Usability and Accessibility Changes Needed in Perform

June 2012

Written by Theresa Wilkinson

Contents

Executive Summary 5
Perform problem areas 5
Scrolling 6
Problem area: Three scrollbars, or target areas, very close together 6
Recommendations 6
Solution 1- Employee - Deductions 6
Solution 2 - Pull-down menu with scrollbar 6
Solution 3 - Pull-down menu with scroll buttons 8
Solution 4 - Transient scrollbar 8
UX research 9
Day 1 of Perform training 9
Age-related information from Perform usability tests 9
Age-related research 9
Problem area: Horizontal scrolling 11
Recommendation 11
Solution: Remove horizontal scrolling 11
UX research 12
Horizontal scrolling 12
Problem area: Screen does not scroll but has a scrollbar 15
Recommendations 15
Solution – Remove scrollbar because it is not needed 15
UX research 16
Scrolling 16
Problem area: Users are not scrolling on screens 16
Recommendations 18
Solutions 18
Solution: Vertical layout of screens and make them look "cut off" 18
Solution: Move the Save and Cancel buttons to the bottom right or near the last text field or control on all screens and modal windows 18
UX research 19
Day 1 of Perform training 19
From January Usability Test, Task 1 19
"F" Pattern 19
Visual flow 20
Primary and secondary actions 20
General UI Issues 21

Problem area: Small target area on Paydata grid to open employee drawer....21
Recommendations....22
Solution – Enable the clickability of the employee name on Paydata grid and enlarge the arrow....22
UX research....22
User comments from usability test findings: January test, Task 5....22
January test findings, End of test questions....22
March test findings....23
Anti-pattern: Tiny Targets....24
Tiny Click Targets....24
Problem area: Users could not tell the difference between red and orange outlined fields when fields were in error....25
Recommendations....25
Solution – Error message display....25
UX research....27
Information from March usability test findings....27
Error message behavior and display....27
Color blind research....28
How many people in US are color blind?....29
Problem area: Not all required fields marked with asterick....29
Recommendations....29
Solution – Mark fields either required or optional....29
UX research....30
Information from March usability test findings....30
Required form fields....30
Problem area: Users not seeing light-colored buttons....31
Recommendations....32
Solutions....32
Solution: Screens with no information....32
Solution: Screens with information....32
Solution: Make the Additional Payrun indicator more obvious....34
UX research....34
March usability test results, Task 3....34
Primary and secondary actions....35
Visual flow....36
Problem area: Confusing layout and no feedback....36
Recommendations....37

Solutions 37
Solution for confusing screen layout 37
Solution for lack of feedback 38
UX research 38
Day 1 of Perform training 38
March usability test results, Task 2 38
March usability test Task 2: No feedback 38
March usability test Task 2: Not understanding screen layout 39
No feedback 39
Problem: Reset button 40
Recommendations 40
Solution for reset button 40
UX research 41
Reset button 41
Problem: No Undo button 42
Recommendations 42
Solution: Add an Undo and Redo function on all forms. 42
UX research 42
Undo button 42
Day 1 of Perform training 42

Executive Summary

Accessibility of a web application describes how well users with limitations can access it. These limitations can be technical, such as a slow Internet connection, or an outdated operating system. They can also be physical, like color blindness, or for example a handicap that makes it impossible for users to operate a mouse. An accessible web application offers solutions for these limitations, like low resolution images to ensure a short loading time, screen reader optimization, or the compatibility with alternative input devices.

From the Mac OS X Human Interface Guidelines: "Making your app usable by people with disabilities (that is, making your app accessible) is the right thing to do. Making your app accessible benefits you, too, because in many places accessible apps are the only apps that governments and some institutions can purchase."

This document details some of the accessibility and usability issues within the Perform application. Problem areas were identified through usability testing, heuristic review, and Perform training feedback.

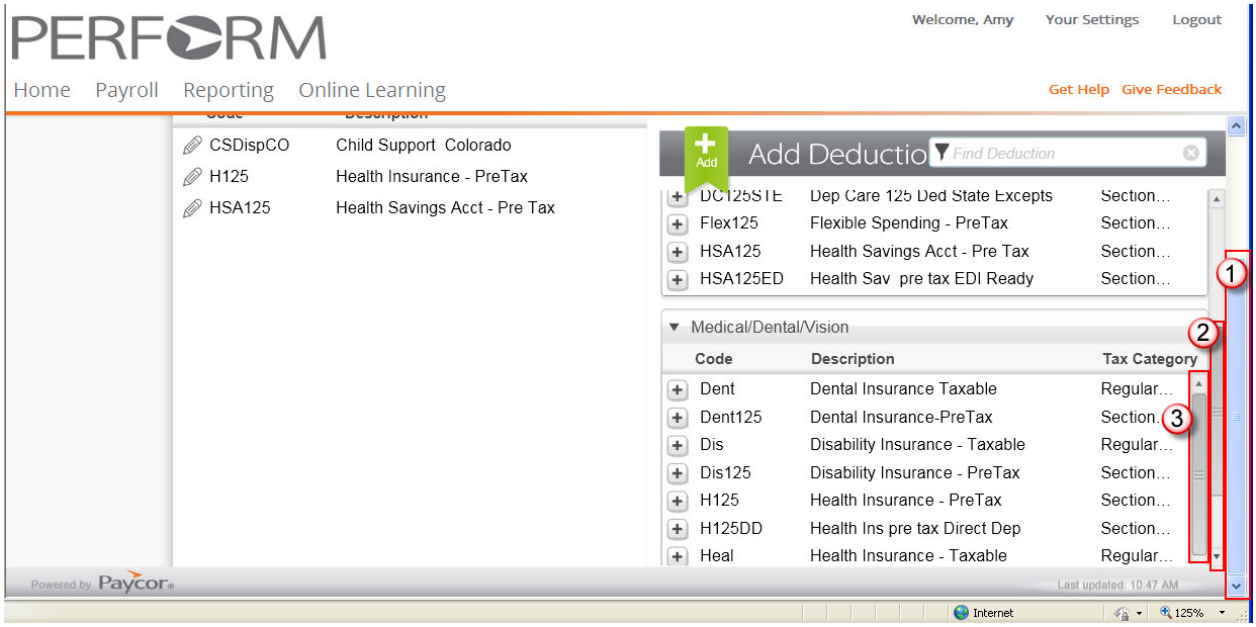
Proposed solutions to fix the issues are also presented, along with the UX research used to determine each solution.

Perform problem areas

- Three scrollbars, or target areas, very close together
- Horizontal scrolling
- Screen does not scroll but has a scrollbar
- Users are not scrolling on screens
- Small target area on Paydata grid to open employee drawer
- Users could not tell the difference between red and orange outlined fields when fields were in error
- Not all required fields marked with asterisk
- Users not seeing light-colored buttons
- Confusing layout and no feedback
- Reset button
- Undo function

Scrolling

Problem area: Three scrollbars, or target areas, very close together



Three scrollbars on Configure Company – Deductions screen

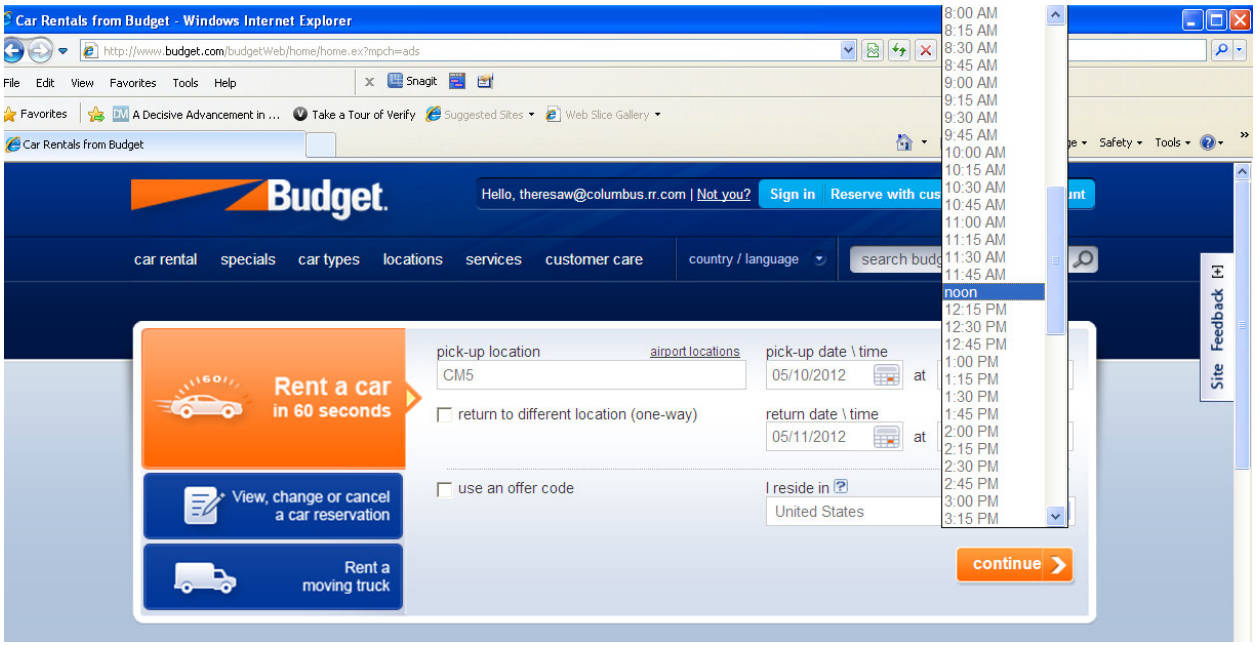
Recommendations

Solution 1- Employee - Deductions

Redesign all screens in the Configure Company section look and function like the Employee Deductions screens, which **all users in the March usability test found easier to use and understand**. For accessibility reasons, this is the preferred solution to this problem.

Solution 2 - Pull-down menu with scrollbar

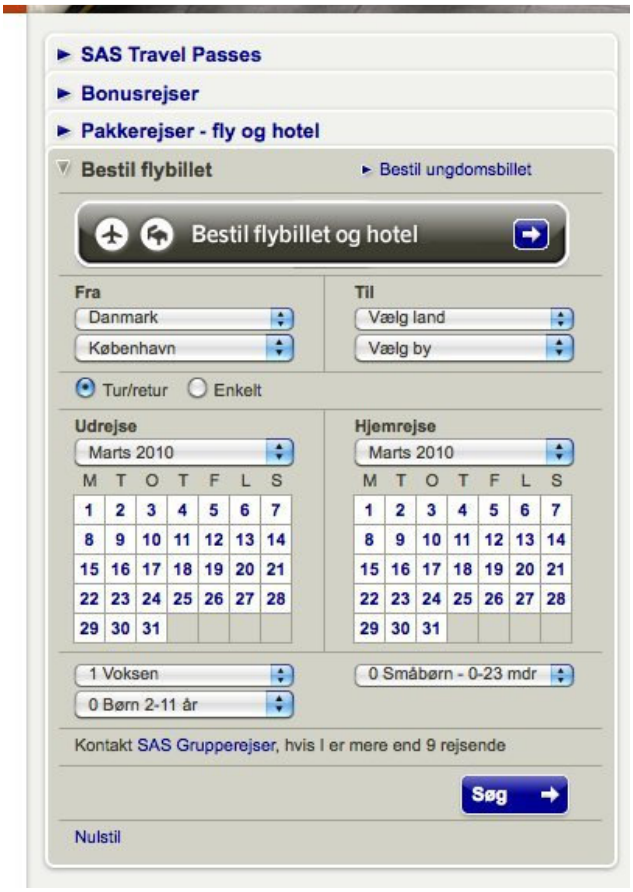
Users can view the information in the pulldown by scrolling up and down.



Pull-down menu with scrollbar example

Please note the distance between the browser scrollbar and the pulldown menu scrollbar ensuring the user does not accidently click on the wrong one.

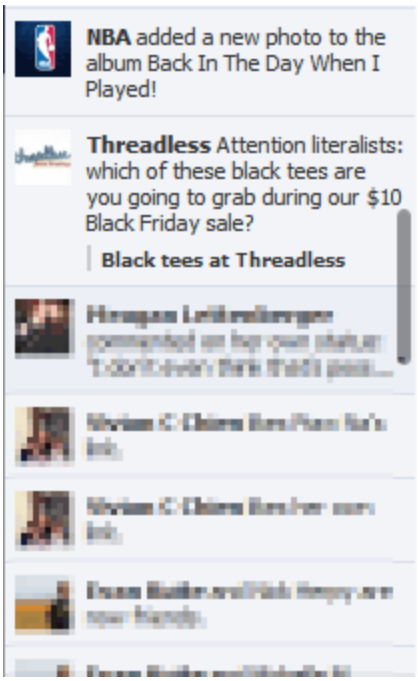
Solution 3 - Pull-down menu with scroll buttons



Pulldown menu with scroll buttons example

Solution 4 - Transient scrollbar

Transient scrollbars are not persistently visible, in that they disappear when your mouse stops moving.



