LunchBunch Heuristic Evaluation

Testing disclosure:

Most testing was conducted on a HP Touchpad that is running Android OS 4.0 (Ice Cream Sandwich). This is a tablet device, not a phone, and so this may not accurately reflect the intended platform. While I do have an android phone, it is running on Android 2.3.3, and therefore would not run the application. In order to accurately test the application on a display of the intended size, I installed the application on a PC based Android emulator.
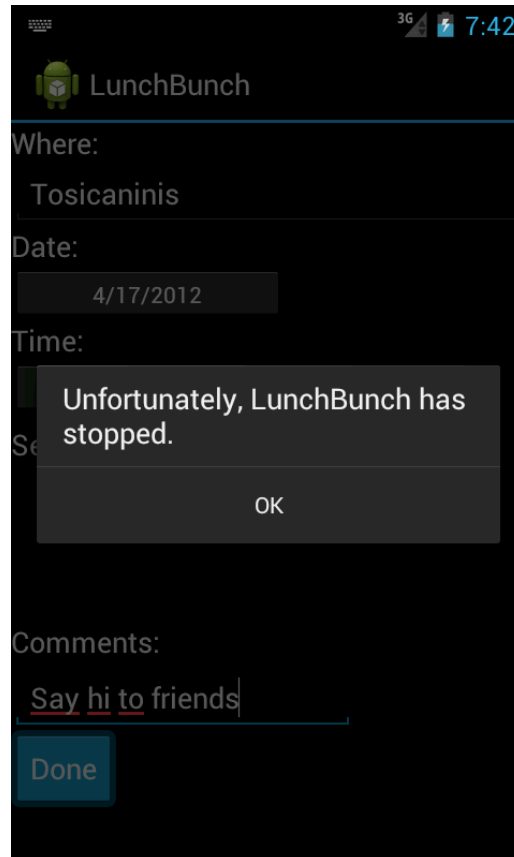
About this evaluation:

The following comments are organized in decreasing order of severity. When appropriate I have provided recommendations.

Problem description:

If I go to create a new event, fill out the form and then click done, the application crashes. The following is a partial error trace:

```
Caused by: java.lang.ClassCastException: android.widget.Button
cannot be cast to android.widget.EditText
source.code.CreateNewLunch.onDoneClicked(CreateNewLunch.java:117)
```


Usability heuristic: Error prevention (S)

Severity: Catastrophic

Problem description:

There is no way to recover from accidently declining an event. For example, if I decline the Taco Bell invite, due to a schedule conflict, but then later the schedule conflict goes away, there is no way for me to get the Taco Bell invite back.

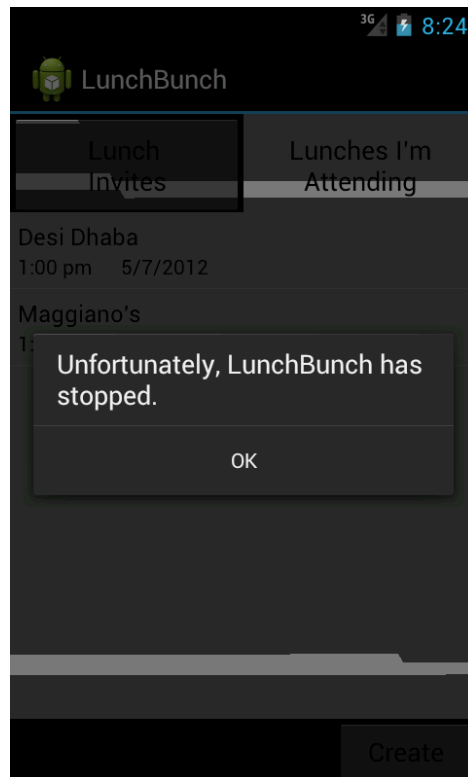## Usability heuristic: **Error reporting, diagnosis, and recovery (S)**

Severity: **Major**

Problem description:

