aparece en la misma posición en todas ellas. Si volvemos a la plantilla y hacemos un link en ese logo también podremos observar que ese link funciona en las pantallas, de forma que no sólo se comparte el contenido sino también los eventos.

Un prototipo puede tener varias plantillas que se crean pulsando el botón de “+” en la pestaña de las plantillas. Para indicar que una pantalla usa una plantilla en concreto hay que pulsar en la pantalla con el botón derecho y seleccionar la opción de “Propiedades”. Seguidamente seleccionar la opción de “Plantilla” y seleccionar la plantilla que queremos que use la pantalla.

Cada plantilla define un estilo por defecto de cada tipo de componente de pantalla. Así una etiqueta que se arrastra en un par plantilla-pantalla puede tener un aspecto por defecto diferente de otro. Estos estilos por defecto se indican pulsando con el botón derecho sobre la plantilla y seleccionando la opción “Estilos por defecto”. También se puede indicar que el estilo por defecto sea el de un elemento que ya está en una pantalla. Para ello basta con seleccionar dicho elemento y pulsar la opción de “Asignar

como estilo por defecto” representada por el icono  situado dentro de la pestaña de Propiedades del elemento. Al pulsarlo las propiedades de ese elemento quedan marcadas como las propiedades de estilo que tendrán a partir de ese momento todos los componentes del mismo tipo que se arrastren a pantallas que usen esa plantilla. Justo encima de este botón hay otro botón que hace el efecto inverso, es decir, cambiar las propiedades de estilo del elemento actual por las que se han marcado como por defecto en la plantilla.

Las plantillas, como tantos otros componentes dentro de Justinmind Prototyper, se pueden exportar e importar en otros prototipos para ir cada vez más rápido haciendo un nuevo prototipo.

Truco: ¿Cómo hacer unas pestañas que vayan cambiando de pantalla pero que además dejen remarcada la pestaña que hace referencia a la pantalla actual? Muy fácil, primero se dibujan todas las pestañas como si estuvieran ‘desactivadas’ en la plantilla y se hacen los links correspondientes. Seguidamente, en cada pantalla, se dibuja encima de la pestaña de esa pantalla una pestaña con estilo ‘seleccionada’. Sólo queda comprobar que efectivamente el efecto es el deseado usando el botón de ‘Simular’.

Masters

Los masters son agrupaciones de elementos de pantalla y se pueden usar en todo el prototipo. Sirven, además de para agrupar elementos, para propagar cambios de forma global. De esta forma, si cambiamos el contenido de un master cambiará en todas aquellas pantallas del prototipo en las que se haya incluido.

Para crear un nuevo master hay que pulsar el botón de “+” situado en la pestaña de Masters, justo debajo de la lista de componentes y al lado de la pestaña de Plantillas. Una vez pulsado el botón y darle un nombre al master se abrirá el contenido en el lienzo.

En el master se puede arrastrar cualquiera de los componentes disponibles en la paleta como si fuera una pantalla. También se pueden definir eventos siempre y cuando actúen solamente sobre componentes del master. Una vez acabado, el master está listo para usarse en las diferentes pantallas. Para incluir un master en una pantalla basta con arrastrarlo de la lista de masters a la zona de la pantalla que queramos. Una vez colocado veremos que el tamaño del master se ajusta a su contenido.

Los masters se pueden arrastrar a todas las pantallas que queramos incluso más de una vez en la misma pantalla. Una vez colocados los masters, si queremos hacer un cambio en alguno de los elementos que contiene deberemos abrir el master haciendo doble click en la lista de masters o haciendo doble clic sobre el master que está en la pantalla. Se nos abrirá en el lienzo el contenido del master y podremos realizar los cambios. Una vez hechos los cambios observaremos como se han propagado a todas las pantallas en las que habíamos arrastrado ese master. De esta forma se pueden realizar cambios globales en partes comunes del prototipo de manera centralizada. También hay la opción de hacer que un master deje de propagar los cambios para una

pantalla en concreto. Para ello deberemos seleccionar el master en esa pantalla y abrir el menú contextual. Seguidamente seleccionar la opción de “Romper master” de forma que deje de ser un master y pase a ser una copia independiente del contenido del master. Igual que en el caso de las plantillas, los masters también se pueden exportar e importar para compartirlos con otros usuarios o para usarlos en otros prototipos.