Facebook transient scrollbar example

UX research

The latest U.S. Census Bureau brief on data from the 2010 Census shows seniors increasing faster than younger populations, raising the nation's median age from 35.3 in 2000 to 37.2 in 2010, with seven states having a median age of 40 or older.

Day 1 of Perform training

- **Scroll bars definitely caused a problem – users couldn't easily see them**

Age-related information from Perform usability tests

In the January test, of the ten users – **80% were 45 or older.**

- Four users were aged 45 – 55
- Four users were aged 55 – 65

In the March test, of the eight users – **75% were 45 or older.**

- Four users were aged 45 – 55
- Two users were aged 55 – 65

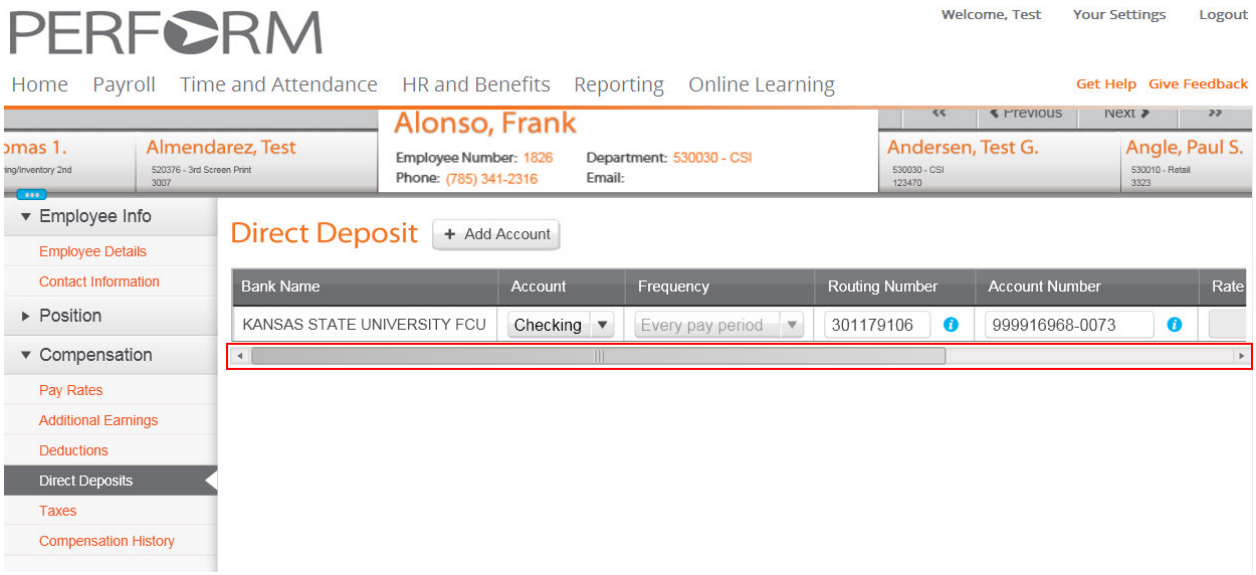
Age-related research

Article Name	Site	Research
Age-Related Research-	http://www.usability.gov/articles/	"At least 70% (of the population) will live past age 65, and about 40% past age 80. Unfortunately, there are

Based Usability Guidelines	newsletter/pubs/112005news.html	definite, predictable degenerative effects of aging on a person's ability to effectively and efficiency interacts with Web-based systems. One effect of aging is diminished vision – loss in near vision, reduced field of view and contrast sensitivity, and reduced color sensitivity in the blue-green range. Psychomotor impairments include increased response time, and poorer tracking with a mouse. Also, older users experience reduced attention, and reduced memory capability, including working memory, episodic memory and procedural memory. Studies show that many web sites clearly are not designed with the aged in mind." Research-Based Guidelines Some of their most important guidelines for seniors included: • **Provide large targets, e.g., larger clickable graphics**, for the mouse pointer. • To reduce the number of clicks, **do not require double clicking, use pull-down menus, nor have a deep hierarchy**. • **Avoid the need to scroll down for information.** • Put most links in a bulleted list (not tightly clustered), and **differentiate between visited and unvisited links**. • **Use few colors, and avoid using blue and green tones.** • When text must be read or scanned, use 12-14 point sans serif (Helvetica, Arial) black text on a white background.

See also: Anti-pattern: Tiny Targets, Tiny Click Targets

Problem area: Horizontal scrolling

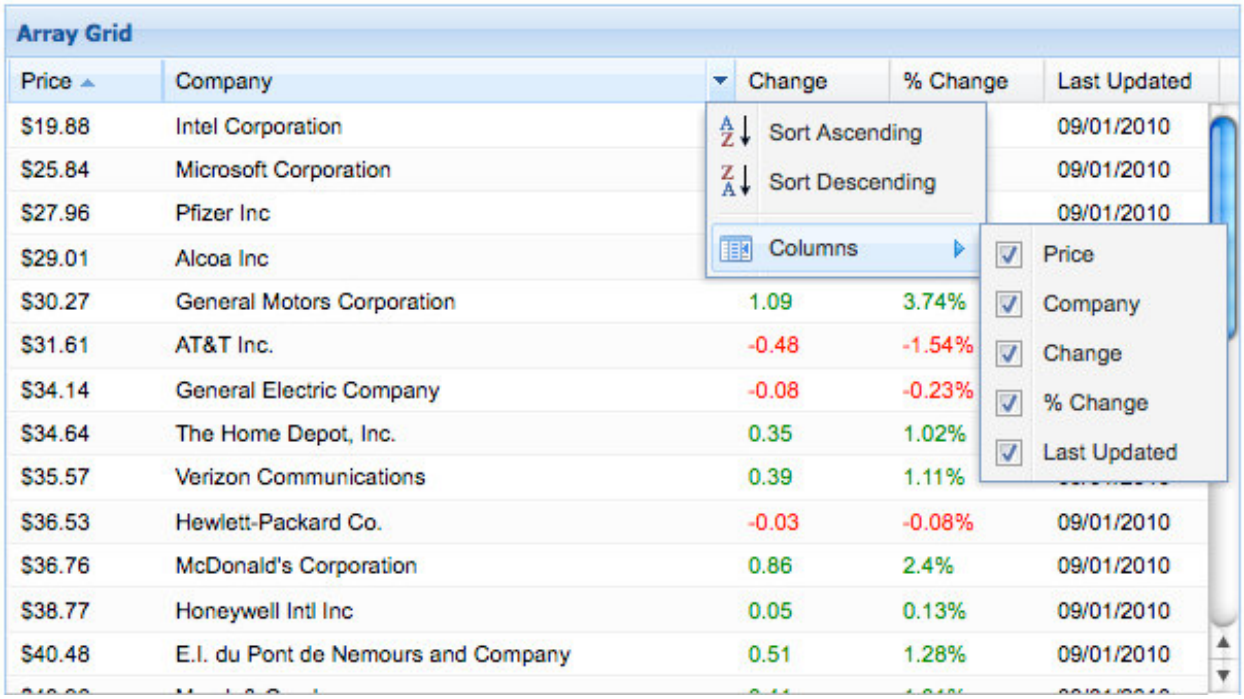


Horizontal scrolling on the Direct Deposit screen

Recommendation

Solution: Remove horizontal scrolling

- Display only part of the bank name and include an information icon. When the user rolls over the information icon, display full bank name.
- Organize the most important columns to the left.
- Experiment with frozen/fixed columns, so if the person does need to horizontally scroll, they can keep context.
- **Only show a set number of columns in the default view (so there is no horizontal scrolling in the default view) and offer a Customize option** so the person can choose to hide or show more columns.



Example of solving super wide table problem

UX research

Horizontal scrolling

Article Name	Site	Research
Ultimate guide to table UI patterns by Theresa Neil	http://designingwebinterfaces.com/ultimate-guide-to-table-ui-patterns	Super Wide Tables Based on a lot of design work Bill Scott and I did for the airline industry, I would instead propose instead applying these principles: • **Organize the most important columns to the left**. • **Experiment with frozen/fixed columns**, so if the person does need to horizontally scroll, they can keep context. • **Only show a set number of columns in the default view (so there is no horizontal scrolling in the default view) and offer a Customize option** so the person can choose to hide or show more columns. ExtJs has this built into the column dropdown; I usually add a customize button to the table toolbar

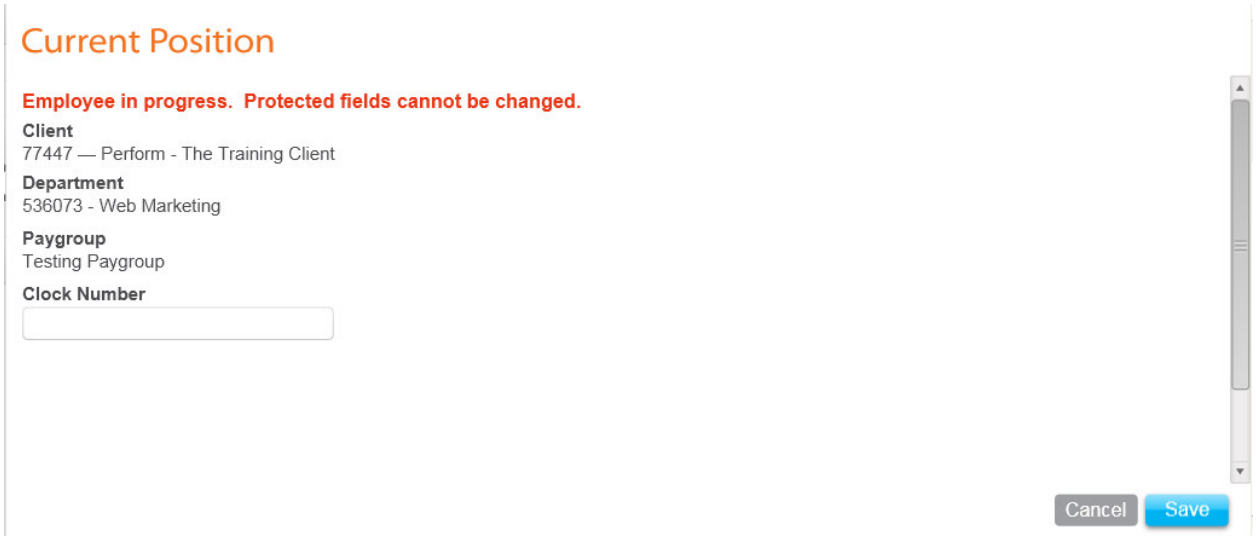
		with Hide/Show column functionality. • Offer resizing of columns. • Offer rearranging of columns. • If you have a table with some columns editable and other read-only, group editable with editable, read-only with read only. • Don't abbreviate column titles, reduce spacing or padding, or drop to a smaller font to fit your table on the screen. That won't help anyone use your app. • Try out fat rows like this example from Survs. Instead of having a column for title, created by, created on, last updated on, all of that information is in the second column. Good visual design can help organize the information in a more meaningful way which makes your data easier to scan. • Use a summary row to chunk the data if appropriate. I know this won't make your table any narrower, but it might make it more readable.
Scrolling and Scrollbars by Jakob Nielsen	http://www.useit.com/alertbox/20050711.html	We know from user testing that **users hate horizontal scrolling** and always comment negatively when they encounter it. **Customer satisfaction is surely reason enough to avoid horizontal scrolling**. There are two other reasons as well: • On the Web, **users expect vertical scrolling**. As with all standard design elements, it's better to meet user expectations than to deviate. • **When pages feature both vertical and horizontal scrolling, users have to move their viewport in two dimensions, which makes it hard to cover the entire space. For people with poor spatial visualization skills, it's especially challenging to plan movements along two axes across an invisible plane**. (Typically, users score lower than designers on spatial reasoning and visualization tests.) In contrast, one-dimensional scrolling is a simple way to move

		across content without advance planning: you just keep moving down. In any case, **all key information should be visible on the initial screen** because scrolling can cause **accessibility problems**: • The additional action that scrolling requires can be difficult for users with motor skill impairments. • Low-literacy users can't easily reacquire their position in the text after it moves. • Elderly users often have trouble getting to the right spot in scrolling menus and other small scrolling items.
Horizontal Attention Leans Left by Jakob Nielsen	http://www.useit.com/alertbox/horizontal-attention.html	People spent **more than twice as much time looking at the left side** of the page as they did the right: • **Left** half of screen: **69%** of viewing time • **Right** half of screen: **30%** of viewing time The remaining **1%** of viewing time was spent to the right of the initially-visible 1,024 pixels. Such information is visible only after horizontal scrolling, and the minute amount of attention it attracts confirms the guideline to **avoid horizontal scrolling** (mistake #3 of 2002). Information to the right of the initially-visible area is in essence "below the fold," except that they are beyond a right-hand fold instead of a bottom-of-window fold, and thus not literally "below." **Another way of looking at vertical vs. horizontal scrolling is that users allocate 20% of their attention past the fold in the vertical dimension but only 1% past the fold in the horizontal dimension**. (I would actually classify **horizontal scrolling** as much more than twenty times worse, because **it also *annoys* users, besides attracting less attention**.)