Clicking on the "Lunches I'm attending" tab while already in that tab crashes the application. Clicking on the "Lunch Invites" tab while already in that tab closes the application with no error. Here is a partial error trace:

```
Caused by: java.lang.RuntimeException: Button ID unknown
at source.code.BrowseAttending.onButtonClicked
(BrowseAttending.java:67)
```
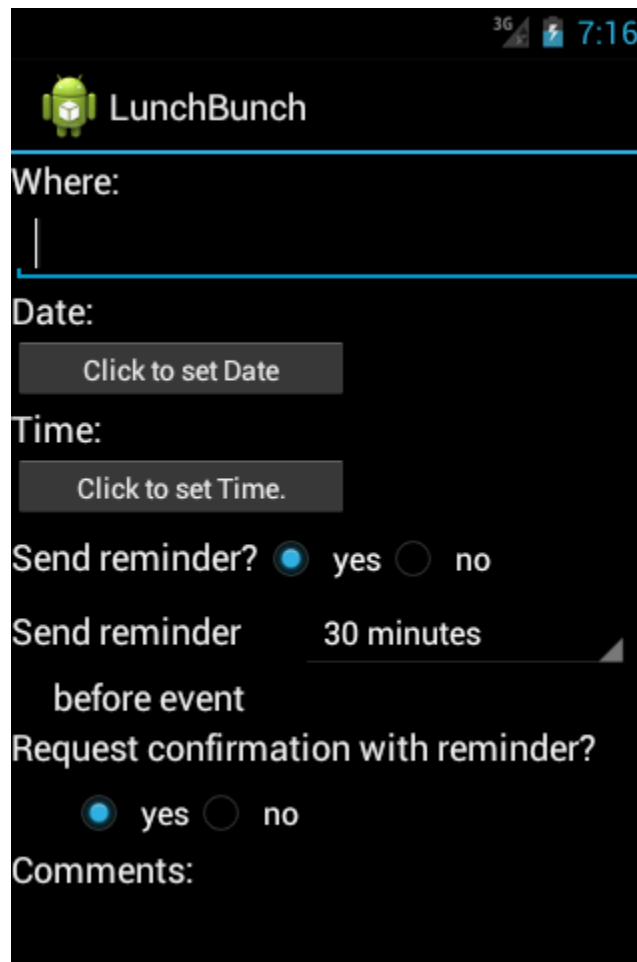

Suggested solution:

There is likely an error in how you have implemented the TabWidget and its children. If you aren't using a TabWidget, you probably should be.

Usability heuristic: Error reporting, diagnosis, and recovery (S)

Severity: Major

4

Problem description:

There are multiple display problems that occur on both small and large (but not medium size) displays. On small displays (such as the one used by the first android phone: the G1), the bottom section of the screen is cutoff making some buttons (like *done*) unreachable. On large displays (such as a tablet), the tab headers are removed from their content resulting in an awkward whitespace.

Usability heuristic: User control & freedom (S)

Severity: Major* (unless you are only supporting one screen size for the final product)

5

Problem description:

