**Cognitive walkthrough**

**Task 1** - The user doesn’t like the images currently being shown, so attempts to change the images.

**Process:**
1) To view options, the user mouses over the image and all the buttons appear. The user clicks on the information button in the lower right hand corner of the widget box. The widget spins around to display a few tabbed options. The options shown are Window Settings, Slideshow Customizations, and Miscellaneous.

2) The user clicks on the Slideshow Customizations option and is shown sliders that control image switching speed, zooming speed, zoom amount, and an area showing the current location from which the widget is pulling images.

There are two radio buttons if you want to pull images from Flickr Explore, or from your own Flickr feed. The third option is a Custom button, and there is a browse button next to a bar, and by clicking on it, you can choose any folder on your computer to take images from.

3) The user chooses the custom button, picks his inspiration folder and is returned to the previous screen. If he is done, he can click Save, this will return him out of the information button screens. Otherwise, he can choose Cancel, which will cancel any changes made.

**Task 2** - The user likes an image and wants to look at it more closely.

**Process:**
1) The user can click on the title button at the bottom of the picture if he wants more information about it. This causes a small transparent box to pop up with title and name of artist. If the slideshow progresses before they are done looking at it, you can go back with the back arrow key.

2) For more information, the user can click on the photograph and it will open the window, either by opening the containing folder, or if possible, can open it with the default picture viewer. If the images are from Flickr, this action will open the corresponding webpage.

3) A small box can pop up and ask the user if they want to make this image their permanent image for the time being. If this is annoying to the user, it can be turned off in the information part of the widget.

Possible problems with this project are:
1) I am a little worried about where I will be putting the title and name of artist when wanted. Should this be transparent over the image? Should it pop up if you even mouse over the photo, therefore eliminating the need for a title button? Maybe the user should have control over where the title bar should go?
2) If the user wants to use images from more than one folder, or perhaps specify exactly which pictures he wants to use and exclude others, will my program be able to handle that?

3) Should there be options for multiple images per slide? Or maybe the images could layer over the previous images like the Mac screensavers can do?

4) This tool MUST be simple to use. If its difficult, no one will want to deal with updating it every time new pictures are added or different functionality is needed.

5) How will random non-maximized placement of the widget affect other programs?

**Heuristic Walkthrough:**

**Validity of system status** – There should be a small circle counting down how much time until the next slide will show. This will allow the user to pause if the image is about to change, and will also give an expected time so that the user knows that the slide WILL eventually switch if they set the slide switching time for a high value.

**Match between the System and the Real World** – There will be a match because there will be minimum controls from the home screen, only an i for information, and arrows for forwards and backwards. Inside the menus, terminology will be straight-forward, with the use of things like z-values and opacity replaced with “Always on top?” and “transparency.”

**User Control** – Every step should be easy to back out of. If the user wants to exit, they will always have the option of saving and quitting out, or canceling everything and quitting out. This is only in the menus however; there is no interface to deal with from the front.

**Consistency and Standards** – This should be obvious, there is only one way to change things, and that is by clicking on the information button, which is a common theme in Mac widgets and most Yahoo widgets also.

**Help users recognize, diagnose, and recover from errors** – The only errors should really come from not choosing an image folder or choosing one with no images. If this is the case, we will show a message box containing the message information and asking if they want to try again or choose a Flickr site.

**Error prevention** – Errors might be easy to make if you are not too familiar with your file structure. Perhaps I can implement a method of showing how many images or which images are kept in which folders. It would be ideal if I could configure the browsing window to show thumbnails of images in each folder.

**Recognition Rather Than Recall** – Actions will not be visible unless the user mouses over the image, but this is necessary in order to maintain the integrity of images. Hopefully, on mouseover, the user will remember which button is the information button and will be able to know which things he can change. If I manage to incorporate the circle clock countdown, I could make it clickable and that could open the slideshow.
options, but it is debatable as to whether users would know that that was a button AND a feedback switch.

**Flexibility** – This was not considered too much for the first run-through of this, but if there is time, I could add shortcut keys to control the speed of images, and probably the title/name tabs as well. Other changes would be much less cost-efficient.

**Aesthetic and minimalist design** – Yes, this design is as minimal as possible, there should only be 4-5 buttons for the entire thing, and only 3 tabs of options on the back.

**Help and documentation** – Help could be provided with another button inside the information side of the widget, but not very much should be necessary, the design is straight-forward enough that everything you can do should visible by clicking on each button or tab.