See also: Age-related information from Perform usability tests, Age-related research, Anti-pattern: Tiny Targets, Tiny Click Targets

Problem area: Screen does not scroll but has a scrollbar

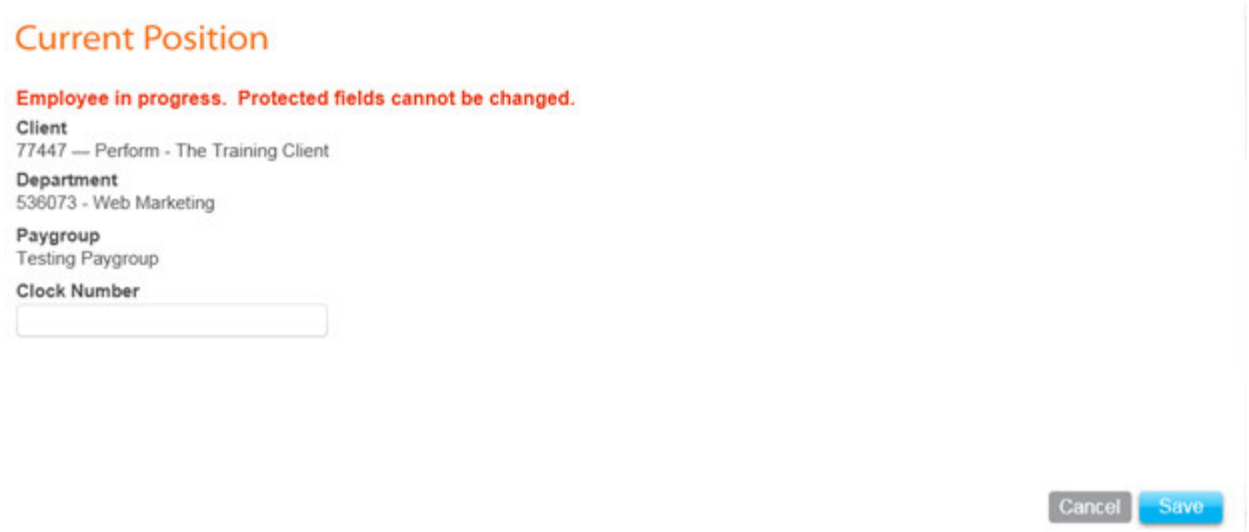
This could be confusing to users who try to scroll down the screen but cannot.



Current Position screen with all information displaying but with a scrollbar

Recommendations

Solution – Remove scrollbar because it is not needed



Current Position screen with no scrollbar

UX research

Scrolling

Article Name	Site	Research
Scrolling and Scrollbars by Jakob Nielsen	http://www.useit.com/alertbox/20050711.html	1. **Hide scrollbars if all content is visible. If people see a scrollbar, they assume there's additional content and will be frustrated if they can't scroll.**

See also: Age-related information from Perform usability tests, Age-related research, Anti-pattern: Tiny Targets, Tiny Click Targets

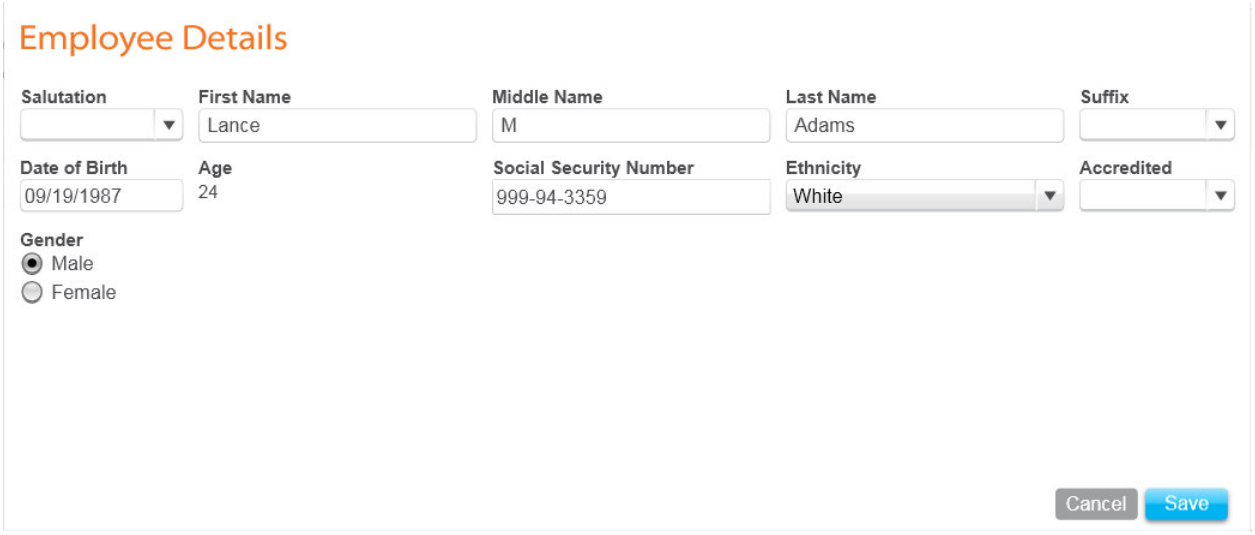
Problem area: Users are not scrolling on screens

Issues:

1. Screens do not appear as though any information is "below the fold." Since users scan a page using an "F" pattern, they miss the scrollbar on the right side of the screen because there is no other visual indication that information is below the fold.

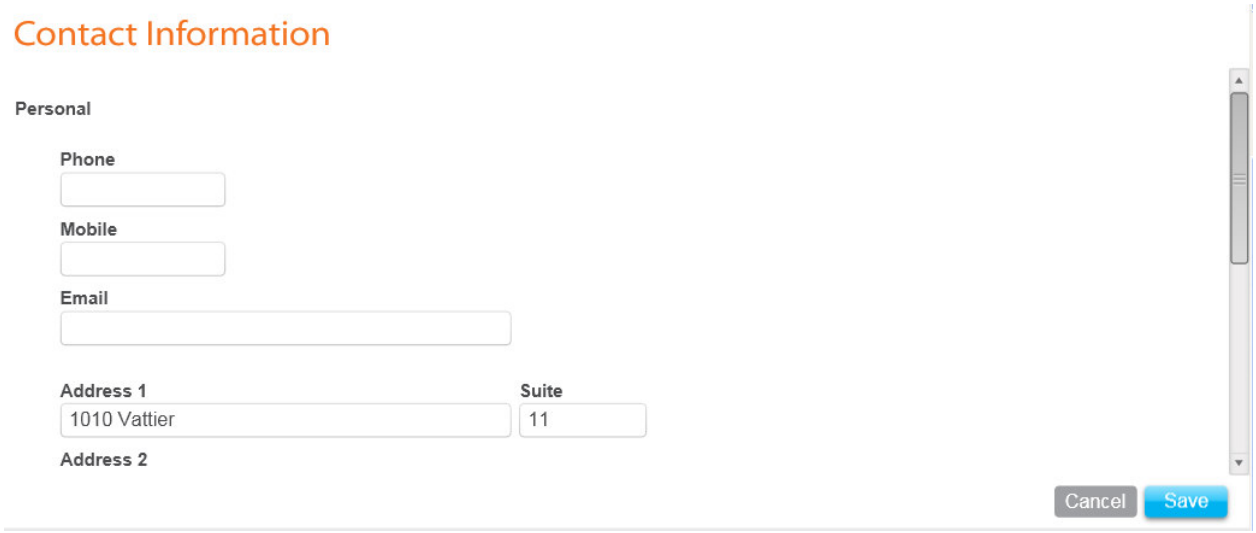
"F" pattern and "cut off" look example

2. Some screens have a horizontal layout while others have a vertical layout. With horizontal layouts, all information is above the fold and the Save and Cancel buttons "appear" to be at the bottom of the screen. Having both screen layouts in the application confuse users into thinking there is no information below the fold of a vertical screen (this was an issue in the January usability test).



Horizontal layout example

3. Save and Cancel buttons are locked in position but not at the bottom of the most screens. Buttons locked on a screen interrupt the visual hierarchy and the visual flow of the screen: "A good visual hierarchy uses focal points (in this case, buttons) to pull eyes to the right place in the right order." The locked buttons distract the user from following the correct sequence. **This visual flow tells the user to stop when they see the buttons because buttons are usually located near the last text field or control on a screen**.



Vertical screen with Interrupted visual flow example -- this screen does scroll

Users, in both usability tests and on screens that were laid out quite differently, did not scroll to find additional information below the fold due to the issues stated above.

Recommendations

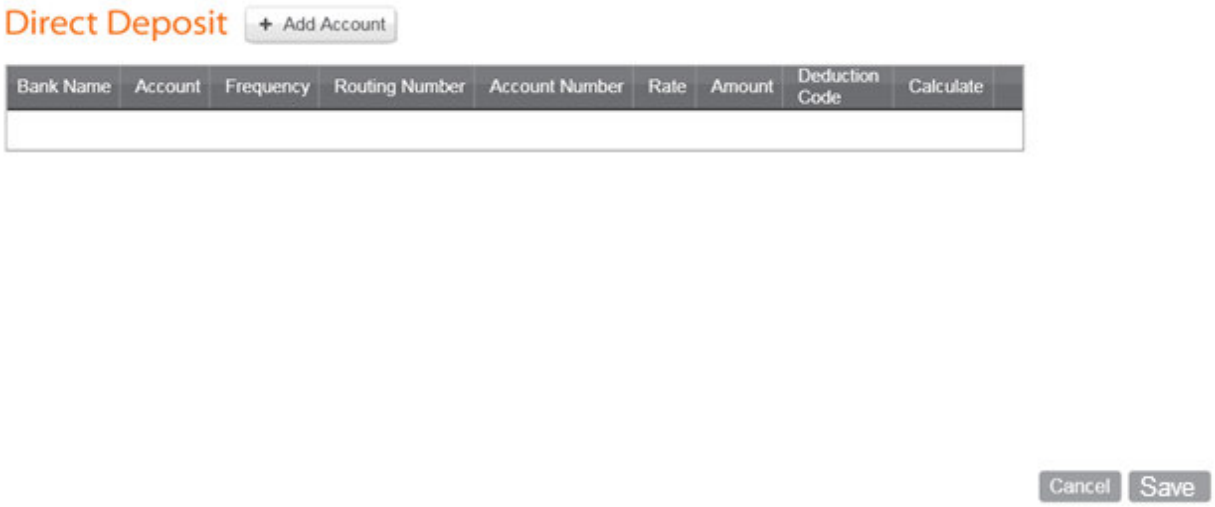
Solutions

Solution: Vertical layout of screens and make them look "cut off"

All screens should have the vertical layout and appear to look "cut off" if information goes below the fold. Example: on the Contact Information screen, the top of the Address 2 text box should also appear as a visual indication of more information below the fold.

Solution: Move the Save and Cancel buttons to the bottom right or near the last text field or control on all screens and modal windows

Unless a user makes a change on a screen, the Save and Cancel buttons should appear inactive or "grayed out." For more information on this issue, see Problem area: Users not seeing light-colored buttons.



Direct Deposit with Inactive Save and Cancel buttons

UX research

Day 1 of Perform training

- **Scroll bars definitely caused a problem – users couldn't easily see them**
- Scrolling on compensation history did not go over well – thought it was not user friendly for clients

From January Usability Test, Task 1

- User 10 - "Ah, I did not see the (scroll bar) earlier. Once you start using it, you would be fine." "It was hard to see (on the page)."
- User 12 – (Is there anything to indicate that there is more information {on the Contact Information screen)? She is scrolling through employee cards at the top. I show her the scroll bar – Did you see this here?) "No." (What would make it more obvious that there is more information on this screen? Did you think that this was the end of the screen because of the buttons right here?) "Yup, yup. I thought that was the only info on that page. I didn't know if maybe it was on something other than this screen."

