

World Usability Day 2007

Designing Usable Healthcare Web Applications

Jason Withrow

University of Michigan Health System

jwithrow@med.umich.edu

Overview

- Healthcare User Needs Analysis
- Designing Applications vs. Websites
- Role-Based Access
- Search Before Navigation
- Never Enough Screen Real Estate
- Additional Visual Design Issues
- The Myth of the Paperless Workplace
- Improving Data Entry
- Supporting the Occasional User

Healthcare User Needs Analysis

- One of the great advantages of working in healthcare is access to the people who are (or will be) using the application
- This opens up a lot of possibilities:
 - User representation on the project team
 - Easy access to subject matter experts
 - Less of a need for personas, since you are directly interacting with real users
 - Paper prototyping, usability testing, etc.

Shadowing / Contextual Inquiry

- There is also the opportunity to shadow the users, learning their work processes through direct observation
- The value of this is substantial; directly observed actions are always more reliable than any self-report data

User Considerations

- Work environment is fast-paced
 - Task efficiency is critical
- Users are mobile
 - They are not sitting at the computer all day
- Users share a computer
 - For example, multiple clerks may be using the same workstation

User Considerations

- Computers are locked down
 - Installing software is limited or prevented
 - Changing screen resolution may be blocked
- Patient safety is essential
 - Error prevention, such as warnings before committing significant changes, are important
- Patient privacy is essential
 - HIPAA compliance

Designing Applications vs. Sites

- Web applications focus much more on task analysis and interface design for efficiency
- Websites are different and tend to involve deliverables such as site diagrams and content inventories

Role-Based Access

- One of the first usability considerations is the primary audiences for the application
- These become the roles and each role has different levels of access
 - Some roles have limited access; other roles have access to everything

Role-Based Access

- When presenting navigation, only show the options available for their role
 - Do not show unavailable options as disabled
 - Users typically do not need to know (and have little interest in knowing) what else the application can do
 - More importantly, users expect that if they see the option that somehow they can enable it

Examples of Roles

- Diet Order Entry
 - Unit Clerk
 - Diet Clerk
 - Dietitian Tech
 - Registered Dietitian
 - Administrator
 - Quality Assurance
- Interpreter Scheduling
 - Administrator
 - Staff Interpreter
 - Community Interpreter
- Wound Referral
 - Team Member
 - All Other Caregivers

Role-Based Navigation

Registered Dietitians:

[Lists](#) [Consults](#)

Unit Clerks:

[Unit Clerk Inbox](#) [Basic Diet List](#) [CBORD Reports](#)

Dietitian Techs:

[Diet Tech Inbox](#) [Lists](#) [Labels & Talleys](#) [Patient Education](#)

Choosing Between Roles

- There will be cases where one user belongs to multiple roles
- Rather than trying to integrate the roles ask them to select a role after logging in

Log in as:

- Quality Assurance
- Diet Clerk
- Registered Dietitian
- Unit Clerk
- Administrator
- Dietitian Tech

Login

Switching Between Roles

- For those individuals you can also allow them to switch roles once logged in, which saves them the step of logging out and then logging in again

Logged in as: adminuser1 ([Logout](#)) [Switch to Staff](#)

Selecting the Best 'Landing Page'

- Each role has different priorities and needs, so there is not always a common page that all users will see upon logging in
 - Clinical personnel may be automatically redirected to a 'My Worklist' screen
 - Administrative personnel may see a screen for running reports
- This redirect saves them time and energy

Search Before Navigation

- In many healthcare applications the most common task is searching for a patient, so having a globally available search function is essential
 - Typically the search is by patient name or registration number, although location may also be included (especially if the focus is on inpatients)
- Navigating through the thousands of patients would be too time-intensive

Avoid Overloading Search

- Too many choices slows down users and allows strange combinations:

Patient Search	Reg # <input type="text"/>	Last <input type="text"/>	First <input type="text"/>	Search
	SS # <input type="text"/>		Sex <input checked="" type="radio"/> Any <input type="radio"/> F <input type="radio"/> M	Clear

Use a Flexible 'Smart' Search

- To save users time, have the search engine auto-detect the type of query:



Reg # or Last Name:

- Sometimes they need to make a choice; give them a good default value:



Search: Patient Caregiver

After Search Comes Navigation

- Once a patient has been located the focus shifts to navigation and ideally the screen most commonly used will load automatically
- At this time the navigation bar also expands to show patient-specific navigation options
 - It is helpful during the planning process to designate which pages are patient-specific and which are not patient-specific

Patient-Specific Navigation

No patient selected:

[Unit Clerk Inbox](#) [Basic Diet List](#) [CBORD Reports](#)

Patient selected:

[Unit Clerk Inbox](#) [Basic Diet List](#) [CBORD Reports](#) [Special Requests](#) [Calendar View](#) [Order History](#)

