

Trường Đại học Bách Khoa Hà Nội Hanoi University of Science and Technology

Chapter 4. Graphical User Interfaces



Hanoi University of Science and Technology





© **HUST 2012**







Content

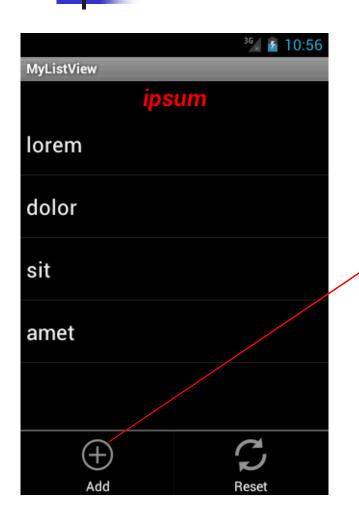
- 4.4a. Menu
 - Option Menus
 - Context Menus
- 4.4b. Showing Pop-Up Messages

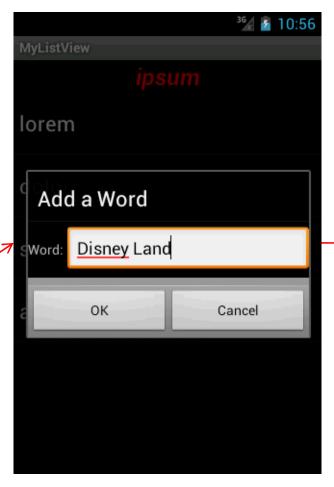
Ví dụ - XML Layout

```
<LinearLayout
   android:orientation="vertical" >
   <TextView
      android:id="@+id/selection"
        />
   <ListView
      android:id="@+id/list"
      />
</LinearLayout>
```



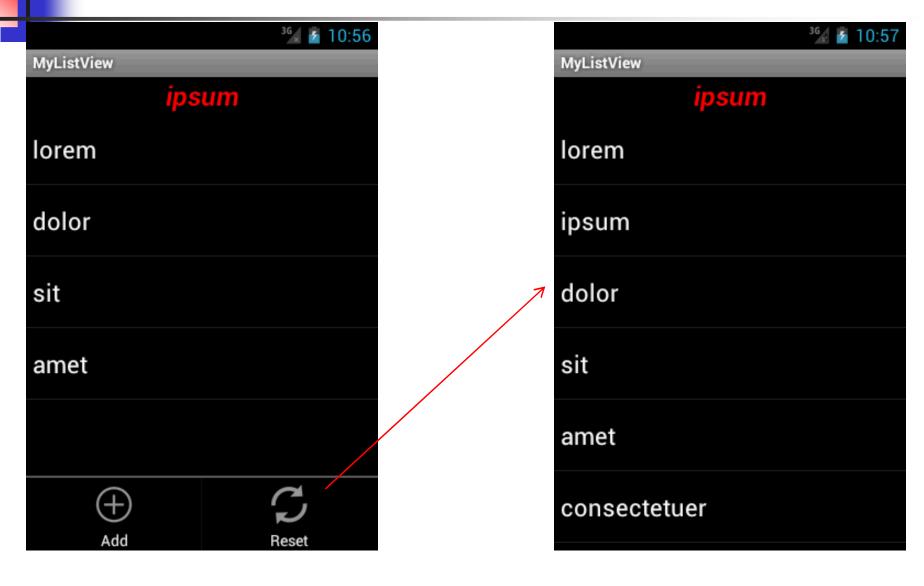
Adding "options menu"



