

Informe HCI TPE2

June 19, 2011

1 Decisiones de diseño

Vista de Categorías, Subcategorías y productos

Si bien se trató de ser lo mas fiel posible al diseño original, se hicieron cambios leves.

En cuanto a las listas, se optó por:

- Agregar el precio a cada producto, ya que este dato es lo primero que el usuario mira al momento de tomar una decisión.
- Usar una tipografía mas grande. Cuando se diseñó el menú, este aparentaba ser mas “agradable” cuando cada item utilizaba una única línea, pero al ver los resultados obtenidos en la práctica, se decidió que era un buen cambio. Además, no se había tenido en cuenta a los usuarios con problemas de visión (hay que tener presente que estos son dispositivos con pantallas muy reducidas y que además uno puede encontrarse en ambientes muy iluminados y no verse bien el brillo de la pantalla) ni a los usuarios con dedos grandes, a los que puede dificultarse seleccionar un ítem de la lista debido a su tamaño reducido.
- Se agregó un texto en la parte inferior de la pantalla que muestra por unos segundos la opción seleccionada (para complementar al breadcrumb que puede no ser notado debido a su tamaño).

En cuanto a su implementación, por experiencia con el trabajo práctico anterior, se implemento un "*category manager*", que se encarga de almacenar una lista de categorías y subcategorías que ya fueron pedidas al servidor y de esta manera, se reduce a uno la cantidad de peticiones de cada categoría. Consideramos esto un detalle muy importante a tener en cuenta ya que (por lo general), los dispositivos moviles poseen conecciones a internet mediante 3g (conecciones muy inferiores a las que se tienen en el hogar) y que además es cobradas por paquetes. Por lo que el uso del programa terminaría saliendole muy caro al cliente si va y viene constatemente en la lista.

Se consideró hacer lo mismo para el listado de productos, pero se presentaba

el problema que pueden llegar a existir cientos de productos en el servidor y esto resultar en un programa final mas lento (debido a demasiada información cargada en memoria¹).

En cuanto a la vista de producto, se logró implementar una pantalla muy fiel a la planeada. Un detalle muy importante que hay que resaltar es el hecho que al momento de listar los datos del producto, se usa la misma tipografía para la etiqueta y su valor, esto se vería muchísimo mejor si fuese como en la vista de información de ordenes.

Filtro, paginación y ordenamiento de la información

Al principio del trabajo se pensó en implementar una barra de búsqueda en la parte superior de la pantalla. Sin embargo, al momento de comenzar a programar, se encontró que android ya ofrecía una de estas (y de hecho era casi igual a la pensada originalmente!). Aunque por falta de tiempo, no se pudo llegar a implementar esta funcionalidad ni la de paginación (load on demand).

En cuanto al filtro, éste si fue aplicado. En cualquier momento se puede ingresar texto y la lista se fija si hay algún producto que *contenga* las palabras ingresadas. Como no es para nada intuitivo el ingresar un texto sin que halla ningún lugar para escribir (solo seleccionar) se optó inicialmente por crear un menú *ayuda* en el menú principal en donde se hable sobre esto (y posiblemente sobre otras cosas mas). Pero, por experiencia, se sabe que el usuario no ingresará en este menu a menos que le sea imprescindible. Por lo que se tuvo que pensar otra solución, ya que no se quiere ofrecer funcionalidad que no se use por no ser “visible”.

La solución que actualmente se encuentra aplicada es la de mostrar un texto corto explicando la funcionalidad mencionada en un *Toast* por un corto intervalo de tiempo. Por supuesto, en un lugar poco invasivo en la pantalla.

Menú principal

Para simplicidad de la aplicación y mejor aprovechamiento del espacio en la pantalla, se trató de reducir al mínimo la cantidad de objetos a mostrar. Actualmente (y debido a la cantidad de funcionalidad) solo se tienen tres botones. Uno para ver productos, otro para ver ordenes existentes y un botón para desloguearse.

Por experiencia en trabajos anteriores, se sabe que no es una buena práctica, habilitar botones al usuario cuando en realidad no tiene sentido debido al estado en el que la aplicación se encuentra. Y es por esto que el botón de ver ordenes se desactiva automaticamente cuando el servidor indica que el usuario actual no

¹Se tuvo en cuenta el hecho de usar una lista de tamaño fijo en la que se guarden los últimos elementos consultados y los demás sean descartados, pero se dejó esto para una futura implementación ya que se tenían otras prioridades de funcionalidad mas importantes.

posee ordenes.

2 Conclusiones

Luego de concluido el trabajo práctico, nos dimos cuenta sobre el potencial de *Android*, lo fácil que es diseñar aplicaciones con esta herramienta (sin mencionar el complicado arranque desde cero) y que nos pareció una idea muy innovadora que se halla implementado el uso de este framework en el curso .

No esta de más mencionar que nos hubiese resultado muy interesante poder probar nuestro resultado final en un aparato real ya que después de todo es donde esta destinada a andar.