"F" Pattern

Article Name	Site	Research
Introducing the F-Layout	http://webdesign.tutsplus.com/articles/ design-theory/understanding-the-f-layout-in-web-design/	The F-Layout relies upon various eyetracking studies for it's foundational concept. **These scientific studies show that web surfers read the screen in an "F" pattern – seeing the top, upper left corner and left sides of the screen most… only occasionally taking glances towards the**

		right side of the screen. These eyetracking studies argue in favor of **placing the most important elements of your site on the left side** of the design. Allow me to walk you through the general behavior pattern: • **Visitors start at the top left** of the page. • Then **they scan the top of the site** (navigation, subscription, search, etc.) • **Next they move down, reading the next full row of content**… all the way to the sidebar. • Last, surfers enter a "scanning pattern" once they hit the bulk of the site content.

Visual flow

Book Name	Research
Designing Interfaces by Jennifer Tidwell, pages 136-137, Visual Flow	**Visual flow deals with the tracks that readers' eyes ternd to follow as they scan the page.** It's initmately related to visual hierarchy, of course – a well-designed visual hierarchy sets up focal points on the page wherever you need to draw attention to the most important information. As a designer, **you want to be able to control visual flow on a page so that people follow it in approximately the right sequence**. Focal points are the spots your eyes can't resisit going to. You tend to follow them from strongest to weakest, and skillfully designed pages have only a few – too many focal points dilute the importantance of each one. **A good visual hierarchy uses focal points to pull eyes to the right places in the right order**. Likewise, **if you're designing a form, arrange the controls along a continous path and put "I'm finished" buttons (OK, Cancel, Submit, Buy, etc.) at the end of the line.**

Primary and secondary actions

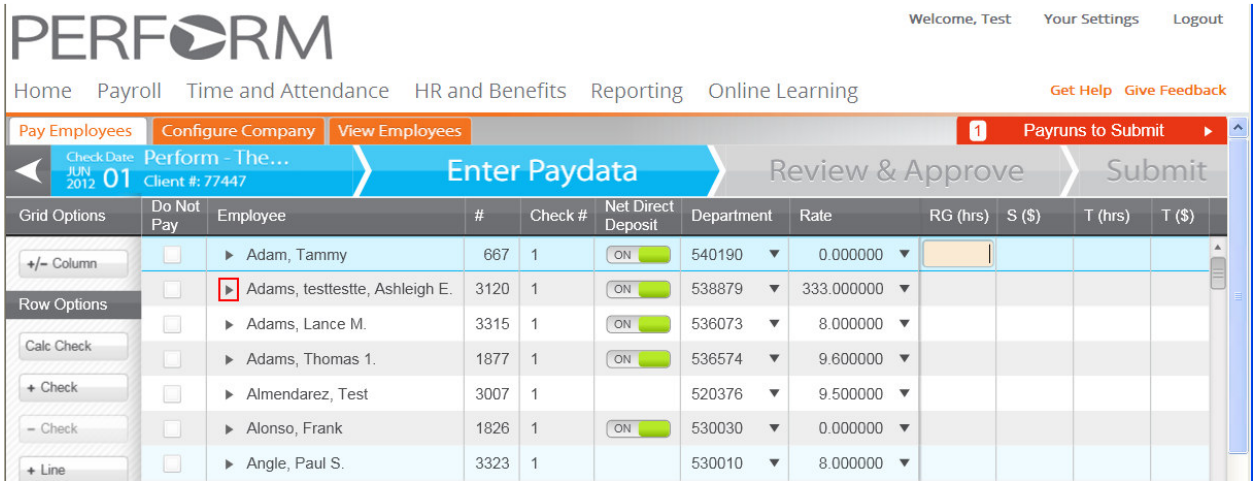
Book Name	Research
Designing Interfaces by Jennifer Tidwell, pages 257-258, Prominent "Done" Button	**What: Place the button that finishes a transaction at the end of the visual flow; make it big and well labeled.** **Use when:** Whenever you need to put a button such as Done, Submit, OK, or Continue on your interface, you should use this pattern, More generally, **use a visually prominent button for the final step of any transaction**. **Why: A well-understood, obvious last step gives the user a sense of closure**. There's no doubt that the transaction is done when that button is clicked; don't leave them hanging, wondering whether their work took effect. **Making the last step obvious is what this pattern is really about.** **How: Place the button where the user is most likely to find it. Trace the task flow down through the page or form or dialog box, and put the button just beyond the last step. Usually that will be on the bottom and/or right of**

	the page. **In any case, make sure the button is near the last text field or control.**

Article Name	Site	Research
Primary & Secondary Actions in Web Forms, by Luke Wroblewski	http://www.lukew.com/ff/entry.asp?571	**Conversely, the alignment of actions with a form's input elements provides a clear path to completion that helps people complete forms faster. Be conscious of where you place form actions as primary actions** directly aligned with input fields tend to increase completion rates and the less time people have to spend on your forms, the happier they will be.

General UI Issues

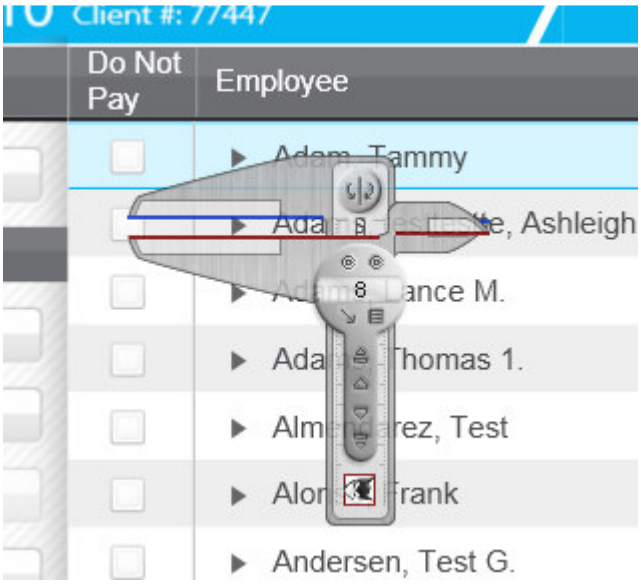
Problem area: Small target area on Paydata grid to open employee drawer



Employee drawer arrow on Patadata grid

Recommendations

Solution – Enable the clickability of the employee name on Paydata grid and enlarge the arrow



8x8 arrow on Paydata grid

UX research

User comments from usability test findings: January test, Task 5

	Issue	
4	Problem finding arrow next to employee name to open EE drawer	User 7 – "**Took me a little while to find the little arrow next to his name**. That is nice though that you can do it on the fly. You did not have to get out of the screen and go back and add it and then go back into payroll." (What would make it more obvious that you could do that.) "Just knowing you could, once you know it is there." (**When you looked at that arrow, you did not realize that is what it did?) "I really did not notice it.** So I am looking at where can I add this. When you said you don't think you can do this from this page, I am looking a little harder here. Then when I saw the arrow next to the name, okay well then you can add it in there. (EE drawer) "I do like this better than what we have now."

January test findings, End of test questions

User #	Q1	Q2	Q3
5	Yes. I think that what	Yes	(**Was it obvious to find that (arrow on the employee name)?** And what it does?) "**I don't know if I would say it**

	is hard is that you have changed some of the wording from what it is currently. Which is okay **but then there will be a learning curve**."		**was obvious but I would expect it. So the arrows do mean you can click on it?**"
9	"Yes but it is somewhat of a limited view. (How so?) I like to see everything at once."	Yes	Stopped recording: EE drawer: **This user did say that he was confused on how to open the EE drawer. He singled clicked and double clicked on the name and clicked within the row but nothing happened. He expected clicking on the employee name to open it**.
10	Yes	"**I am blind** and being on a laptop, it does not appear to be all that great."	EE drawer: After I stopped recording, this **user did say that he was confused on how to open the EE drawer. He singled clicked on the name and clicked within the row but nothing happened. Only clicking the arrow, which he thinks he did by accident because he cannot see well, opened the drawer. He expected clicking on the employee name to open it.**
11	See Q 3	"**Some of the things are hard for me to see on the screen**."	

March test findings

- As in the January usability test (user 7, user 10, user 9), four users (User 4, User 6, User 8, User 11) in this test clicked on the employee name on the Enter Paydata screen expecting something to happen. Three users became confused when

nothing happened and failed Task 5 due to this issue. **Prior to this task, as in the January test, all users first interacted with the View Employees screen. On that screen, users could click on the employee name to interact with the system. In both tests, almost half the users (7 out of 18) seemed to expect this functionality on Enter Paydata as well**.

Anti-pattern: Tiny Targets

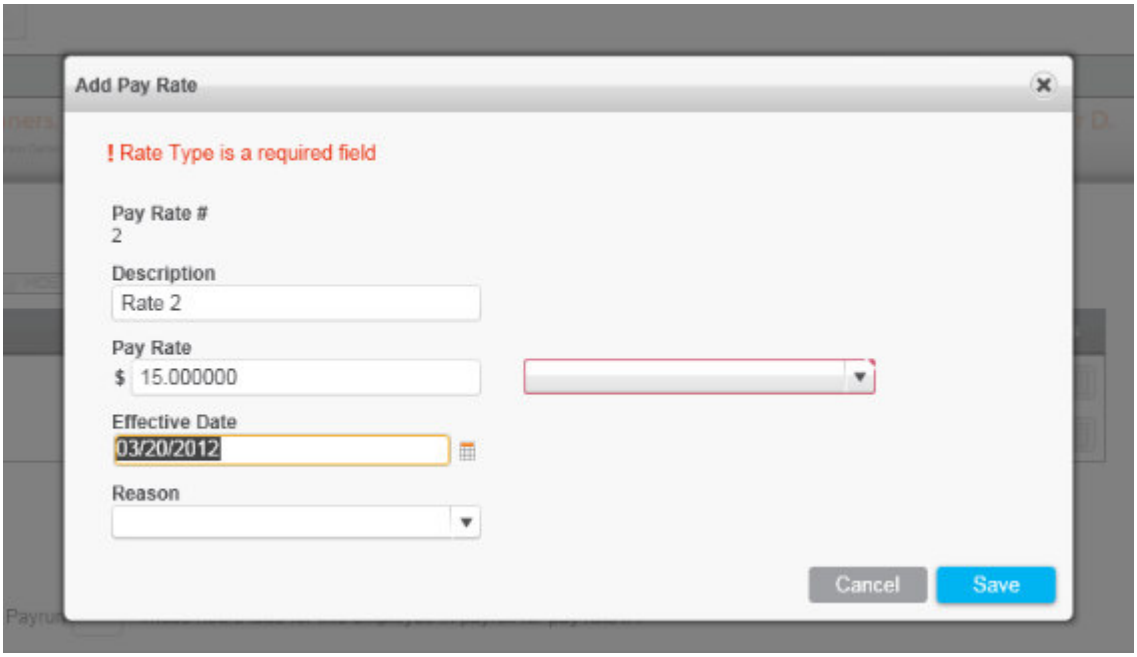
Book Name	Research
Designing Interfaces by Jennifer Tidwell, page 97	"At the beginning of this chapter, we discussed **Fitt's Law. Recall that the time to acquire a target is a function of both distance and size**. Even if tools are placed close by in context, don't forget to make them large enough to target. **Tip: Never use small targets. Make targets large enough to notice and interact with."** "**The arrow is tiny (8x8 pixels)**... Providing Tiny Targets makes interaction much more difficult... **Alternately, providing a larger target for the arrow would improve findability and targeting.**"

Tiny Click Targets

Article Name	Site	Research
Top-10 Application-Design Mistakes by Jacob Nielsen	http://www.useit.com/alertbox/application-mistakes.html	**An associated problem here is click targets that are so small that users miss and click outside the active area. Even if they originally perceived the associated affordance correctly, users often change their mind and start believing that something isn't actionable because they think they clicked it and nothing happened.** (Small click zones are a particular problem for old users and users with motor skill disabilities.)

