

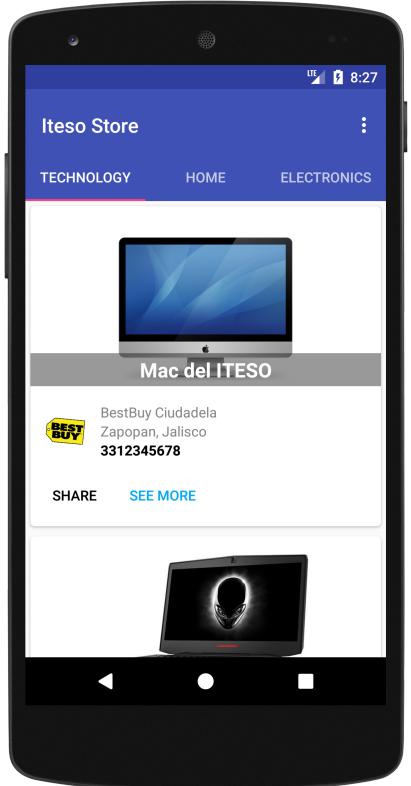
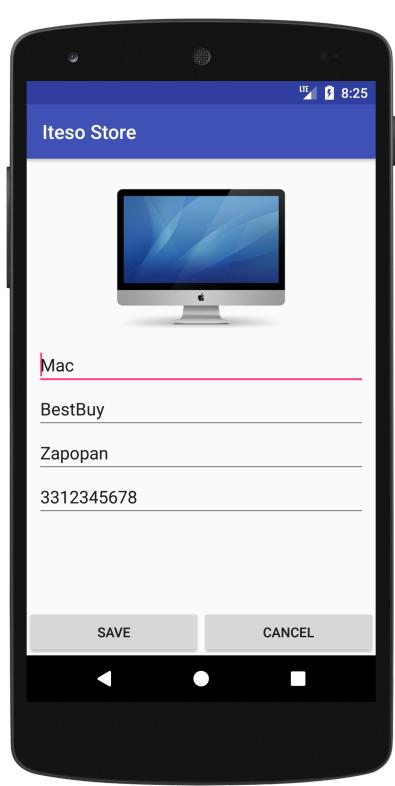
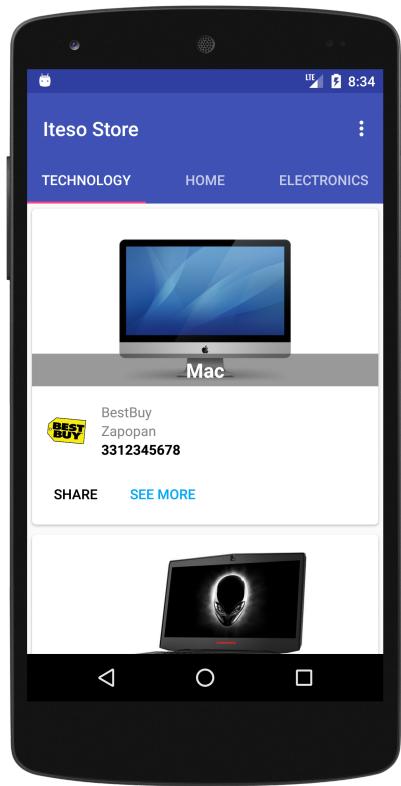
Número de tarea	4
Sesión en la que se dejó	12
Fecha de entrega	2 - Octubre -2018

Temas evaluados:

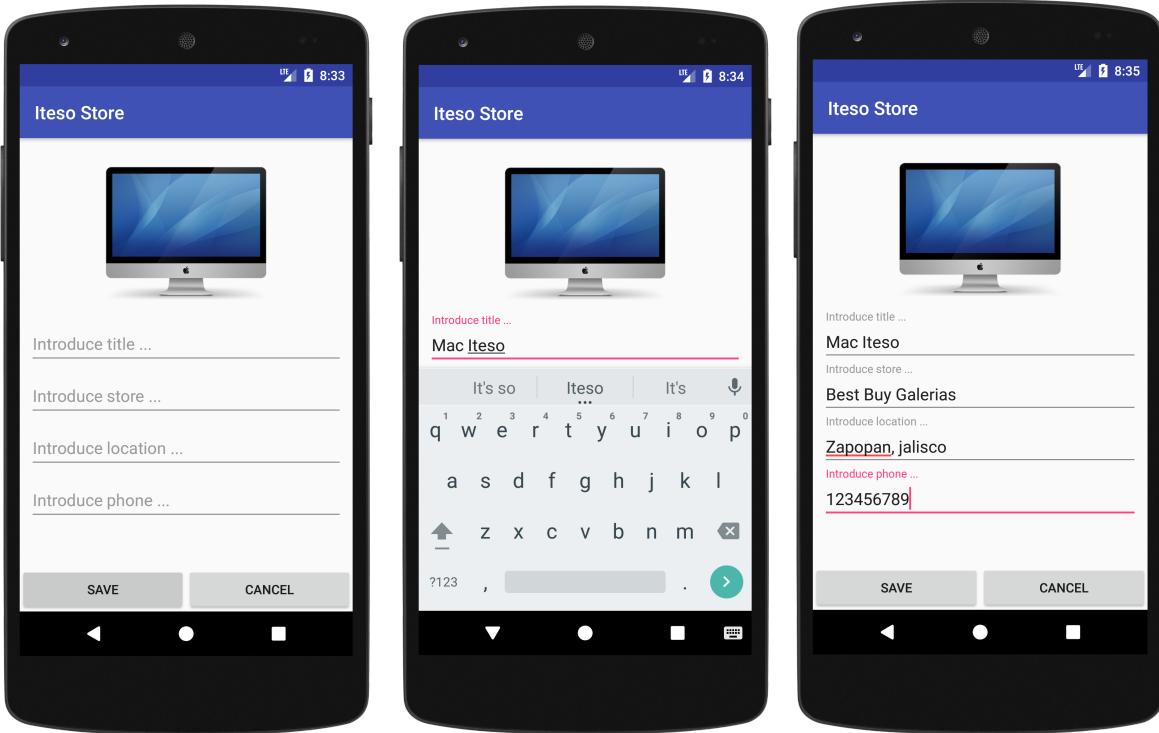
- *Parcelable*
- *startActivityForResult ... onActivityResult*

Descripción:

- 1) Crear una nueva aplicación que tenga como base algunos elementos de la aplicación de la tarea #3.
 - a. Clase `ItemProduct`
 - b. Crear una clase `ActivityMain` en la cual su contenido sea un layout con un solo elemento de `CardView` con un objeto `ItemProduct` desplegado.
- 2) Agregar un atributo al `ItemProduct` entero llamado `code`
- 3) Implementar la interfaz `Parcelable`
- 4) Al dar tap en un Producto deberá abrirse una actividad llamada `ActivityProduct` la cual permita editar el nombre, tienda, ubicación y teléfono. Y deberá presentar en el bottom un par de botones “Guardar” y “Cancelar”
- 5) El resultado de dar tap en “Guardar” deberá retornar el objeto de `ItemProduct` con las nuevas propiedades actualizadas al `ActivityMain` y actualizar el elemento con los nuevos valores.
- 6) El resultado de dar tap en “Cancelar” es únicamente para cerrar la actividad sin cambio al producto.
- 7) Entregables Esperados en Documento PDF.
 - a. Liga de GIT donde se encuentre su código
 - b. Recursos. (res)
 - i. layout
 1. `activity_main`
 2. `activity_product`
 - c. Código Java
 - i. `beans\ItemProduct`
 - ii. `ActivityProduct`
 - iii. `ActivityMain`
 - d. ScreenShots de Aplicación en ejecución
 - i. `ActivityMain`. Mostrando los datos de producto a editar
 - ii. `ActivityProduct` con datos modificados
 - iii. `ActivityMain`. Mostrando producto con el resultado de haber dado tap en Guardar.



5 Puntos Extra a la tarea si hacen que el layout presente el siguiente comportamiento:



1. Los EditText cuentan con un hint.
2. Al dar tap dentro del EditText el hint tiene una animación hacia arriba para que el usuario no olvide de que trata el campo de texto.
3. Se mantienen los hints en todo momento en el top del EditText.