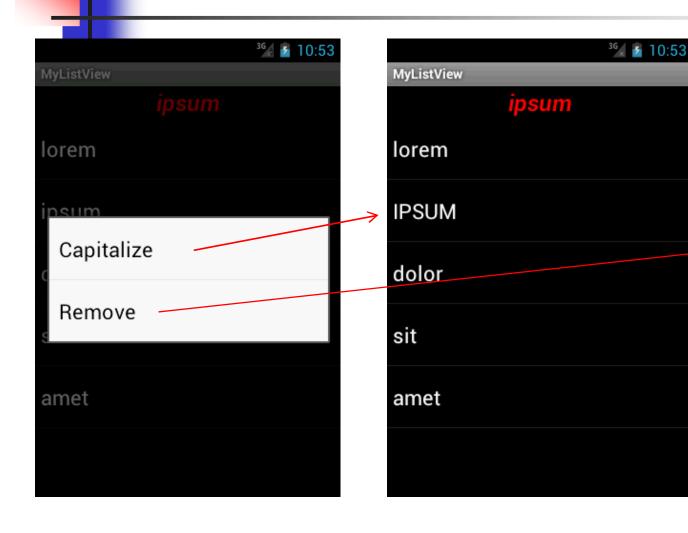




Adding "options menu"



Adding "context menu"







Option menu

- The options menu is triggered by pressing the hardware "Menu button" on the device
- In addition, the options menu operates in one of two modes: icon and expanded.
- When the user first presses the "Menu button, the icon mode will appear, showing up to the first six menu choices as large, finger-friendly buttons in a grid at the bottom of the screen.



Option menu

- If the menu has more than six choices, the sixth button will become "More" – clicking that option will bring up the expanded mode, showing the remaining choices not visible in the regular menu.
- The menu is scrollable, so the user can get to any of the menu choices.

Context menu

- On a traditional GUI, this might be triggered by the right-mouse button. On mobile devices, context menus typically appear when the user "taps-and-holds" over a particular widget.
- For example, if a TextView had a context menu, and the device was designed for finger-based touch input, you could push the TextView with your finger, hold it for a second or two, and a popup menu will appear for the user to choose from.

Option menu

 Display Option menu: public boolean onCreateOptionsMenu(Menu menu);

Process item menu selected events:
 public boolean
 onOptionsItemSelected(MenuItem item);

→ item.getItemId()

Context menu

Display context menu:

```
public void
  onCreateContextMenu(ContextMenu menu,
  View v, ContextMenuInfo menuInfo);
```

Process item menu selected events:

public boolean
 onContextItemSelected(MenuItem item);



Example: ListView with Menus

4

XML Layout: main.xml



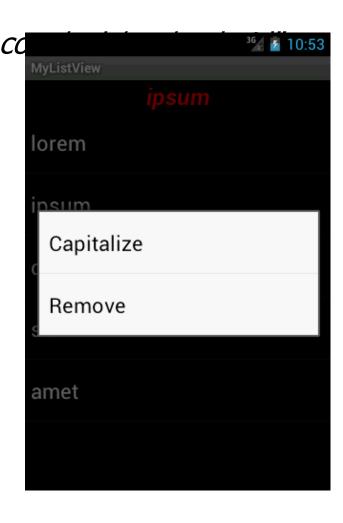
Option menu: Layout option.xml

<menu xmlns:android="http://schemas.android.com/apk/res/android" >



Context menu: Layout context.xml

```
<menu xmlns:android="http://schemas.android.cd"</pre>
  <item
    android:id="@+id/cap"
   android:title= "Capitalize"
   />
  <item
    android:id="@+id/remove"
   android:title="Remove"
</menu>
```



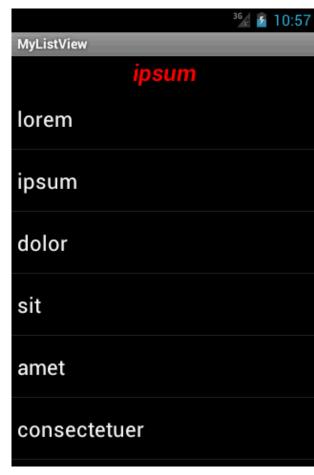
Data model and adapter for ListView

```
String[] dataModel = {"lorem", "ipsum", "dolor",
"sit", "amet", "consectetuer"};
ArrayAdapter<String> adapter;
private ArrayList<String> words=null;
words = new ArrayList<String>();
for (String s : dataModel) {
     words.add(s);
adapter = new ArrayAdapter < String > (this,
     android.R.layout.simple_list_item_1,
     words );
list.setAdapter(adapter);
```