See also: Age-related information from Perform usability tests, Age-related research

Problem area: Users could not tell the difference between red and orange outlined fields when fields were in error

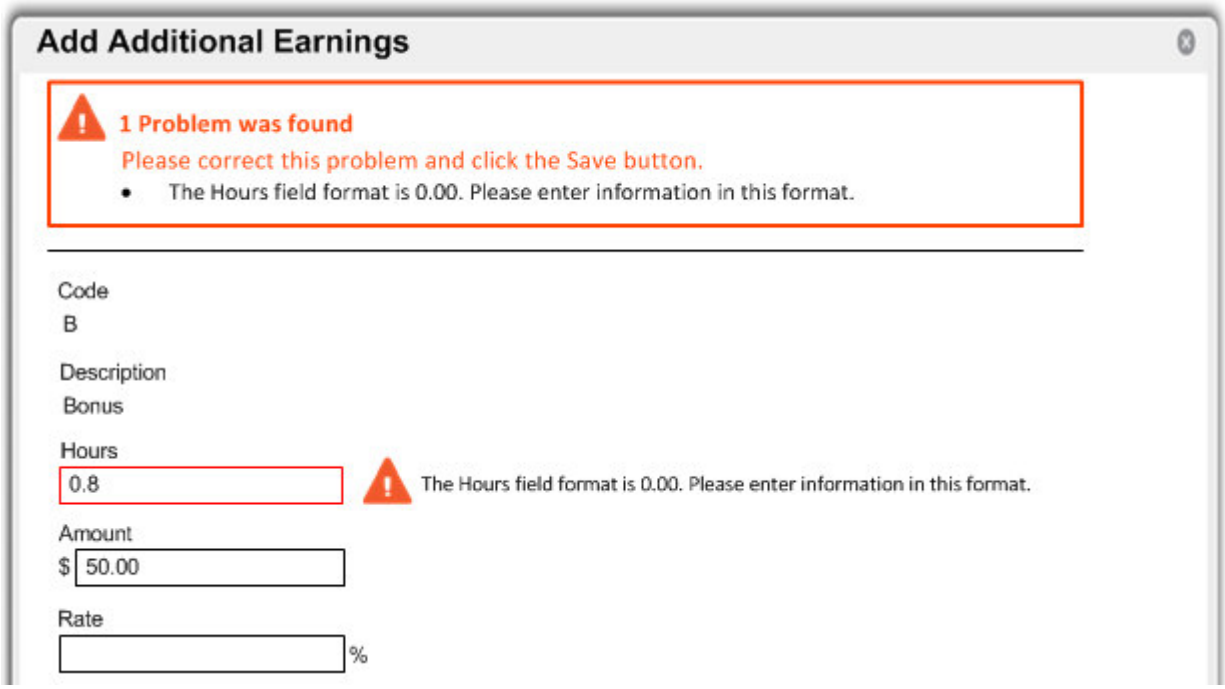


Add Payrate screen with red and orange field outlines

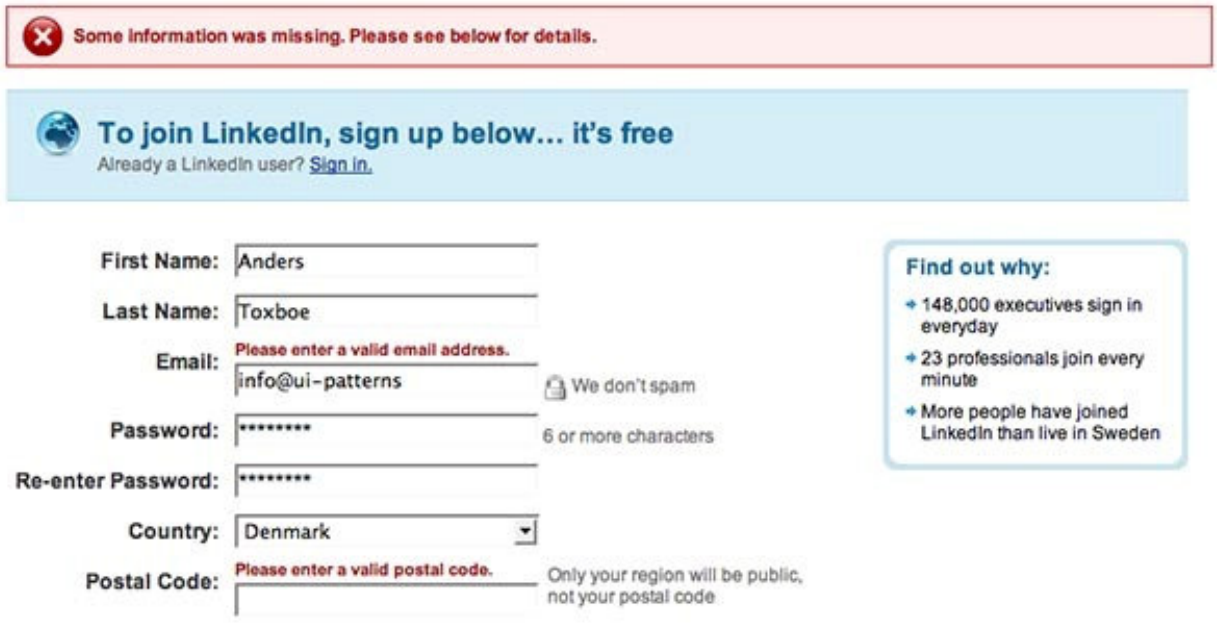
Recommendations

Solution – Error message display

Make the red outline around fields in error much darker (thicker line). Display an icon with error message beside the field in question. Display both when user clicks Save.



Error message example 1



Error message example 2

UX research

Information from March usability test findings

- **Some users could not tell which field was in error due to similar colors of field outlines**. To most users, the red outline of missing information was not obvious. **Two users tried to "correct" the field outlined in orange and not in red.** This could have been because they were color blind and these colors looked very similar.
- **Although the error description at top of screen did display, the error message beside the field did not display until user clicked into field. User 4 did not seem to see field outlined in red and tried to correct the wrong field**. If error message beside the field had displayed, this, and all other errors, would have been corrected much faster.

Error message behavior and display

Book Name	Research
Designing Interfaces by Jennifer Tidwell, page 388-389, Getting Input from Users: Forms and Controls	**What:** Place form messages directly on the page with the form itself; **mark the top of the page with an error message, and if possible, put some indicators next to the originating controls.** **Use when:** Users might enter information that somehow isn't acceptable. They may skip required fields, enter numbers that cannot be parsed, or type invalid email addresses, for instance. You want to encourage them to try again. **You want to point out typos before they become a problem, and help puzzled users understand what is asked for**. **Why:** Even better, **some web forms put error messages physically next to the controls where the errors were made. Now the user can see at a glance where the problems were** – no need to hunt down offending field based on just its name – and the instructions for fixing it are right there, easily visible. **How:** Clearly mark all the required fields as required (with astericks), and don't ask for too many required fields in the first place. **When errors do happen, you should show some kind of error message on top of the form**, even if you put detailed messages next to the control. **The top is the first thing people see** (It's also good for visually impaired users – the top of the form is read to them first, so they know immediately that the form has an error.) **Put an attention-getting graphic there**, and **use test that's stronger than the body text: make it red and bold, for instance**. **Now mark the form fields that caused the errors. Put specific messages next to them**, if you can – this will require extra space beside, above, or below the fields – but at least **use color and/or a small graphic to mark the field**. **Users commonly associate red with errors in this context**. **Use it freely, but since so many people are color blind with respect to red, use other cues**, too; language, bold text (not huge), and **graphics**.

Color blind research

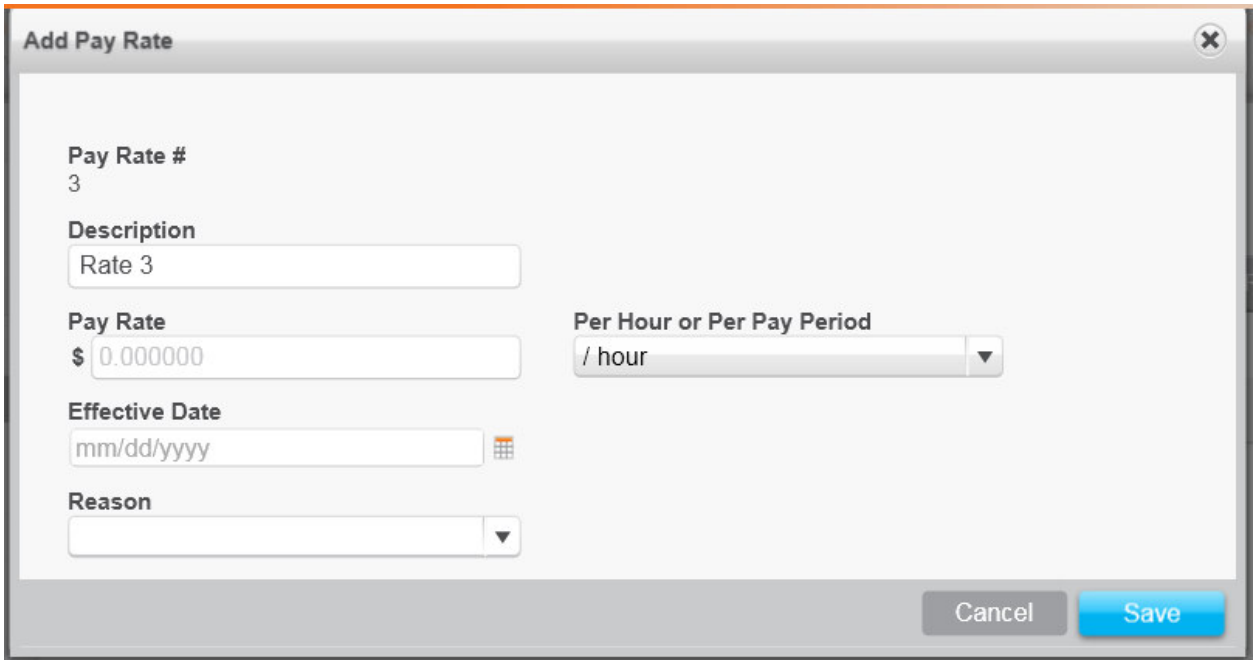
Article Name	Site	Research
Research-Based Guidelines	http://www.usability.gov/articles/newsletter/pubs/112005news.html	1) Avoid the situation where important information is conveyed only in the form of color. 4) Caution when using red. For non-color blind people, **red** is the bright and vivid color. But **for color blinds, it is as dull as blue or dark green**. Especially for protanopes, who cannot detect long wavelength of red light, dark red appears almost as black. Color blind people, however, still feel certain ranges of reds as bright and vivid colors. Instead of pure dark red (RGB=100%, 0%, 0% or #FF.), please use vermilion (yellowish red with shorted wavelength: RGB=100%, 32%, 0% or #FF2000), or light red (mixed with white: RGB=100%, 8%, 8% or #FF1414).
First Principles of Interaction Design by Bruce Tognazzini	http://www.asktog.com/basics/firstPrinciples.html	Color Blindness **Any time you use color to convey information in the interface, you should also use clear, secondary cues to convey the information to those who won't be experiencing any color coding today.** Most people have color displays nowadays, but they are not universal. **In addition, approximately 10% of human males, along with a rare sprinkling of females, have some form of color blindness**. The cones in the eye are the source of color vision. We have cones separately sensitive to red, green, and blue. If the red ones are not functioning that is called protanopia. If the green are not functioning, that is called deuteranopia. Absence of blue, extremely rare, is called tritanopia. Protanopia and deuteranopia are the most popular forms of color blindness, collectively called red/green blindness. (There are, in fact, significant differences in their effects, but those differences have no real effect on interaction design.) While tritanopia is

		far more rare, it nonetheless rules out dependence on yellow-blue differentiation without secondary cues. **Secondary cues can consist of anything from the subtlety of gray scale differentiation to having a different graphic or different text label associated with each color presented.**

How many people in US are color blind?