I find the ability to confirm an event confusing. If I have already accepted an invite, why do I then need to confirm it?
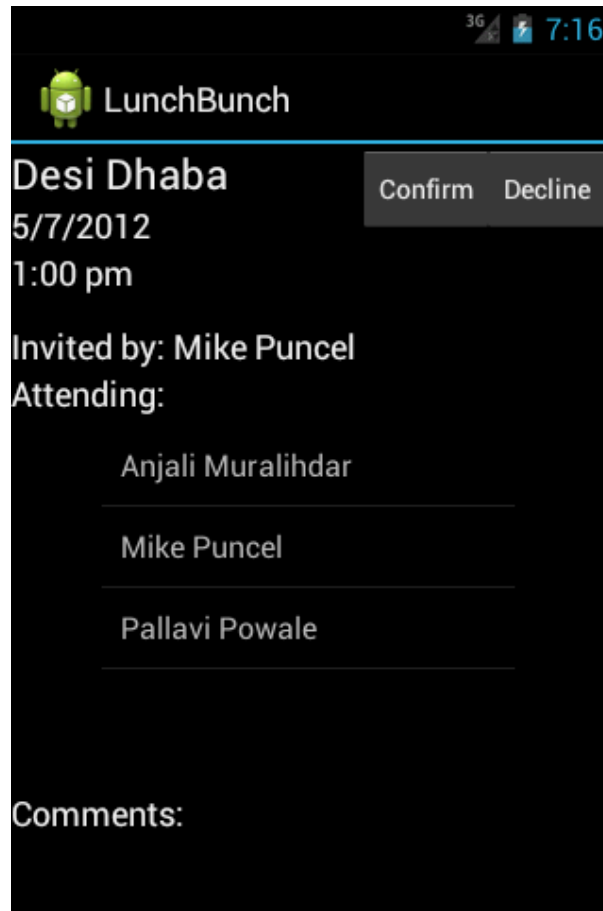
Usability heuristic: Help & Documentation (L)

Severity: Minor

Problem description:

The *confirm* and *decline* buttons are too close together. This is dangerous when combined with lack of error recovery mentioned in problem 2.

Usability heuristic: Error prevention (S)

Severity: Minor

7

Problem description:

The list of attendees provides a clicking affordance, when there is actually no reaction to clicking the attendee names.

Suggested Solution:

Either disable the clickable property of that ListView (if that's what you used), or make the names a series of TextViews (which are not clickable by default).

Usability heuristic: **Consistency & standards (L)**

Severity: **Minor**
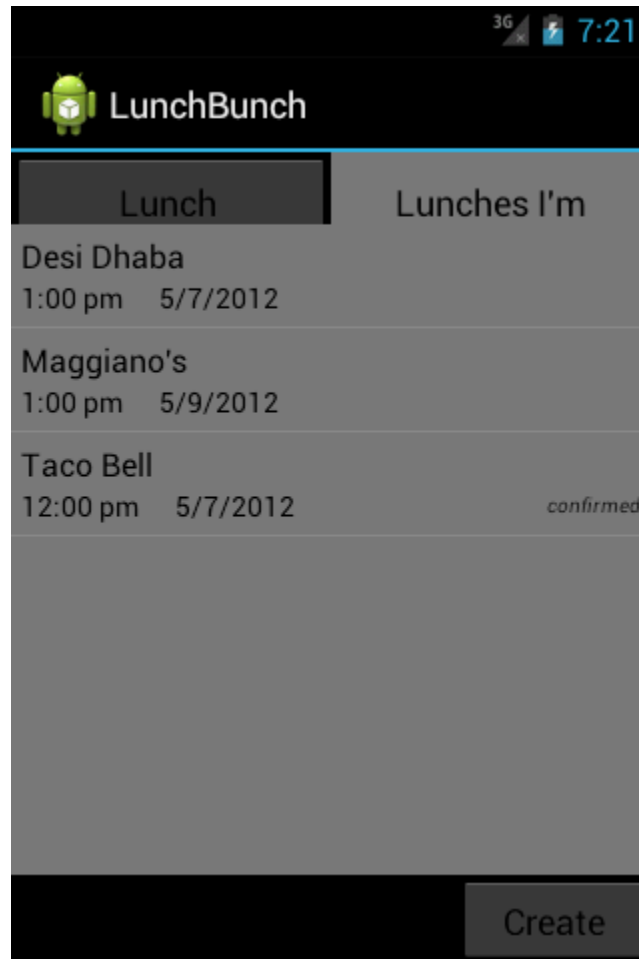
Problem description:

Android back button doesn't provide behavior consistent with its typical usage. In your application, the back button almost always closes the application, while typically the back button returns to the previously viewed screen. For example, if I click one of the events and then press the back button, it should return me to the list of invites, not close the application entirely.