Never Enough Screen Real Estate

- Healthcare applications are often data-intensive, in terms of both entering data and displaying data
- Over time the amount of data displayed only increases; it is rare that we ever take away columns from an existing table in the user interface when an application is enhanced

Dealing with Limited Space

- The first thing to remove is any left-hand or right-hand navigation

Archive Records											
Delete Volume											
Volume Loader											
View Medical Record Summary											
Print a Barcode											
View Entire Medical Record History											
Outstanding Requests											
Update	Request Type	Requestor	Volume	Content	Created At	Required Date/Time	Status	Reason	Updated By	Comments	
History											
Location	Volume	Content	Created At	Service	From Location	Reason	Date Time	Required Date Time	Status	Updated By	Comments
DOCUMENT & RECORD MGT : NCAC : RETURNING TO FILE	Inpatient 1	General	NCAC		UNKNOWN OTHER UNIT : MI : OFFICE OF CLINICAL AFFAIRS	Emergency Visit	08/08/2007 21:42	09/13/2006 10:18	Delivered	mrtadm1	
UNKNOWN OTHER UNIT : MI : OFFICE OF CLINICAL AFFAIRS	Inpatient 1	General	NCAC		UNKNOWN OTHER UNIT : TC : TEST	Emergency Visit	09/13/2006 10:18	09/13/2006 10:18	Delivered	mrtuser1	
UNKNOWN OTHER UNIT : TC : TEST	Inpatient 1	General	NCAC		RELEASE OF INFORMATION : NCAC : RELEASE OF INFORMATION	Record Review/Filing	07/31/2006 11:02	07/31/2006 11:02	Delivered	mrtadm1	

Dealing with Limited Space

- Horizontal navigation frees up needed space

Print Barcode :: Delete Volume :: Archive Records :: Administrative :: Help
Logged in as: **mrtadm1** (Logout)

[Patient Search](#) [Request Volume](#) [Return Volume](#) [Sign in to D&RM](#) [Change Location](#) [View Summary](#) [View History](#)

Request Volume

Patient Profile

Reg #: [REDACTED] Name: [REDACTED] DOB: [REDACTED] Sex: M

Select Volumes

Select	Volume	Description/Service	Status	Admit/Discharge	Location	Phone	Updated
<input type="checkbox"/>	Outpatient	Psychiatry	Created		OTO AUDIOLOGY CLINIC (OTOAUD-TC) : [REDACTED]	(734) 936 - 5392	
<input type="checkbox"/>	Outpatient	General	Created		OTO AUDIOLOGY CLINIC (OTOAUD-TC) : [REDACTED]	(734) 936 - 5392	

Additional Visual Design Issues

- Color
- Text Sizing
- Size of Targets

Color

- Various issues arise with the use of color
- The first is to provide sufficient contrast between foreground and background colors
 - Users may need to operate in low-light conditions (e.g., using a computer in the room where the patient is sleeping)
 - Users may be operating on limited sleep, which can impact their visual acuity

Color and Visual Scanning

- Use of color to promote visual scanning
 - Muted colors make boldface information easier to detect and are easier on the eyes
 - Avoid using too many colors and especially the use of highly saturated colors that make reading the screen difficult
 - Watch for vibration concerns with warmer and cooler colors overlapping or being used in close proximity

Color Meaning Varies by Culture

- The meaning of colors also differs across cultures, so choose carefully
 - Blue has the fewest issues cross-culturally, so it is a safe choice (the tie-in with UM colors is a nice benefit)

Color-Coding Information

- The final consideration is color-coding information so that it can be quickly picked up (at the perceptual level) rather than via the slower process of reading it
 - For example, to denote statuses or warnings
- For accessibility reasons be sure to also convey the information in some other way than color
 - For example, also outputting the status as text

World Usability Day 2007

Yellow is <24 Hrs; Red is <3 Hrs

Russian

<i>Patient</i>	<i>Location</i>	<i>Date</i>	<i>Time</i>	<i>Duration</i>	<i>Status</i>
Test Patient 1	INTER-UH	8/17/07	10:00 AM	30 min.	Pending
Test Patient 12	INTER-UH	8/17/07	11:00 AM	45 min.	Pending
Test Patient 12		8/17/07	11:00 AM	30 min.	Unmatched
Test Patient 12		8/17/07	2:00 PM	45 min.	Unmatched
Test Patient 4	INTER-UH	8/17/07	3:00 PM	120 min.	In Progress
Test Patient 5	INTER-UH	8/17/07	4:00 PM	60 min.	Pending

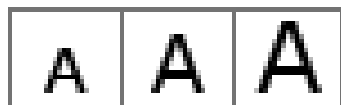
Hide Requests

Spanish

<i>Patient</i>	<i>Location</i>	<i>Date</i>	<i>Time</i>	<i>Duration</i>	<i>Status</i>
Test Patient 9	INTER-UH	8/16/07	3:00 PM	30 min.	Pending