NOTA: Cuando se elimina un master de la lista de master se eliminan todas las ocurrencias de ese master en las pantallas del prototipo. Eliminar una ocurrencia de un master en una pantalla no afecta al resto de ocurrencias del master en el prototipo.

Una vez explicados los Masters y las Plantillas puede surgir la duda de en qué casos usar unos u otros. Las plantillas se suelen usar para definir todo aquel contenido que es común a varias pantallas y que siempre aparece en la misma posición y nunca aparece por encima de un elemento de la pantalla. En las plantillas se suelen poner las cabeceras, los pies de página y la navegación principal. Los masters se suelen usar para definir el contenido común a varias pantallas pero que aparece en diferentes posiciones según la pantalla. También se suelen usar para definir contenido común que en cierto momento aparece por encima del contenido de la pantalla, como por ejemplo un menú de navegación desplegable. Se suelen usar los masters para hacer cajas de links, buscadores o cajas de registro.

Librerías de componentes

Las librerías de componentes permiten guardar agrupaciones de elementos de pantalla para poder usarlos en otras pantallas u otros prototipos. Se parecen a los masters pero en este caso un cambio en un componente de la librería no afecta a los componentes que se han arrastrado en la pantalla. Son ideales para plasmar guías de estilo o para prototipar un cierto tipo de aplicaciones. Por ejemplo, en la sección de extras hay un apartado con librerías para prototipar los diferentes tipos de interfaces de dispositivos móviles.

En la pestaña de “Mis widgets” situada en la paleta de componentes salen listados todos los componentes de librerías de nuestro Justinmind Prototyper. Para importar una librería ya existente basta con pulsar el botón de importar, situado al lado del botón de ‘+’ dentro de la pestaña de “Mis Widgets” y seleccionar el archivo .jpl que contenga los elementos de la librería. Los elementos de esa librería aparecerán en la lista de “Mis widgets” y se pueden arrastrar a las pantallas del prototipo. Una vez importada una librería estará disponible para todos los prototipos que se hagan con ese Justinmind Prototyper.

Patrones de diseño

A continuación explicaremos ejemplos de patrones de diseño que se pueden simular con Justinmind Prototyper.

Imágenes interactivas

En este ejemplo usaremos el widget llamado “mapa de imágenes” para definir eventos sobre zonas de una imagen.

Primero incluimos una imagen, en este caso el logo de Justinmind, y después arrastramos el widget de “mapa de imágenes” encima de la letra “d” de la palabra “Justinmind” en la imagen que hemos incluido. Definimos el tamaño del mapa de imágenes para que sólo ocupe esa letra. Después añadimos un componente de texto rico al lado de la imagen del logo. Seleccionamos el mapa de imágenes y creamos un evento que al clicar con el ratón se esconda el texto que está al lado de la imagen. Pulsamos el botón de simular y comprobamos que, efectivamente, si se pulsa sobre la letra “d” el texto situado a su derecha desaparece. De esta forma tan sencilla se pueden animar zonas de imágenes y hacerlas interactivas.

Cambiar el valor de un texto dinámicamente

En este caso vamos a simular un texto que si lo pulsamos nos permite editarlo directamente y ver que efectivamente el valor del texto se sustituye por el introducido por el usuario.

Primero añadimos un componente de “paneles dinámicos” y un texto dentro de este panel (nota: cuando se arrastra un componente de pantalla encima de un “panel dinámico” el componente queda insertado dentro de la capa que se esté viendo en ese momento). Después seleccionamos el panel dinámico y creamos una nueva capa. En esta nueva capa insertamos un campo de entrada de texto y un botón. Ya tenemos todos los elementos en la pantalla, ahora faltan las interacciones.