Article Name	Site	Research
Extrapolation of Prevalence Rate of Color blindness to Countries and Regions	http://www.rightdiagnosis.com/c/ color_blindness/stats-country.htm	Approx 1 in 76 (1.30%) or **just over 4 million people in USA** (total US population 312.8 million in 2012) **are color blind.**

Problem area: Not all required fields marked with asterick

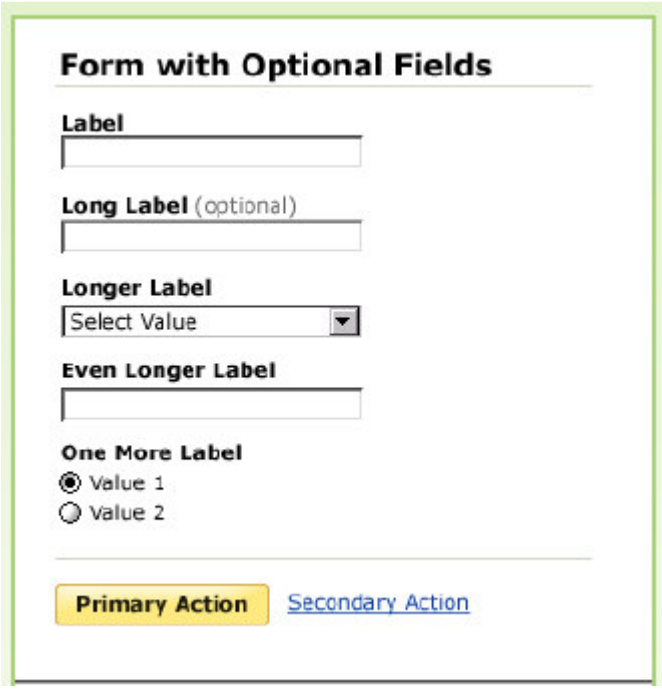


Add Payrate modal window – No required fields marked

Recommendations

Solution – Mark fields either required or optional

If there are few fields on a form, mark all non-required fields as "Optional." If there are a lot of fields on a form, mark all required fields as "Required."



Form with optional field marked

UX research

Information from March usability test findings

- All users had problems with recognizing required fields on screens and modal windows that were not marked with an asterisk.

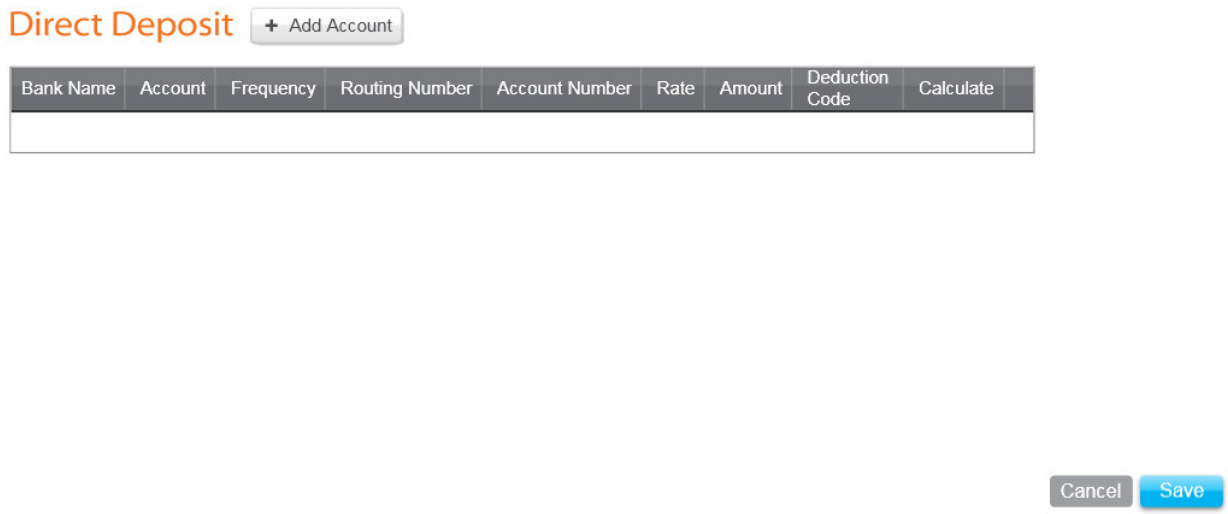
Required form fields

Article Name	Site	Research
Web Forms by Luke Wroblewski	http://static.lukew.com/ webforms_lukew.pdf	Indication of required fields is most useful when • There are lots of fields • But very few are required • Enables users to scan form to see what needs to be filled in Indication of optional fields is most useful when • Very few fields are optional Neither is really useful when • All fields are required

Problem area: Users not seeing light-colored buttons

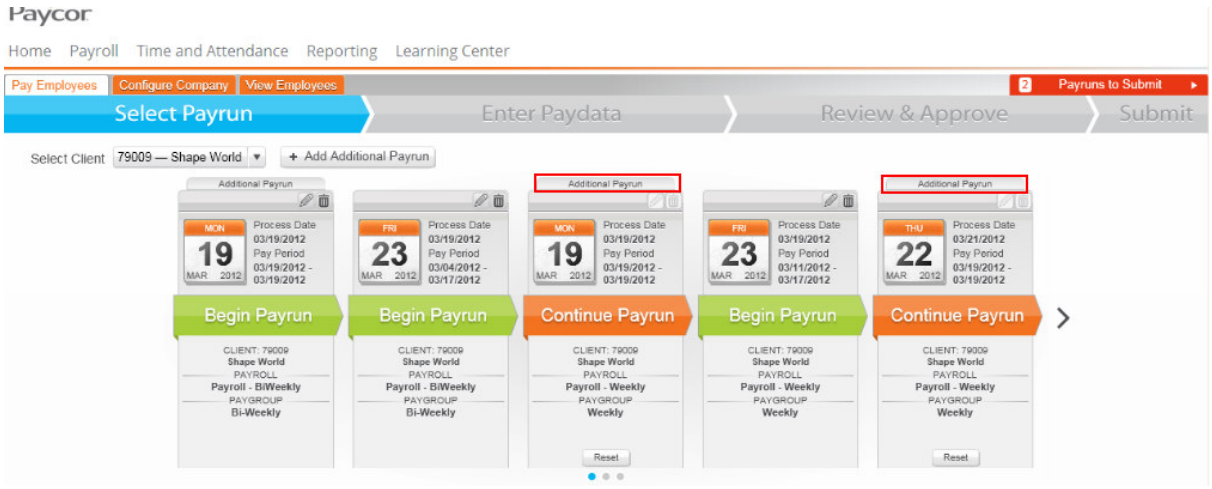
Issues:

- Light-colored buttons are generally accepted as being secondary actions because their color. Even though no information has been added to this screen, the Save and Cancel buttons are still the primary and secondary actions so users are not quickly finding the Add Account button at the top due to this interrupted visual flow. The primary action of this screen should be to add therefore the Add button should be colored to indicate this action to the users. Users also had problems seeing the Add Pay Rate button in the both usability tests.



Direct Deposit screen displaying no information

- Users did not see the Additional Payrun indicators on the Select Payrun screen after they added a new payrun.



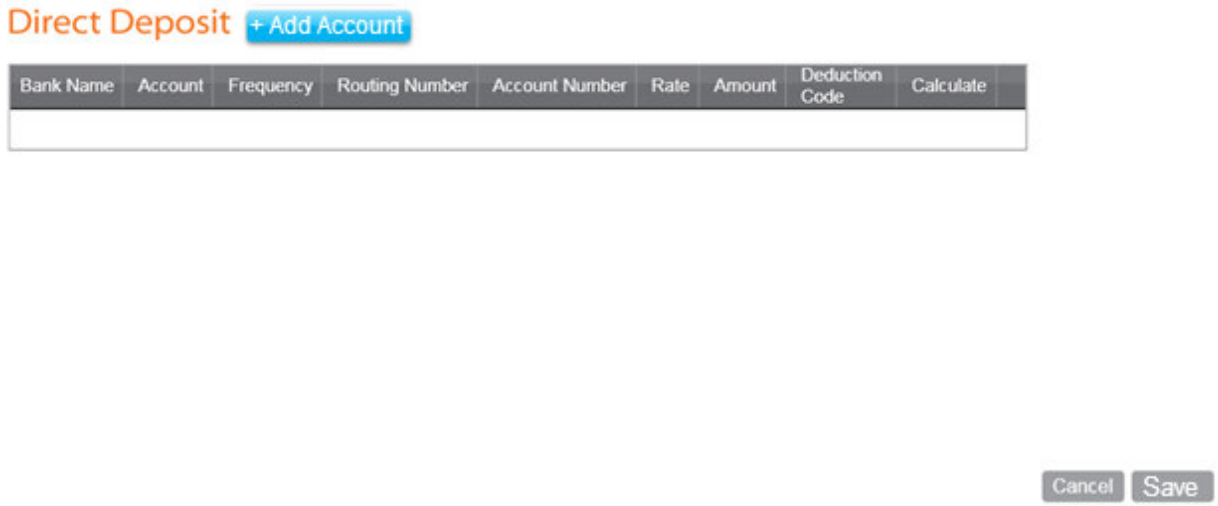
Additional Payrun indicators on the Select Payrun screen

Recommendations

Solutions

Solution: Screens with no information

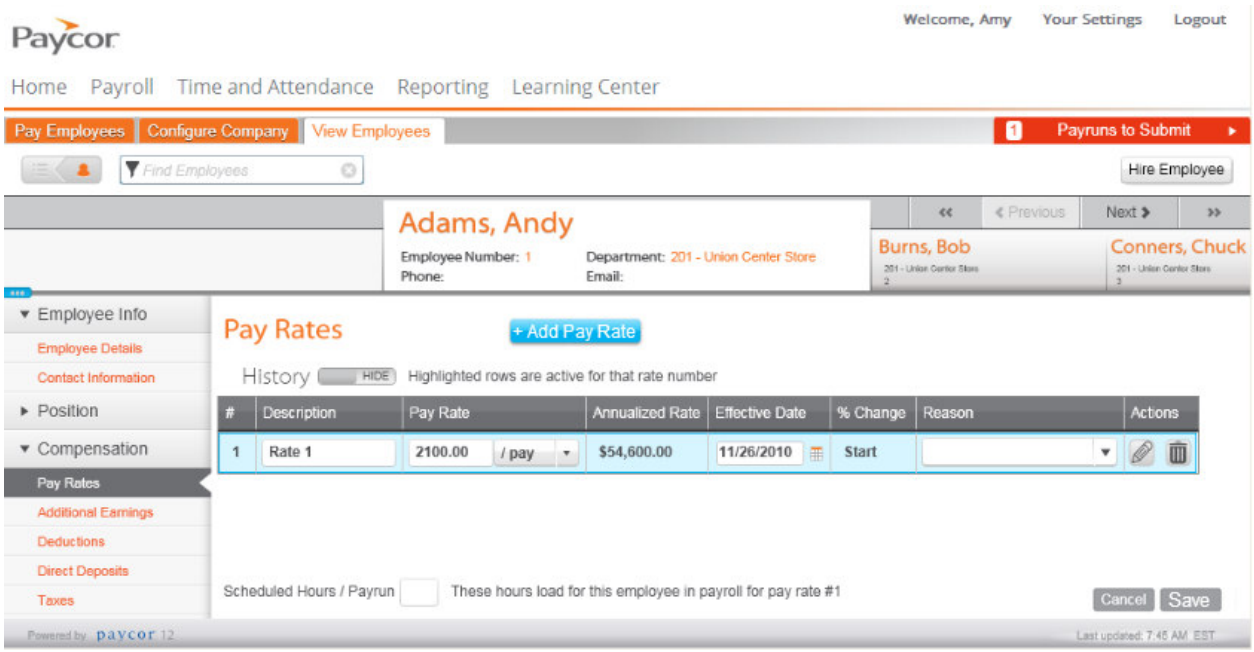
Disable or "gray out" the Save and Cancel buttons so they are not the primary and secondary actions on the screen. Color the Add button to visually indicate to the user that is it the primary action.