Update TextView

private AdapterView.OnItemClickListener onListClick = new

```
AdapterView.OnItemClickListener() {
@Override
   public void onItemClick(
      AdapterView<?> arg0,
      View arg1, int position, long arg3)
      String word = words.get(position);
      text.setText(word);
```



Display Option menu

@Override

public boolean onCreateOptionsMenu(Menu menu) {

```
MenuInflater inflater = getMenuInflater();
inflater.inflate(R.menu.option, menu);
return true;
```

1

Display Context menu

- onCreate method:
 - registerForContextMenu(list);
- Insert onCreateContextMenu method:

```
@Override
```

onContextItemSelected

@Override

```
public boolean onContextItemSelected(MenuItem item) {
   AdapterView.AdapterContextMenuInfo info=
    (AdapterView.AdapterContextMenuInfo)item.getMenuInfo();
   ArrayAdapter<String> adapter=(ArrayAdapter<String>)list.getAdapter();
    switch (item.getItemId()) {
                                   String word = words.get(info.position);
        case R.id.cap:
                                   word = word.toUpperCase();
                                   adapter.remove(words.get(info.position));
          return(true);
                                   adapter.insert(word, info.position);
        case R.id. remove:
          adapter.remove(words.get(info.position));
          return(true);
    }
    return super.onContextItemSelected(item);
```

onOptionsItemSelected

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
   switch (item.getItemId()) {
       case R.id.add:
        add();
        return(true);
       case R.id. reset:
        initAdapter();
        return(true);
   return super.onOptionsItemSelected(item);
```

4

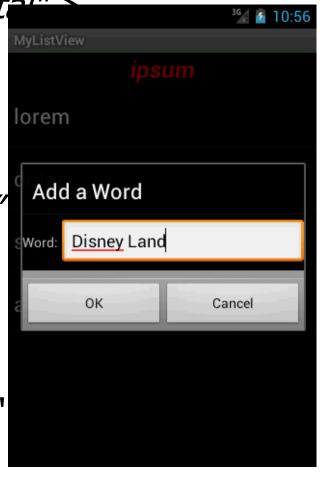
Method initAdapter()

```
private void initAdapter() {
    words = new ArrayList<String>();
     for (String s : dataModel) {
       words.add(s);
     }
     adapter = new ArrayAdapter<String>(this,
        android.R.layout.simple_list_item_1,
        words );
     list.setAdapter(adapter);
```

Method add() Layout "add.xml"

<LinearLayout android:orientation="horizontal"</pre> <TextView android:text= "Word:" android:layout_width="wrap_content" android:layout_height="wrap_content" <EditText android:id="@+id/title" android:layout_width="fill_parent" android:layout_height= "wrap_content"

</LinearLayout>





Method add() AlertDialog

```
AlertDialog.Builder dialog = new
AlertDialog.Builder(this);
dialog.setTitle("Add a Word");
dialog.setView(addView);
dialog.setPositiveButton("OK", ...);
dialog.setNegativeButton("Cancel", null);
dialog.show();
```

Method add() AlertDialog – OK button

```
dialog.setPositiveButton("OK",
    new DialogInterface.OnClickListener() {
        @Override
       public void onClick(DialogInterface dialog, int
         whichButton) {
           ArrayAdapter < String > adapter =
             (ArrayAdapter<String>)list.getAdapter();
           EditText title=(EditText)addView.findViewById(R.id.title);
           adapter.add(title.getText().toString());
```



Trường Đại học Bách Khoa Hà Nội Hanoi University of Science and Technology