Seleccionamos el botón de la capa dos y pulsamos la opción de “añadir interacción” dentro de la pestaña de eventos. Seleccionamos la acción de mostrar/esconder e indicamos que queremos mostrar la capa 1 y pulsamos “ok”. En la pestaña de eventos aparecerá el evento que acabamos de crear. Seguidamente pulsamos la flecha situada al lado de la palabra “hacer” en la pestaña de eventos para que salga el menú contextual. Seleccionamos la opción de “crear acción” que nos permitirá concatenar otro comportamiento al que ya tenemos definido. Saldrá entonces el diálogo de escoger acción. Seleccionamos la acción de “cambiar valor”, seleccionamos la capa uno del panel dinámico y dentro de esa capa seleccionamos el elemento de texto. Después seleccionamos el valor pulsando la opción de “Calculado”. Se mostrará entonces el diálogo para construir expresiones calculadas. En la pestaña “Pantalla” seleccionamos la capa dos del panel dinámico y arrastramos el elemento de entrada de texto a la parte superior en el cuadro punteado. Pulsamos “ok” y volvemos al diálogo de eventos. Volvemos a pulsar “ok”. Acabamos de definir que cuando el usuario pulsa el botón, lo que haya escrito en el elemento de entrada de texto se va a escribir en el texto de la capa uno. Finalmente seleccionamos la capa uno y seleccionamos el elemento de texto que hemos insertado en ella. Vamos a la pestaña de eventos y añadimos una interacción que al clicar sobre el texto nos muestre la capa dos. Si pulsamos el botón de simular podremos comprobar que, efectivamente, si el usuario pulsa sobre el texto

aparece un elemento de entrada con un botón y que, al pulsar el botón, se vuelve a ver el texto pero con lo que había introducido el usuario.

Tooltip

Para simular que un cierto elemento ha de mostrar un tooltip es tan sencillo como seleccionar el elemento e indicar el texto que ha de salir en ese tooltip en la pestaña de propiedades. Si no hay ningún texto definido en esa propiedad, no saldrá ningún tooltip.

Tooltips personalizados

En este caso usaremos el evento de “pasar el cursor sobre” para simular un tooltip personalizado. Lo primero es dibujar el contenido de ese tooltip, en este caso lo podríamos hacer agrupando un rectángulo y un componente de texto rico. Seleccionamos el grupo y en las propiedades lo marcamos como “oculto” de forma que cuando se pulse el botón de simular el grupo no se verá. Después arrastramos cualquier elemento, por ejemplo una imagen. Seleccionamos la imagen y pulsamos la opción de “al clicar” situada en la pestaña de eventos. Se desplegará entonces una lista con las interacciones que se pueden capturar del usuario y seleccionamos la opción de “pasar cursor sobre”. Seguidamente añadimos una nueva interacción que muestre el grupo y pulsamos el botón de simular para comprobar que mientras el cursor del ratón está encima de la imagen se muestra el grupo formado por el rectángulo y el texto rico.

Incluir un video de youtube y un mapa de Google Maps

En este ejemplo usaremos el widget “HTML” dentro del grupo de widgets llamado “web” para incluir un video de youtube y un mapa de Google Maps en un prototipo. Primero arrastramos el widget a la pantalla. Después abrimos la página de Youtube en nuestro navegador y seleccionamos el video que queramos insertar. Pulsamos el botón de “insertar” situado debajo del video y copiamos el trozo de código que sale. Seguidamente volvemos a Justinmind Prototyper, seleccionamos el componente HTML y en pegamos el código de youtube en la pestaña de propiedades. Pulsamos el botón de grabar dentro de esa pestaña y veremos como en la pantalla se carga una previsualización del video. Redimensionamos el componente de HTML hasta que se vea todo el video y listo.

Se puede hacer lo mismo con Google Maps, indicando una dirección en la web de Google maps, pulsando la opción de “enlazar” y copiando el código dentro del componente de HTML tal y como hemos hecho con el video de youtube.