Direct Deposit with Add Acount as primary action

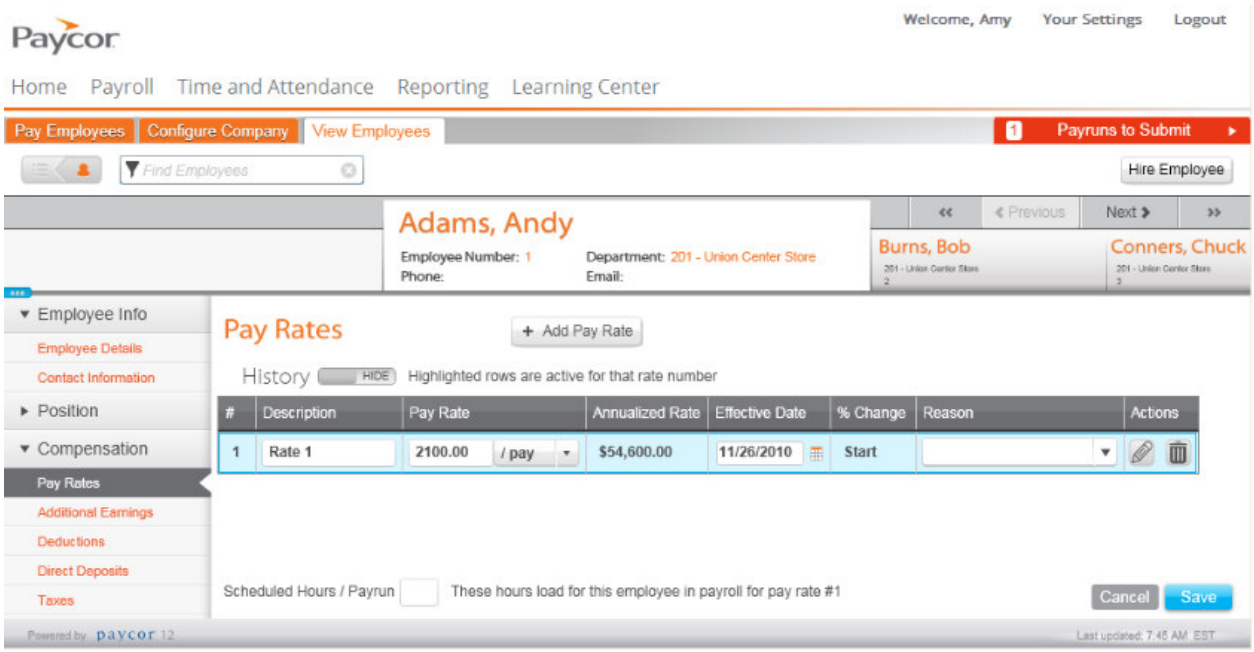
Solution: Screens with information

Until a user clicks into a field indicating they want to change information, make the Save and Cancel buttons appear inactive or "grayed out" on the screen. Color the Add button to visually indicate to the user that it is the primary action.



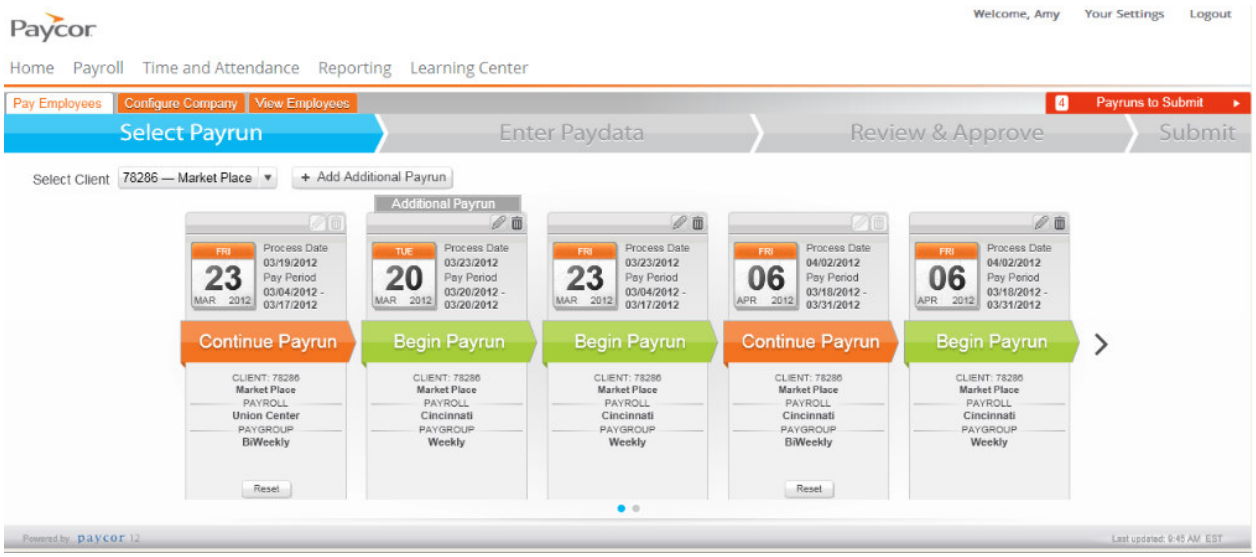
Pay Rates before user clicks into a field to edit – Add is the primary action

When the user does click in a field, they are indicating that they want to edit information so that becomes the primary action. The Add button is then "grayed out."



Pay Rates after user clicks into a field to edit – Edit is the primary action

Solution: Make the Additional Payrun indicator more obvious



Select Payrun screen with an obvious Additional Payrun indicator

UX research

March usability test results, Task 3

	Issue	
2	User did not seem to see Addtional Payrun at top of ticket **(For users 4 – 11, I adjusted the screen size so the Additional Payrun indicator at the top of the screen was more noticable.)**	User 4 – (User clicks Add Additional Payrun button. **How can you tell which one you just added?) "By the date?"(What if they are all the same date? Is there anything else there that would tell you what you just added?) "I can go by payroll since I just added bi-weekly. I would look at those first. Process date 3/20." (Is there anything else?) "It has today's date right there?" (Is there anything else?) "Pay period dates." (User not seeing text at top. Anything else?) "Begin Payrun?" (Anything else? Nothing? You don't see anything?) "Additional Payrun?"** User 6 - (**Can you tell which one (payrun) you just added. User selects correct payrun. How are you selecting that?) "Because it says Process Date.." (Is there anything else on there that is different from the others?) "Just the dates." (Anything at the top?) "It says Additional Payrun."** User 11 - (User clicked Additional Payrun button and fills out screen. Now what would you do?) "Select Payrun." **(Do you see anything different? Can you tell which one you just added?) "Additional Payrun." (There you go.) "And it was not real obvious to me immediately."**

Primary and secondary actions

Article Name	Site	Research
Primary & Secondary Actions in Web Forms, by Luke Wroblewski	http://www.lukew.com/ff/entry.asp?571	"**Secondary actions, on the other hand, tend to be less utilized and most often allow people to retract the data they've entered**. Options like "Cancel", "Reset", or "Go Back" represent **secondary actions that are counter to most people's primary goal of completing the form they started**. **Because secondary actions can have negative consequences, especially when used unintentionally, I've often argued they should be absent from forms**. Imagine filling in a long form online only to hit the "Reset" button and have all your data erased. That said there are situations where secondary actions make sense ("Save for later", "Export", etc.). In these conditions, the best practice I've advocated has been to visually distinguish primary and secondary actions so people have an clear path illuminating their primary goal: completing a form.

Book Name	Research
Designing Interfaces by Jennifer Tidwell, pages 257-258, Prominent "Done" Button	**What: Place the button that finishes a transaction at the end of the visual flow; make it big and well labeled.** **Use when:** Whenever you need to put a button such as Done, Submit, OK, or Continue on your interface, you should use this pattern, More generally, use a visually prominent button for the final step of any transaction. **Why: A well-understood, obvious last step give the user a sense of closure**. There's no doubt that the transaction is done when that button is clicked; don't leave them hanging, wondering whether their work took effect. Making the last step obvious is what this pattern is really about. **How: Create a button** that actually looks like a button or use medium-size button graphics **with bold colors** and well-defined borders. **This will help the button stand out on the page and not get lost among other things**. **Place the button where the user is most likely to find it. Trace the task flow down through the page or form or dialog box, and put the button just beyond the last step. Usually that will be on the bottom and/or right of the page.** I**n any case, make sure the button is near the last text field or control. If it's too far away the user may not find it immediately upon finishing her work, and she may go look for other affordances in her quest for "what to do**

	next."

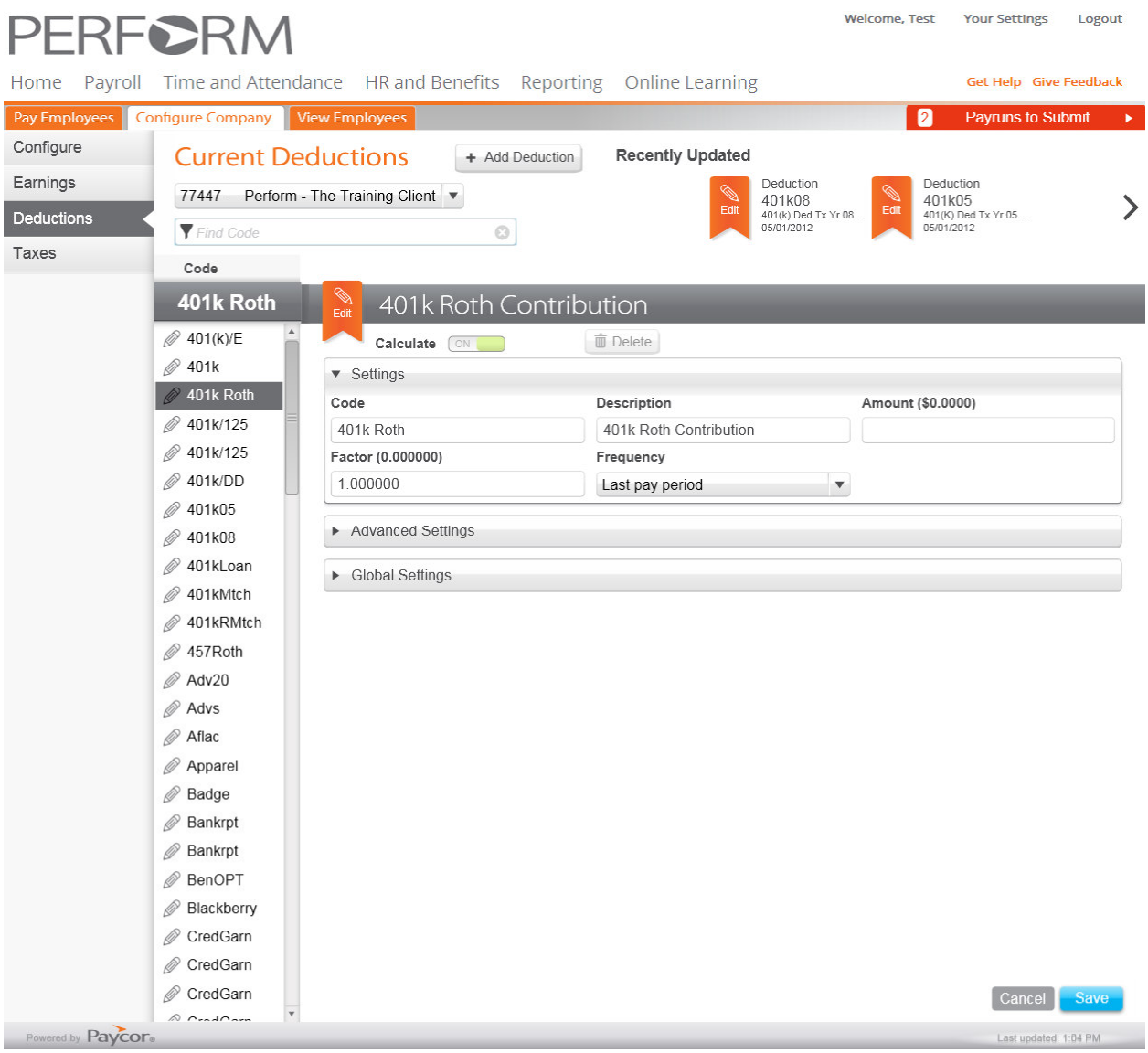
Visual flow

Book Name	Research
Designing Interfaces by Jennifer Tidwell, pages 136-137, Visual Flow	**Visual flow deals with the tracks that readers' eyes ternd to follow as they scan the page.** It's initmately related to visual hierarchy, of course – a well-designed visual hierarchy sets up focal points on the page wherever you need to draw attention to the most important information. As a designer, you **want to be able to control visual flow on a page so that people follow it in approximately the right sequence**. Focal points are the spots your eyes can't resisit going to. You tend to follow them from strongest to weakest, and skillfully designed pages have only a few – too many focal points dilute the importantance of each one. **A good visual hierarchy uses focal points to pull eyes to the right places in the right order**. Likewise, **if you're designing a form, arrange the controls along a continous path and put "I'm finished" buttons (OK, Cancel, Submit, Buy, etc.) at the end of the line.**

Problem area: Confusing layout and no feedback

Issues:

- Due to the screen layout on Configure Company - Deductions, users could not tell if they added a deduction.



Configure Company - Deductions

- No feedback was given if the user successfully performed adding a deduction.