Usability heuristic: **Consistency & standards (L)**

Severity: **Minor**

9

Problem description:

The "confirmed" text is small and dark, and the background is gray, providing poor contrast and readability.
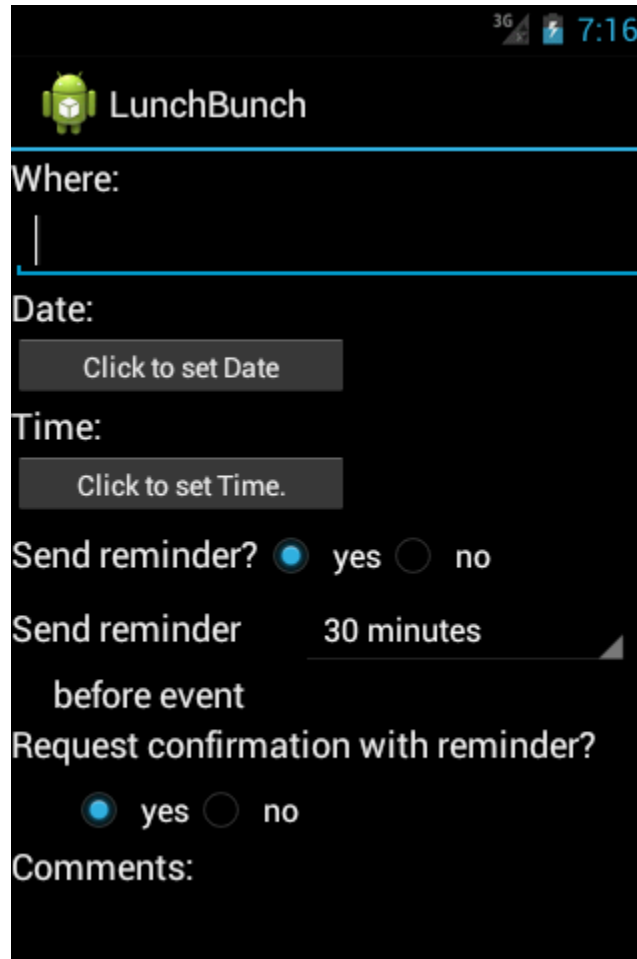

Suggested solution:

Make the text some shade of green and bigger if possible.

Usability heuristic: **Visibility of system status (S)**

Severity: **Minor**

Problem description:

The time and date fields on the create event screen have no default text.
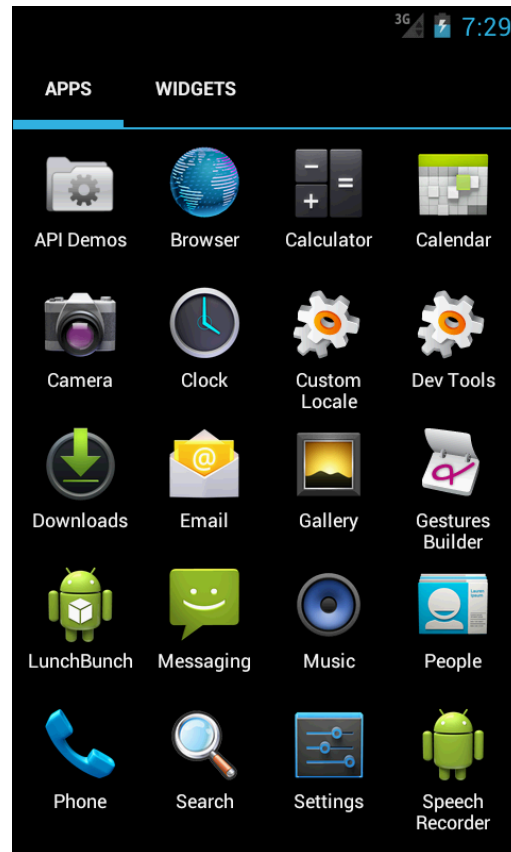

Suggested solution:

Add fragile default text. If the default time works for the user (like tomorrow at noon), then the user won't have to spend the time setting that field.