Barras de scroll

Las barras de scroll son una propiedad de las capas de un panel dinámico. Para hacer un área con scroll basta con arrastrar un componente de panel dinámico e indicar en las propiedades si va a tener scroll horizontal y vertical. Después sólo hay que introducir el contenido que queramos dentro de esa capa y las barras de scroll se actualizarán de forma automática.

Desplegables condicionales

En este ejemplo veremos como el contenido de un desplegable cambia en función de lo que haya seleccionado el usuario en otro desplegable.

Primero arrastraremos dos elementos desplegables a la pantalla. Seleccionamos uno de los dos y pulsamos la pestaña de propiedades. Después pulsamos el icono del lápiz situado al lado del desplegable "Valor" y aparecerá una pantalla que nos permite editar los valores de ese desplegable. Cambiamos los valores por defecto haciendo doble clic sobre cada uno de ellos y ponemos nombres de países. Pulsamos "ok" y seleccionamos la pestaña de eventos. Pulsamos sobre el icono de "clic" de ratón y, en la lista de eventos posibles, seleccionamos el evento de "Cambiar valor". Después pulsamos la opción de "Crear interacción" y seleccionamos la de "Poner Valor" en la lista de acciones de la izquierda. Seleccionamos el segundo desplegable y añadimos valores de ciudades de uno de los países (por ejemplo, si uno de los países añadidos es USA se pueden añadir unas cuantas ciudades como New York, Washintong DC and Chicago. Seguidamente pulsamos sobre la opción de añadir condición para definir que solo salgan esas ciudades en el desplegable si se ha seleccionado "USA" en el otro. Arrastramos el primer desplegable (el de países) al recuadro punteado que dice "Doble clic para editar...". Después arrastramos encima la función "=" situada en la pestaña de funciones "General". Por último hacemos doble clic en el nuevo recuadro punteado que ha aparecido y escribimos USA. Pulsamos "Ok" y en la pantalla de eventos volvemos a pulsar "ok". En la pestaña de eventos nos aparecerá entonces la descripción de la interacción que acabamos de definir. Si pulsamos sobre el texto que dice "si no..." podremos definir el resto de casos para el resto de países igual que hemos hecho el primero. Finalmente si pulsamos el botón de simular veremos como cada vez que cambia el valor del desplegable países se cargan unas ciudades diferentes en el desplegable "ciudades".

Saltar entre campos de entrada usando el tabulador

En formularios con varios campos de entrada es muy usual que el usuario salte de campo en campo usando la tecla tabulador. Este comportamiento se simula de forma automática en el HTML exportado desde Justinmind Prototyper y el orden lo define la profundidad de los elementos en la pantalla. A continuación veremos como definir el orden del tabulador de una forma más específica.

Primero arrastramos varios campos de entrada de texto a la pantalla. Después seleccionamos el primero de ellos y pulsamos en la pestaña de eventos. Pulsamos donde pone "clic" con el icono del ratón y seleccionamos la opción de "pulsar tecla". Seguidamente pulsamos la opción de "Crear interacción" y se muestra la pantalla de definir acciones. A diferencia de otros casos, al lado del botón de "Añadir condición" aparece un elemento que permite detallar exactamente qué tecla va a provocar el evento. Por defecto cualquier tecla que se pulse dentro del campo de entrada provocará la acción que definamos pero lo cambiaremos para que sólo tenga en cuenta la tecla tabulador. Pulsamos el icono del lápiz y seguidamente la tecla de tabulador. Después seleccionamos la acción de "Poner foco en" y seleccionamos otro campo de entrada de texto. Pulsamos "ok" y pulsamos el botón de simular. Si

escribimos algo en el primer campo de entrada y pulsamos la tecla tabulador observaremos como el cursor salta al siguiente campo.

Nivel avanzado

Esta sección profundiza en las diferentes opciones de simulación de Justinmind Prototyper y se enfoca en las que tienen que ver con simulación de datos, listas dinámicas, formularios y simulación con datos reales.