Text Sizing

- Some users will have ‘locked down’ computers that cannot be modified to any great extent
- This can include the screen resolution
- To improve readability of text, support resizing via the browser controls and consider additional controls as well:



Size of Targets

- Make sure that buttons and widgets are large enough to be easily acquired
 - Keep in mind Fitts's law and increase sizing to reduce target acquisition time
 - Small targets close together also increase the odds of clicking the wrong one
- Associate field names with their respective form element via <label>, so that clicking the text puts focus in the indicated form element

Myth of the Paperless Workplace

- You might think that a web application reduces paper use
 - The reality is that many users print reports at the start of the day (or the start of their shift) and carry around the report all day
- In some cases, there is more paper use once the application is built (compared to when it was on a mainframe or totally paper-based)

Myth of the Paperless Workplace

- Sometimes users need the information in an area where there is limited access to computers
 - An interpreter printing his/her schedule before driving out to various clinics for appointments
 - A caregiver seeing patients around the hospital
- Some print them as “insurance” against a system crash (often the system prior to the web application was unreliable)

Visual Design of Reports

- There is a strong focus on printing reports
- Present the information in a format that promotes visual scanning yet is efficient (to reduce the number of pages printed)
 - Use of negative space but not overdoing it
 - Use of boldface for pop-out
 - Visual hierarchy to show relationships

Print-Friendly Output: (X)HTML

- Have a print-specific stylesheet for all pages that removes non-essential information (such as navigation) from the print-outs, switches fonts to serifs, etc.

Print-Friendly Output: PDF

- You may need to go with PDF format if there is a danger of patient information breaking across pages
 - CSS allows page breaks but it is difficult to tell when a patient's data will break across pages
 - A generated PDF can ensure that the patient's data stays on the same page, which is critical if the report gets out of sequence or pages get separated

Improving Data Entry

- Some healthcare web applications require extensive data entry by users
- Comparing a desktop application to a web application in this area, the web application has fewer options and presents greater challenges

Data Entry Concerns: Accesskeys

- Accesskeys are restricted in terms of the key combinations possible
- No accesskey standard exists, making it difficult for users to remember from one application to the next which ones exist
- Accesskeys in the web application will overwrite built-in browser accesskeys, leading to user frustration as the browser no longer responds in the expected fashion

Data Entry Concerns: Devices

- To increase usability as well as accessibility, consider input devices beyond the mouse
- Make sure that tab order is logical and that all controls can be tabbed to, for the users that prefer or require keyboards for input
- For one application the use of a bar code scanner greatly improved efficiency; no need for a keyboard or a mouse

Data Entry Concerns: Auto-Saving

- Some users also expect web applications to behave like desktop applications, in the sense that a desktop application does not necessarily require explicitly saving work
- Thanks to AJAX it is possible for web applications to auto-save periodically, which can prevent a lot of user frustration
- AJAX can also speed up task completion by auto-completing values as they are typed

Speeding Up the Selection Process

- There are also some successes; clicking on a row causes its checkbox to check/uncheck

Logged in as: **test_ca_user1** ([Logout](#))

Search: Patient Caregiver

[Inpatient Worklist](#) [Outpatient Worklist](#) [My Charts](#)

Inpatient Worklist (5 Unanalyzed Charts / Oldest is 7 Days Ago)

Patient	Reg #	Date ↑	Service	Assigned To	Reassign
[REDACTED]	01010101	10/1/2007 - 10/1/2007	MMV	[REDACTED]	<input type="checkbox"/>
[REDACTED]	00000000	10/2/2007 - 10/3/2007	LIS		<input type="checkbox"/>
[REDACTED]	00001111	10/1/2007 - 10/3/2007	PSY	[REDACTED]	<input type="checkbox"/>
[REDACTED]	12345678	10/2/2007 - 10/7/2007	VIS	[REDACTED]	<input type="checkbox"/>
[REDACTED]	11111111	10/3/2007 - 10/8/2007	AMB	[REDACTED]	<input type="checkbox"/>

Assign Checked Items to:

Supporting the Occasional User

- While many web applications will have an established user base that logs in every day, there will be some that need to support the occasional user
- Depending on the user's task a tutorial could be helpful or the process could be designed as a wizard to walk them through it in short steps

Building the Wizard

- Provide an overview at the top of each screen showing the user's progress
- Each step should only have one to three choices/fields, with supporting instructions provided as necessary
- Wizards are also helpful if there are branching pathways, as those can be easily accommodated

Wizard Example: Consult Request

Consult Request

1. Reg # ----> 2. Request Type ----> 3. Consult Reason ----> 4. Referral Source

1. Reg #:

Proceed to Request Type -->

In Conclusion

- Designing usable healthcare web applications is definitely possible
- Keep in mind the user's context and needs
- Never assume that you are done; there is always room for improvement and further iteration