Software Analytics - Bug Report

Talal El Afchal

March 9, 2017

Bug

Material−dialogs is an Android customizable dialogs API. The Git repository has 8758 ⋆ and 1440 commits. The framework provides many types of dialog window, one of them is the *Basic list* dialog which consists of a list of items and one check-box as shown in figure 1
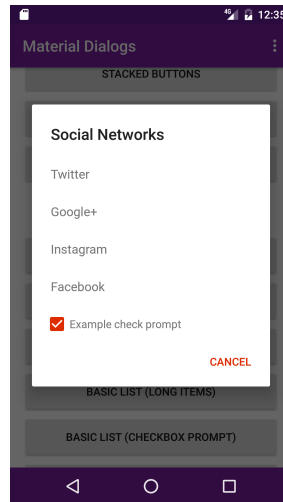

Figure 1: portrait

When the phone is in the landscape mode the check-box disappears as shown in figure 2, but the check-box must be visible like it is in the portrait mode.
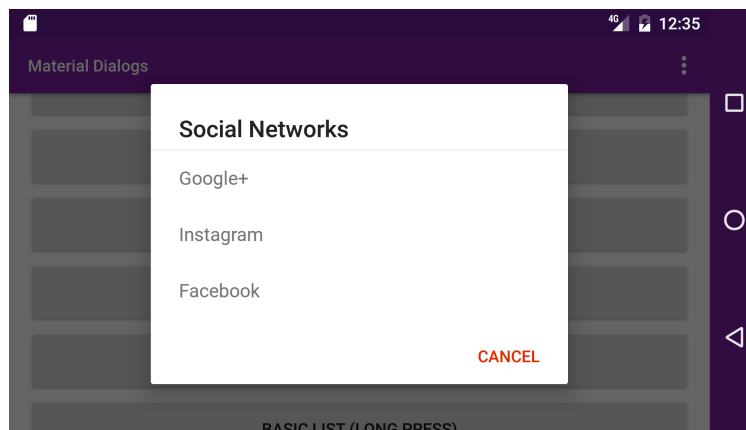On November 4, 2016 a *"bug"* label was added and on Jan 4, 2017 a *"help wanted"* label was also added.

Figure 2: landscape

Bug replication

The check-box in the dialog list disappears when the phone is in landscape mode. Therefore the bug is visible and can by replicated as shown in figure 2.

Bug localization

In order to localize the bug, the first step is to open the project in Android Studio and then from the main activity xml file we can retrieve the id of the *BASIC LIST(CHECKBOX PROMPT)* button, that generates the dialog list in which we are interested, as shown in figure 3.
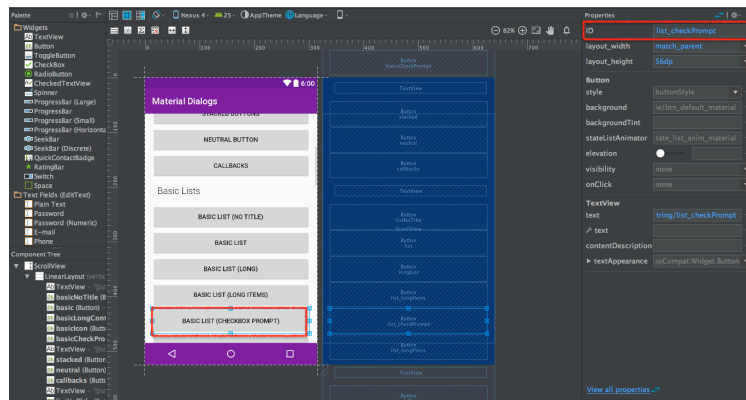

Figure 3: Android Studio

The next step is to use the id to figure out which method is implementing the button's functionality.

I found that the method *showListCheckPrompt()* in *Main Activity* class is called when the button is clicked. This method creates an object of type *MaterialDialog.Builder* (Builder is an inner class), and several methods are invoked on this object as shown in the snippet code. My first guess is that the bug is located in the *show()* method since it shows the check-box in the dialog window.

```
@OnClick(R.id.list_checkPrompt) public void showListCheckPrompt() {
    new MaterialDialog.Builder(this)
            .title(R.string.socialNetworks)
            .items(R.array.socialNetworks)
            .itemsCallback((dialog, view, which, text) -> showToast(which + ":
                " + text))
            .checkBoxPromptRes(R.string.example_prompt, true, null)
            .negativeText(android.R.string.cancel)
            .show();
}
```

Performing a deeper analysis, *show()* is invoked 31 times on 31 different *MaterialDialog* objects and it is invoking the *show()* method from *MaterialDialog* class which is invoking the android API *show()* method. Since the bug is not replicated in these 31 objects and must probably the bug is not in the Android API, I moved my focus to the other methods.