Listas dinámicas de datos

En este capítulo aprenderemos como simular contenido estructurado ya sea en forma de tablas o en listas y poblar ese contenido de forma dinámica. Veremos como funcionan los data masters, el componente de data grid, como simular búsquedas y como añadir paginación a esas listas de datos. También veremos como preparar una colección de datos de prueba.

Empezamos creando un data master pulsando el botón de '+' situado en la pestaña de Data Masters en la parte inferior izquierda de la pantalla. El programa nos muestra un diálogo para introducir los datos del Data Master. En este ejemplo simularemos una lista de productos así que el Data Master lo llamaremos Producto. En la futura lista de productos queremos que aparezca, para cada producto, un código identificador, un nombre, una descripción y una foto. Añadimos cada uno de estos campos usando el botón de "+" situado justo debajo de donde hemos puesto el nombre del data master e indicamos el nombre de cada uno en la tabla situada a su derecha. Una vez añadidos los atributos podemos observar como, además de una columna nombre, en la tabla también hay una columna "tipo" y que todos los atributos son de tipo "texto". Seleccionamos el tipo "texto" de la fila que ocupa el atributo "foto" y observamos como sale un desplegable con varios tipos disponibles. Seleccionamos el tipo "fichero" porque, como veremos, el tipo afecta a como se representa esa información en listas de datos y en formularios. Por último pulsamos el botón de "ok" no sin antes seleccionar el id como único atributo del descriptor (el descriptor se usa cuando asociamos un data master como fuente de datos de un elemento desplegable en una pantalla).

En la pestaña de Data Master podemos ver como aparece listado el que acabamos de crear. A continuación arrastramos el elemento Tabla de Datos de la lista de componentes de pantalla a la pantalla que tenemos abierta en ese momento. Aparecerá una pantalla para indicar información sobre esa tabla de datos, en concreto deberemos indicar un identificador, seleccionar el data master que se va a usar como fuente de datos (en nuestro caso elegimos el único que hay, el de Producto) y qué atributos del data master se van a usar como columnas en esa lista, en nuestro caso seleccionamos todos los atributos. Pulsamos aceptar y el programa nos enseña cuanto va a ocupar la tabla, pulsamos sobre algún lugar de la pantalla y observamos como se dibuja el data grid con una fila ficticia. Si pulsamos el botón de simular veremos como la tabla se rellena con tres filas de ejemplo.

A continuación cambiaremos esas filas de ejemplo con datos más representativos. Hacemos doble clic en el data master producto dentro de la pestaña de Data Masters y volvemos a ver la misma pantalla que cuando estábamos creándolo. Sin embargo en este caso aparece una nueva pestaña en la parte superior que dice "Ver y editar instancias". La pulsamos y vemos una tabla en la que cada columna corresponde con un atributo del data master, aquí podemos definir un juego de datos de prueba que será el que muestre todas las tablas de datos que usen este data master. Podemos indicar en este caso algunos productos y seleccionar algunas imágenes para el campo foto. Pulsamos ok y pulsamos el botón de simular para ver que efectivamente en lugar de las tres filas de ejemplo que salían antes ahora salen los datos que hemos introducido en el data master y la foto de cada producto.

Las tablas de datos se pueden personalizar hasta formar listas de datos estructuradas. Siguiendo con el ejemplo podríamos mover la información de las celdas de la tabla de productos correspondientes a código, nombre y descripción dentro de la que muestra la foto del producto, hacer esa celda más grande para que quepa toda esa información y borrar el resto de columnas. También podemos seleccionar el listado entero y definir el color de las filas pares/impares.

