Introduction to Embedded help
and
# Examples applied to dotLRN

Created by Ludivine Martin

Tue Mar 18 2008

ABOUT THE HELP SECTION

Help is the last resort

- People don't use the help section, or they use it at a minimum
The users have to realize they need assistance, open the help, locate the relevant information, and return their focus to the application. For many, this process feels like too much effort for the perceived benefit. As a result, users, and ironically especially novice users, do not use the help systems

- Most applications continue to provide unified, external online help, online manual or any kind of tutorial but people don't use them.

- Studies show that
   * Users avoid online help because of the time spent in using it, in that there is not a return on that investment of time.
   * Traditional format of user assistance is ineffective.
   * Switching back and forth between the application software and the related help system is an important source of difficulties, especially for novice users. (The effects of such 'attention- splitting' situations have been thoroughly investigated in the scope of the cognitive load theory.)
   * Users are in general dissatisfied with online help systems

References: [1] [2] [3] [4] [5]
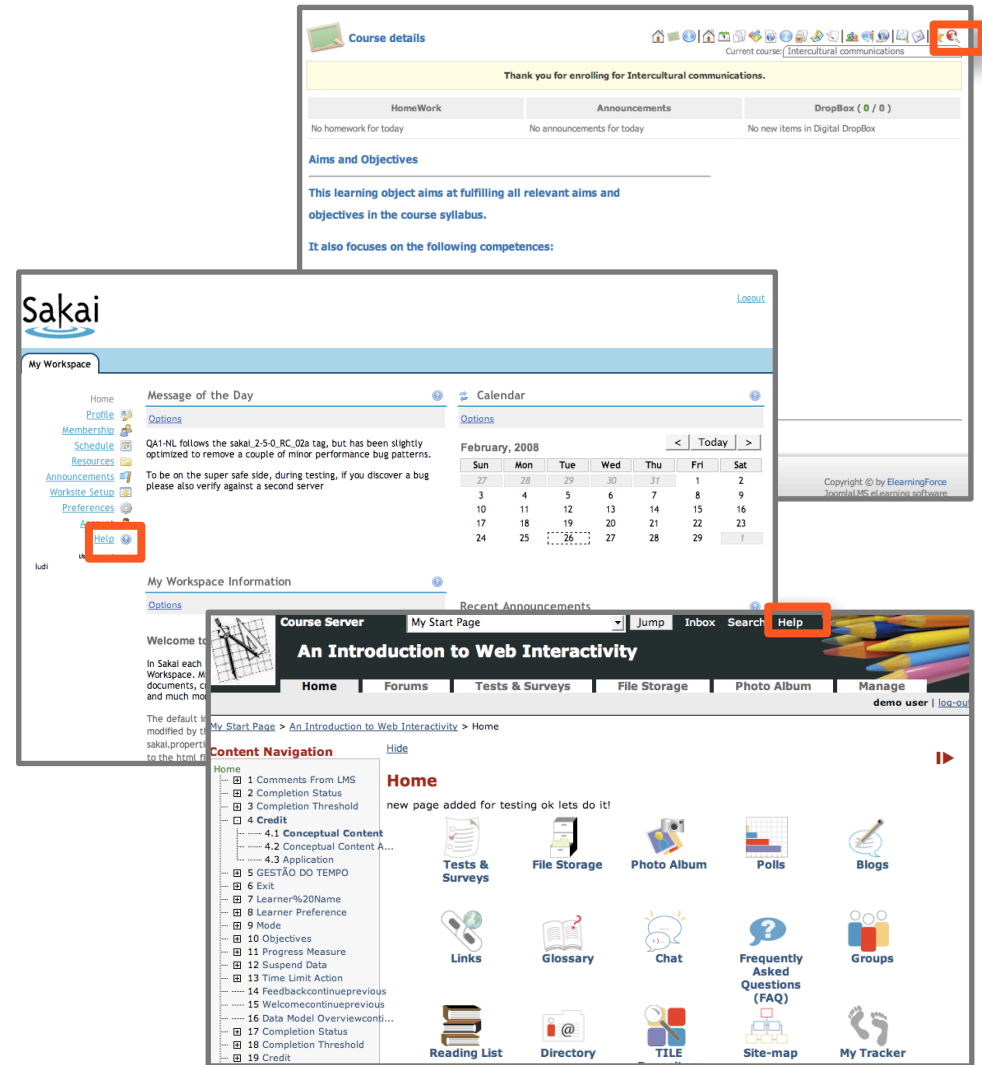
Exemples of LMS with links to a help section

▶ **People don't use the help section**

CONTEXTUAL HELP

Contextual help works better than regular help

- Definition

Contextual help provides specific information about the condition or mode the application is in at the time help is sought.