The next step is to analyze the *checkBoxPromptRes()* method. This method sets the attributes of the checkBox variable of a MaterialDialog object. Performing a deeper analysis the *DialogInit* class maps this variable to the xml file. This path leads us to *md_dialog_list_check.xml* file where I spot the bug.

By removing the items list, the check-box is visible in the landscape mode, as we can see in figure 4 . My conclusion is that the *ScrollView* is hiding the check-box and this why we can't see both of them in the landscape mode.
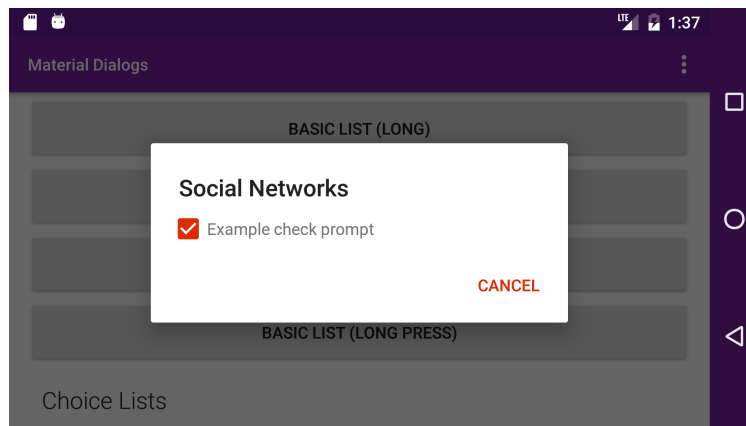

Figure 4: Check-box

Changes impact

The *md_dialog_list_check.xml* file creates the *BASIC LIST (CHECK BOX PROMPT)*'s window interface only. If we make changes in this file, the changes will not affect other dialog windows or other files.

Testing

I tried to add more items to the list to check if the bug also occurs in the portrait mode, and as we can see in figure 5 the check-box is not visible. The problem is not only in landscape mode but indeed the bug is visible in both since the *ScrollView* is taking all the space in the dialog window, and therefor there is no space for the check-box. To test the UI I create an *Espresso* test case to check the check-box visibility, and as expected the application failed to pass this test case.
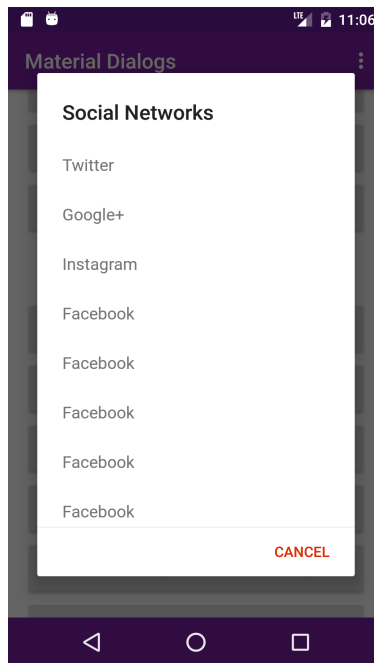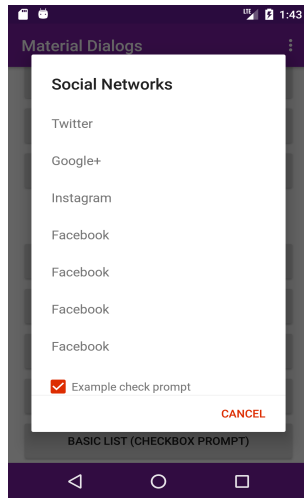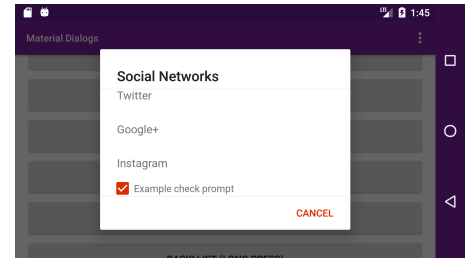

Figure 5: Long list debugging

Fixing & testing

Fixing the bug was straightforward, it was enough to give the check-box layout a higher weight than the scroll-view's weight. And after this modification the application passed the test case. As we can see in figure 6 the check-box and the scroll-view are visible now in both modes.

(a) Portrait

(b) Landscape

Figure 6: Fixed bug