Justinmind Prototyper contiene todo un apartado de funciones para realizar cálculos sobre data masters y tablas de datos. Usaremos por primera vez una de esas funciones para simular una caja de búsqueda sobre el listado de productos. Primero dibujamos la caja de búsqueda añadiendo un campo de entrada de texto y un botón. Después seleccionamos el botón y pulsamos la pestaña de eventos. Pulsamos el botón de "Crear interacción" y seleccionamos la acción de "Poner valor". Seleccionamos la tabla de datos (importante, hay que seleccionar la tabla entera, no una celda o una fila, lo sabremos porque el recuadro de selección agrupa la cabecera y la fila) y pulsamos sobre el botón de "Calculado". En el editor de expresiones pulsamos en la pestaña de "listas" y arrastramos la función de "Filter" a la parte superior. En la función hay dos huecos, el primero hace referencia a la fuente de datos que queremos filtrar, en este caso queremos hacer una búsqueda sobre todos los productos así que seleccionamos la pestaña de Data Masters y arrastramos al primer hueco el data master "Producto". El segundo hueco es la regla que seguirá la función de filtrado para decidir si un valor es filtrado o no. En este segundo hueco arrastraremos de la pestaña Pantalla el campo de entrada de texto situado al lado del botón de búsqueda. Pulsamos "ok" y "ok" otra vez y pulsamos el botón de simular. Tendremos nuestra caja de búsqueda perfectamente operativa.

En ocasiones el número de resultados de una tabla de datos es tan grande que se dividen en varias páginas. A continuación veremos como simular ese tipo de paginación usando los componentes de "Índice" y "Resumen" y las acciones de listado.

Primero hay que preparar la tabla de datos para que use las funciones de paginación. Seleccionamos la tabla de datos de productos que estamos usando como ejemplo y pulsamos la pestaña de propiedades. En la propiedad "Filas por página" indicamos el número máximo de filas que se va a mostrar por cada página, si está a 0 significa que no va a limitar el número de filas y por tanto no se usa la paginación. En este ejemplo

pondremos que muestre sólo tres filas por página. Arrastramos un componente de “Índice” a la pantalla y lo relacionamos con la tabla de datos mediante la propiedad “Listado” dentro de su pestaña de propiedades. Automáticamente el índice muestra el número de páginas que va a tener el listado en función del número de filas por página. Este índice es dinámico de forma que si en simulación se hace una búsqueda también se actualiza el índice.

El “Resumen” es otro texto dinámico que como el índice está relacionado con una tabla de datos. Se configura igual que el índice, arrastrando el componente y seleccionando el listado en las propiedades del componente resumen. Este texto muestra el número de resultados que hay en total en todas las páginas del listado y los valores que se están mostrando en ese momento.

Por último vamos a simular las acciones de ver “siguiente página”, “página anterior”, “última página” y “primera página”. Primero arrastramos cuatro botones y les ponemos como textos “siguiente”, “anterior”, “primero” y “último”. Cada uno va a ejecutar una acción de paginación diferente sobre la tabla de datos. Vamos a hacer el ejemplo con la acción de “ver página siguiente” pero se puede aplicar el mismo método para el resto de botones. Seleccionamos el botón de “siguiente” y pulsamos la pestaña de eventos. Pulsamos el botón de crear interacción y seleccionamos la opción de “Acción de paginar”. Pulsamos el botón que indica “página siguiente” y se cargará el editor de expresiones que ya hemos visto anteriormente para calcular condiciones automáticas. Arrastramos el listado que hay en la pantalla en el hueco de la expresión y pulsamos “ok”. Volvemos a pulsar “ok” en el diálogo de eventos y probamos que realmente van cambiando los valores del listado cada vez que se pulsa el botón de “siguiente”. Es conveniente crear unos cuantos valores en el data master para ver mejor como se sincronizan el índice, el resumen y las acciones de paginación mientras se simula.

Para añadir una columna a una tabla de datos hay que seleccionar la tabla, pulsar con botón derecho del ratón y escoger la opción de “Nueva columna”. Aparecerá entonces una nueva columna vacía a la derecha del listado. Para indicar que se vea un cierto atributo del data master en esa nueva columna sólo hay que arrastrar el atributo a la celda. También se puede modificar el orden de las columnas simplemente seleccionándolas desde la parte superior de la cabecera (el cursor cambia por una flecha apuntado hacia abajo) y moviéndolas en horizontal sobre el listado.

Formularios que funcionan