Contextual help, or context-sensitive help, as opposed to general online help or online manuals, doesn't need to be accessible for reading as a whole. Each topic is supposed to describe extensively one state, situation, or feature of the software.

- contextual help is more effective than non contextual help as regards successful task execution

- But still, contextual help often happens in a new window and although it diminishes the attention splitting, it doesn´t remove it.

References: [2] [6] [7]
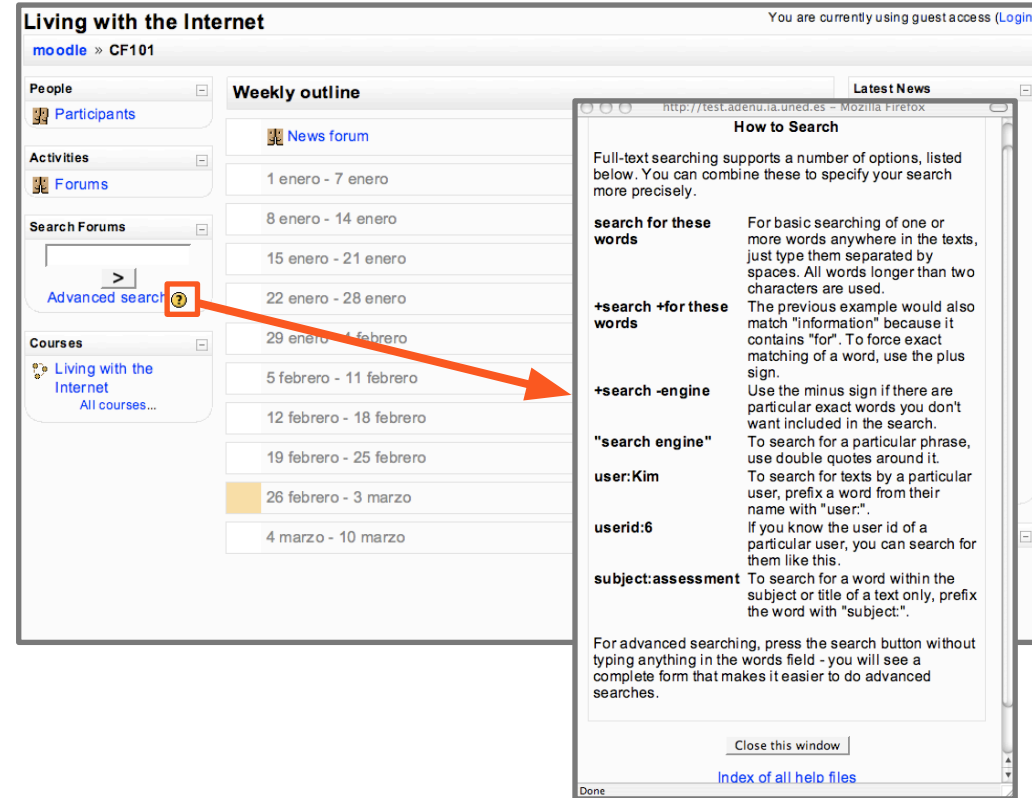
Exemples of contextual help in Moodle

> **Contextual help doesn´t always avoid attention splitting**

CONTEXTUAL HELP

Contextual help and domain information

- Studies show that:

   * it is important to present certain key information up-front, before users start on a task.
   * domain information (key concepts, definitions, rules, conventions, etc.) is a critical a element of help for web applications, and is potentially more useful to users than procedural information. But user don't tend to look for it within a help section or anywhere else. This critical information should be provided to the user without him having to look for it.
   * embedded help is an effective technique to provide domain information

- In recent years, more sophisticated formats of user assistance have evolved. Help files are no longer the "total" help solution but merely one of many types of user assistance.