Usability heuristic: **Flexibility & efficiency (E)**

Severity: **Minor**

Problem description:

The icon for the application does not accurately represent this application (poor information scent), and is visually similar to the icon of other existing applications.
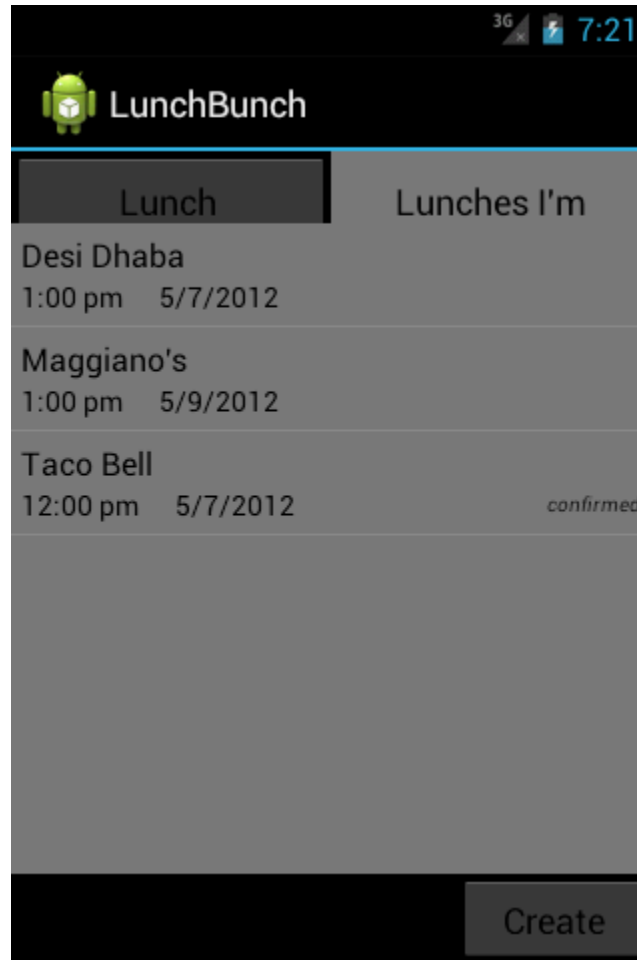

Usability heuristic: **Recognition, not recall (S)**

Severity: **Cosmetic**

Problem description:

The tab names: "Lunch invites" and "Lunches I'm attending" could be shortened. This relates to problem 4, as the current tab titles do not display correctly on devices with screen resolutions lower than HVGA (480x 320).

Suggested solution:

Shorten the titles to "invites" and "attending."

Usability heuristic: **Aesthetic & minimalist design (E)**

Severity: **Cosmetic**

Problem description:

The send reminder field should allow the user to determine the time instead of picking from the limited preset options.

Suggested solution:

Replace the field with a number picker followed by a drop down menu for the units (day, hour, minutes etc).

Usability heuristic: **Flexibility & efficiency (E)**

Severity: **Cosmetic**

Item description:

The use of differing font styles for the location and time of events in the invite and attending list provides useful contrast and better readability.

Usability heuristic: **Aesthetic & minimalist design (E)**

Severity: **Good**

Item description:

On the create event screen, setting the send reminder field to no hides all unnecessary fields. This prevents the user from entering erroneous data.

Usability heuristic: **Error prevention (E)**

Severity: **Good**

Other recommendations:

- I wouldn't use the package name `source.code`. Typical package names represent a hierarchal description that includes the authoring organization and application name. For more information check out the "Package Name" paragraph here: http://developer.android.com/resources/tutorials/hello-world.html

- You can solve some of your lack of space problems by making use of scroll views. This way even if you are displaying more content than a screen provides, the user can scroll down to view the obscured portion.

- I could be wrong, but I didn't see any part of your application that required compilation for Android 4.0+. Currently less than 3% of all android devices would be capable of running your application. If you aren't using any of the features of Ice Cream Sandwich, I would recommend changing your Android build target to something more encompassing such as Android 2.1 (Eclair).