

## McNits: Mike's Palm Programming Nitpicks, Part 3

by Mike McCollister

- Originally published circa June 2002 on the “Handheld Computing Developer” website.
- This article can be found at <https://MikeMcCollister.com/palm/articles/mcnits3/>.
- As this article are quite dated, most of the links will be invalid.

### Introduction

I was a strange child. I loved to organize. My room was clean, so clean that my parents would go out of their way to show it off to guests. For some reason they did not do that for the room that my two sisters shared. You see, my parents wanted to show off something that they were proud of. Why would a parent want to show off a child's room that looks like a member of the mob was looking for a video tape that could put a guy named Guido in prison for an exceedingly long time? The same goes for Palm OS programs. If you have a high-quality program without bugs and looks professional, your current user base will show the program to potential users. That is free advertising! This will result in greater exposure and thus may increase your sales and profits. This article is part three of a series of nitpicks that I have compiled that will help you recognize various traps that a Palm OS programmer should avoid. Heed these warnings and take pride in your work. Don't be satisfied with just “good enough”.

### User Interface Nitpicks

- Don't use menu items to toggle features. Since there is no check mark available for the menus, this can confuse the user. Instead, have a preference form that has the required check boxes.
- Don't put the “i” icon on your form if it is not modal. There is no guarantee that the font width is what you expect it will be and may possibly look bad. Also, on a color device, the icon will be black while the menu tab will be another color.
- If you have a form that contains only buttons, you need to rethink your user interface. Too many buttons on a screen makes it hard for the user to quickly decipher the functionality of the program.
- If you have a popup trigger at the upper right of the screen, such as a category list, make sure that it is right aligned and is at y-position of 1 and not 0, as shown in *Figure 1*. It looks a bit odd if it is at a y-position of 0.
- If you have a modal form that does not use the whole screen and is displayed over any other form, don't erase the screen behind your popup form. The user will wonder what happened to the original form. Also, the large white space is unappealing.



Figure 1: Good Popup Trigger Alignment

### Compatibility Nitpicks

- Don't go out of your way to support Palm OS 1.0 to 3.0 devices. It is best to support devices that use Palm OS 3.1 or better. The original Handspring devices use Palm OS 3.1 and cannot be upgraded to a newer version due to the OS being placed on a non-flash ROM, so this is a good minimum version to select. Palm OS 3.0 devices can be updated for free to version 3.3.

- If it is obvious which built-in application launcher category that your application should be placed, arrange it so that it will be automatically put into the appropriate category. If it is unclear what category a user would use, let it be placed into the default “Unfiled” category. Don't create your own category, like “McPrograms”, that is rude. Remember that there are only fifteen categories available in the built-in launcher.
- If your application changes some settings on the device, make sure those settings get set after a reset. For example, if your application creates an alarm, make sure that it survives a reset.
- There are many keyboards out there that can be added to a Palm OS device. If it makes sense respond to some keyboard specific characters. You can find a list of these characters in the Chars.h header file.

### **Bug Nitpicks**

- Make sure that your preferences use the same application ID as your application so that if the application is deleted, Palm OS knows which preferences to delete. If you don't do this, then there will be remnants left on the device after your program is deleted. This may cause problems if the user installs you program again later. There is a handy tool available, called “Preference Editor” (which can be found at <http://PalmGear.com>), which shows all preferences available on a device. You will be surprised how many applications don't follow this rule.
- Comment your code. Just because someone else will not be reading the code does not mean that you won't benefit from your own comments. I don't know how many times that I have looked at some old code and had no idea why I did something the way I did.
- With large programs or significant updates, recruit a group of beta testers. They will find bugs that you or the emulator will never find. Also, they will help you figure out what features are really needed.
- Don't release an update to your code the same day that you finish changes. Make the changes and let them sit for a few days. You might just find that bug that causes your user base to have serious problems.
- Use your programs frequently. If you are not a pig farmer who wants to keep track of how much Pig Poppers each hog eats, then don't write a program to keep track of this. You won't have a true appreciation on what a user might want, and besides it is more fun to program something that you use.

### **Support Nitpicks**

- If your program is licensed under the GNU GPL (General Public License), make sure that the source code is downloadable from your web page. This makes it easier for those who want to make changes which may make your program better for everyone. While the GNU GPL does not require that you have it readily available for download, you are required to supply it when asked.
- Respond to customer e-mail in a timely manner.
- Be polite and friendly in your e-mail correspondence, even if a user has asked the dumbest question in the world and did not read the manual. Being rude or impolite won't make the user smarter; however, you may lose a potential customer.

**Functionality Nitpicks**

- When a user exits your application, save the state of the application without requiring any input from the user.
- If appropriate, make your application respond to being launched multiple times by a hard button. Just like pressing the date book button multiple times will change the view of the date book.

**Extension (Hack) Nitpicks**

- If your application is an extension (hack) don't use the word "hack". It is too techie and may scare people away. Also, come Palm OS 5, hacks will not be possible, but source code may be modified so as to work as a notification.

**Audio Nitpicks**

- Applications should use the sound settings in the Palm OS Preference Panel and not have their own sound settings. These preference settings are available to your application and should be used accordingly. From the *Palm OS Companion*: "Because the user chooses sound preference settings, your application should respect them and adhere to their values. Further, you should always treat sound preferences as read-only values." (Thanks to Dave Lippincott for this nitpick.)

- 11110 -