Recommendations

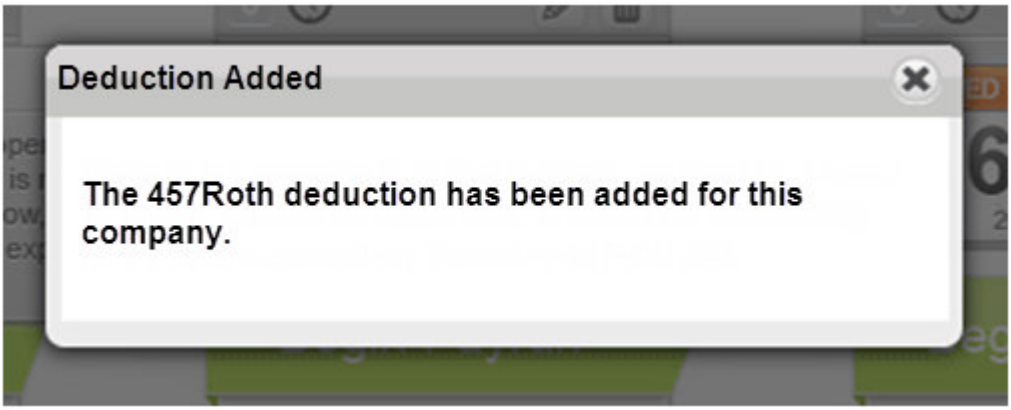
Solutions

Solution for confusing screen layout

Redesign the Configure Company Deduction section to look and function like the Employee Deductions screens, which **all users in the March usability test found easier to use and understand**. For accessibility reasons, this is the preferred solution to this problem.

Solution for lack of feedback

Give users feedback when they do something – some kind of visual indication, either on the screen or in a modal window, that they completed the task.



Feedback example

UX research

Day 1 of Perform training

- Feedback that it should confirm a "save" when adding an Earning - the system was too fast for everyone (they couldn't see the please wait message).

March usability test results, Task 2

- Users prefered the Employee – Deduction flow to Configure Company – Deduction flow saying it was easier to use.
- Users were confused about what the flags meant. Users were confused on which deductions were already added to a company and which fields were required.
- Two users failed this task due to not understanding which codes were already added and which they could add.
- All users were confused as to how this screen worked, including clicking the plus button to add a deduction or if the deduction was actually added or not.

March usability test Task 2: No feedback

Issue	Comments
User not sure if deduction was added	User 11: "**Now I am assuming that if I click Save and I have not entered it properly, it is going to tell me."** (User clicks Save. Do you think it added it? Deduction was added. Can you tell? User is scrolling up and down. What are you thinking?) **"I am thinking that I am not sure if I have added it. I am not sure if it actually – you know. There is no window that came up and said that I did.**"

March usability test Task 2: Not understanding screen layout

Issue	
Not understanding codes already added for company (on the side)	User 2 - "Deductions. Hmm..." (What are you thinking?) **Well I am not sure if any of these** (user is scrolling up and down deductions that are listed.) **are what... I guess I am confused because I am not sure what kind of deduction this should be**. " (Okay – say a dental deduction. A different dental deduction. So what would you do?) "Edit." (INCORRECT) User 6 - (What are these here (under Code and Description heading)) "They are deductions." (**Are those already added? Do you think?) "Probably not because the pens are not highlighted in a certain color." (She clicked on a deduction that has already been added.) "Well that did not mean anything." (What do you mean?) "It did not highlight." (Would you expect that to highlight if it were added to the company?) "I would think so. Because then it tells you at a glance, without doing anything, that you already have something in that field."** User 11 – (How can you tell which ones (deductions) have already been added for the company? Can you tell?) **"I think it would be these that are already up here. I would assume."** (Indicating to the drawer he has open. INCORRECT.)

No feedback

Article Name	Site	Research
The Psychologist's View of UX Design by Susan Weinschenk	http://uxmag.com/articles/the-psychologists-view-of-ux-design?goback=.gde_72842_member_116422312	7. **People Crave Information** • Dopamine is a chemical that makes people seek… food, sex, information**.** Learning is dopaminergic—**we can't help but want more information**. • People will often want more information than they can actually process**.** Having more information makes people feel that they have more choices. Having more choices makes people feel in control. Feeling in control makes people feel they will survive better. • **People need feedback. The computer doesn't need to tell the human that it is loading the file. The human needs to know what is going on**.
Top-10 Application-Design Mistakes by Jakob	http://www.useit.com/alertbox/application-mistakes.html	4. No Feedback One of the most basic guidelines for improving a dialog's usability is to

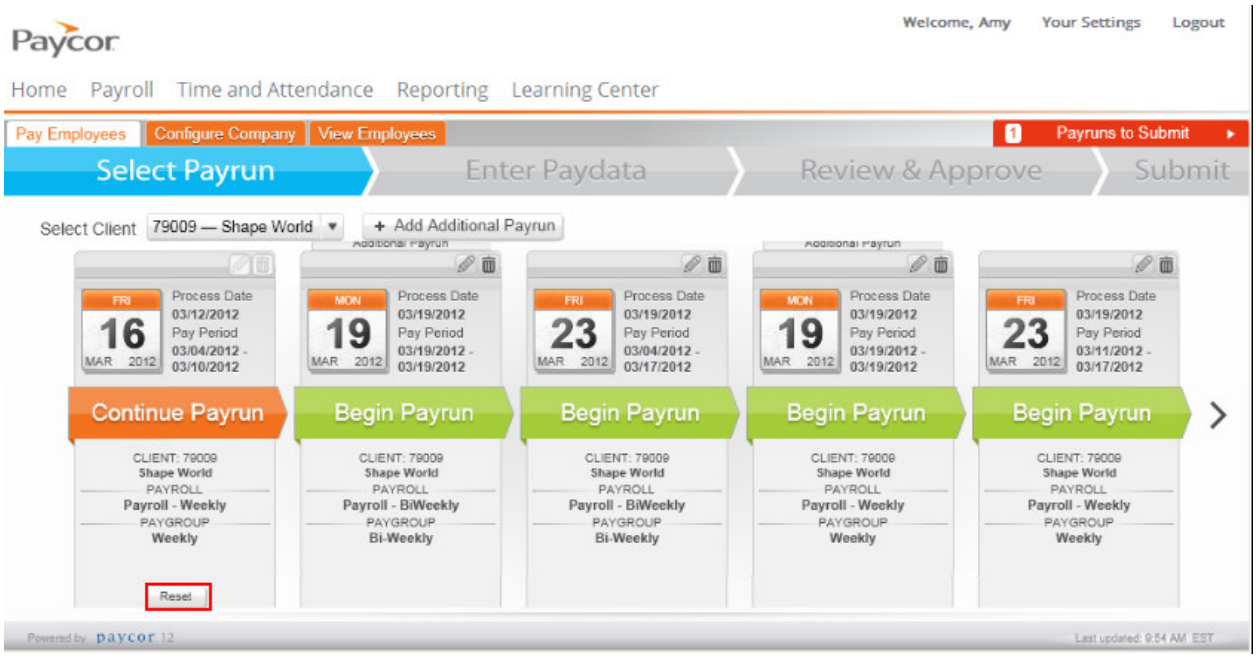
Nielsen		provide feedback: • Show users the system's current state. • **Tell users how their commands have been interpreted.** • **Tell users what's happening.** **Sites that keep quiet, leave users guessing. Often, they guess wrong.**
Mac OS X Human Interface Guidelines	https://developer.apple.com/library/mac/#documentation/UserExperience/Conceptual/AppleHIGuidelines/UEGuidelines/UEGuidelines.html#//apple_ref/doc/uid/TP40002720-SW9	Be Responsive **Responsiveness is how users measure the performance of your software**. Your app might use the best data-processing algorithms and performance-boosting techniques available, but **if it does not instantly respond to the user, it will seem slow**. During app development, pay attention to both the factors that influence the user's perceptions and the actual, measurable performance metrics that your app generates. **Instantly acknowledge the user's commands and input**. **Users expect to receive some type of feedback every time they interact with your app**. For example, buttons highlight when users click them and the pointer changes appropriately as users move it over different controls and areas. Similarly, if a command can't be carried out, users want to know why it can't and what they can do instead. **The quicker you provide feedback for the user's interactions, the more responsive your app appears**. **Provide plenty of informative feedback and communication throughout your app**. Relevant, reliable feedback helps users feel confident that they have enough information to make the right choices.

Problem: Reset button

Recommendations

Solution for reset button

Remove button and enable the Edit and Delete functions.



Reset button on screen

UX research

Reset button

Article Name	Site	Research
Reset and Cancel Buttons by Jakob Nielsen	http://www.useit.com/alertbox/20000416.html	The Reset button hurts users in three ways: • The worst problem about Reset is that users click the button by mistake when they wanted to click Submit. Bang - all your work is gone! • Having two buttons at the bottom of a form clutters up the interface and makes it harder for users to clearly see their next step. **Some small amount of wasted time is spent scanning the useless button** and deciding which of the two buttons is the correct one. • **Even when users do want to eliminate some of the data they have entered into a form, it may slow them down to have a dedicated button for doing so, since the extra button means that users have a choice:** • **Edit the erroneous fields and replace the old text with the**

		new text • **Click Reset and type the new text into nice clean fields** **The extra choice requires extra thinking**, and the time saved by using an optimal interaction technique is often smaller than the time wasted on having to think instead of just moving ahead with a single interaction technique that is always used. It takes at least one second and often two seconds to decide between two possible interaction techniques which is why it is usually better not to offer users a choice. (**A second** may not seem like much, but it **translates into about $100 million in lost productivity per year world-wide**.)
Top-10 Application-Design Mistakes by Jakob Nielsen	http://www.useit.com/alertbox/20000416.html	Bonus Mistake: Reset Button on Web Forms **Making it easy for users to destroy their work in a single click violates one of the most basic usability principles, which is to respect and protect the user's work at almost any cost.** (That's why you need confirmation dialogs for the most destructive actions.)
Primary & Secondary Actions in Web Forms, by Luke Wroblewski	http://www.lukew.com/ff/entry.asp?571	**Because secondary actions can have negative consequences, especially when used unintentionally, I've often argued they should be absent from forms**. Imagine filling in a long form online only to hit the "Reset" button and have all your data erased.

Problem: No Undo button

Recommendations

Solution: Add an Undo and Redo function on all forms.

UX research

Undo button

Day 1 of Perform training

- We should have an undo button in case the user forgot what they had just done.

- Some feedback that the auto save for everything is too much if there is no "undo" – if the client makes a mistake the system still saves the change when moving to another field – the user does not have to hit save.

Article Name	Site	Research
Ultimate guide to table UI patterns by Theresa Neil	http://designingwebinterfaces.com/ultimate-guide-to-table-ui-patterns	Inline Editing Best Practices • Implement tab navigation when you create a table with inline editing. • **Offer undo and redo functionality.**
Mac OS X Human Interface Guidelines	https://developer.apple.com/library/mac/#documentation/UserExperience/Conceptual/AppleHIGuidelines/UEGuidelines/UEGuidelines.html#//apple_ref/doc/uid/TP40002720-SW9	**Create safety nets so that users feel comfortable learning how to use your app**. For example, support Undo as much as possible. Also, consider allowing users to make changes to their content, but defer committing to the changes until a later time. For example, iPhoto allows users to perform all sorts of modifications to a photo without actually changing the photo file until they want to.
First Principles of Interaction Design by Bruce Tognazzini	http://www.asktog.com/basics/firstPrinciples.html	Always allow "Undo." **The unavoidable result of not supporting undo is that you must then support a bunch of dialogs that say the equivalent of, "Are you really, really sure?"** Needless to say, this slows people down. In the absence of such dialogs, people slow down even further. **A study a few years back showed that people in a hazardous environment make no more mistakes than people in a supportive and more visually obvious**

		environment, but they worked a lot slower and a lot more carefully to avoid making errors.
Top-10 Application-Design Mistakes by Jakob Nielsen	http://www.useit.com/alertbox/20000416.html	It is one of the most basic heuristics for interaction design to support user control and freedom by allowing users an "emergency exit" out of any situation they may have entered. **Undo was truly one of the greatest advances in usability.** **The basic rule says to support undo**; it doesn't say how. More specific rules are needed for different types of user interfaces: • In editing systems it is usual to have an Undo command that makes the document revert to the state before the user's most recent changes. **Sometimes, multi-level undo and redo is supported: this can be very useful but confusing**.
The Psychologist's View of UX Design by Susan Weinschenk	http://uxmag.com/articles/the-psychologists-view-of-ux-design?goback=.gde_72842_member_116422312	3. People Make Mistakes • Assume people will make mistakes. **Anticipate** what they will be and try to prevent them. • If the results of an error are severe then **use a confirmation** before acting on the user's action. • Make it easy to "**undo**." • Preventing errors from occurring is

		always better than helping people correct them once they occur. **The best error message is no message at all**. • If a task is error-prone, **break it up into smaller chunks**. • If the user makes and error and you can **correct it**, then do so and show what you did.