References: [4] [8]
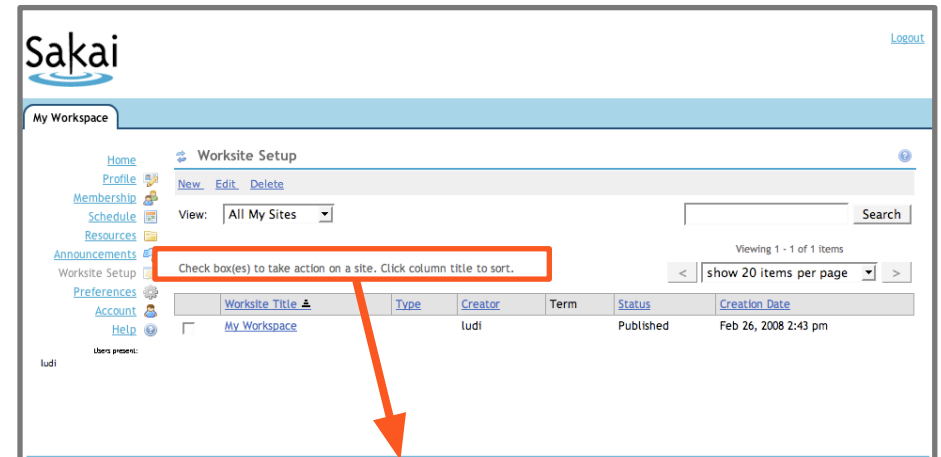
Exemples of domain information in Sakai

▶ **Contextual help is not so efficient if outside the task flow**
▶ **Contextual help provided within the task flow = embedded help**

EMBEDDED HELP

What is it?

- Definition

Embedded help, or "nested help", or "embedded user assistance" provides the user with assistive instruction aside the interaction. It is easier for the user to relate the instruction with how it applies to the web application's functionalities than a disconnected FAQ or help section.
Information is moving out of the separate online help systems into the interfaces.

- How it works

Embedded user assistance
   * integrates information into the interface.
   * is like using the interface itself as a help.
   * can often be the best context sensitive help (i.e.,
replacement for popups).
The closer you get the information to the user, the better.

References: [3] [9] [10]

▶ **Embedded help integrates information directly within the interface**

EMBEDDED HELP

Why using embedded help?

- **It gives users the information they need to perform a task correctly the first time**

- **It gives answers to the user questions at the point of use**
(or at the point of confusion)

- **It is more efficient**

New usability study and concluded that instead of revising the existing online help system models, writers must improve the user interface and then work to provide an extremely context-sensitive, embedded user assistance system.

- **It doesn´t have to be permanent**

That is, we need to implement some tools that stay around long enough to help users learn what they need to know, but then disappear once their purpose has been served.

- **Web user demographic has changed**

The Web is now used by all ranges of technical and most of all non-technical users.
Many developers think that a help section will suffice for teaching users how to operate any application, but this is not the case. The proper way is to bring the help section to them and in a format that can easily be understood.

- **User is more likely to use the help section after**

After using embedded assistance, users are much more likely to click on help links and spend more time using the external help system.

References: [1] [3] [9] [11]

▶ **Embedded help better serves a wide range of users**
▶ **Embedded help provides large benefit for user with practically no effort**
▶ **Users are more likely to refer to the help section after that**

WARNING

Hide it

- For users who are familiar with a specific web application, embedded Help could be counter-productive because it tends to clutter the UI with information that is unnecessary for experienced users.

- We have to be particularly careful regarding this last topic, considering that typical user of dotLRN will use the interface on long time-scales (to follow a course over a semester for exemple), and therefore become experienced

- A solution is to offer an option to hide the help boxes. It could be managed within the User preference and settings with an option such as "Show/Hide all help boxes"
It could also be set to disappear automatically after the 10th login or so.

References: [8] [10]

▶ **Beware of not cluttering the interface too much**
▶ **Always offer a option to hide the extra information**

IMPLEMENTATION

Offer tips

- Tips appear when needed and can
   * explain why information is needed
   * explain or show how to enter the information
   * let the users instantly know what is expected of them
   * helps ensure that the data is entered correctly
So users do not have to resort to help documents, use the
Contact Us link, or leave the website in frustration.

- Instructive text, for example, can be used in form ele-
ments and interfaces in general to guide the user within the
context of her current task without being intrusive.

Tips
- can show up as roll over or on click
- are short and concise
- offer suggestions and examples (e.g. within form fields)
- are specially useful in forms

Tips can be
- persistent: they are always there: for information needed by
the user to complete the task correctly and efficiently
- visible only on roll over or click: for information not required
but helping to provide a good experience to the user
A same page can combine persistent inline tips with the
temporary ones.

Link can be provided within the tips to the help section.
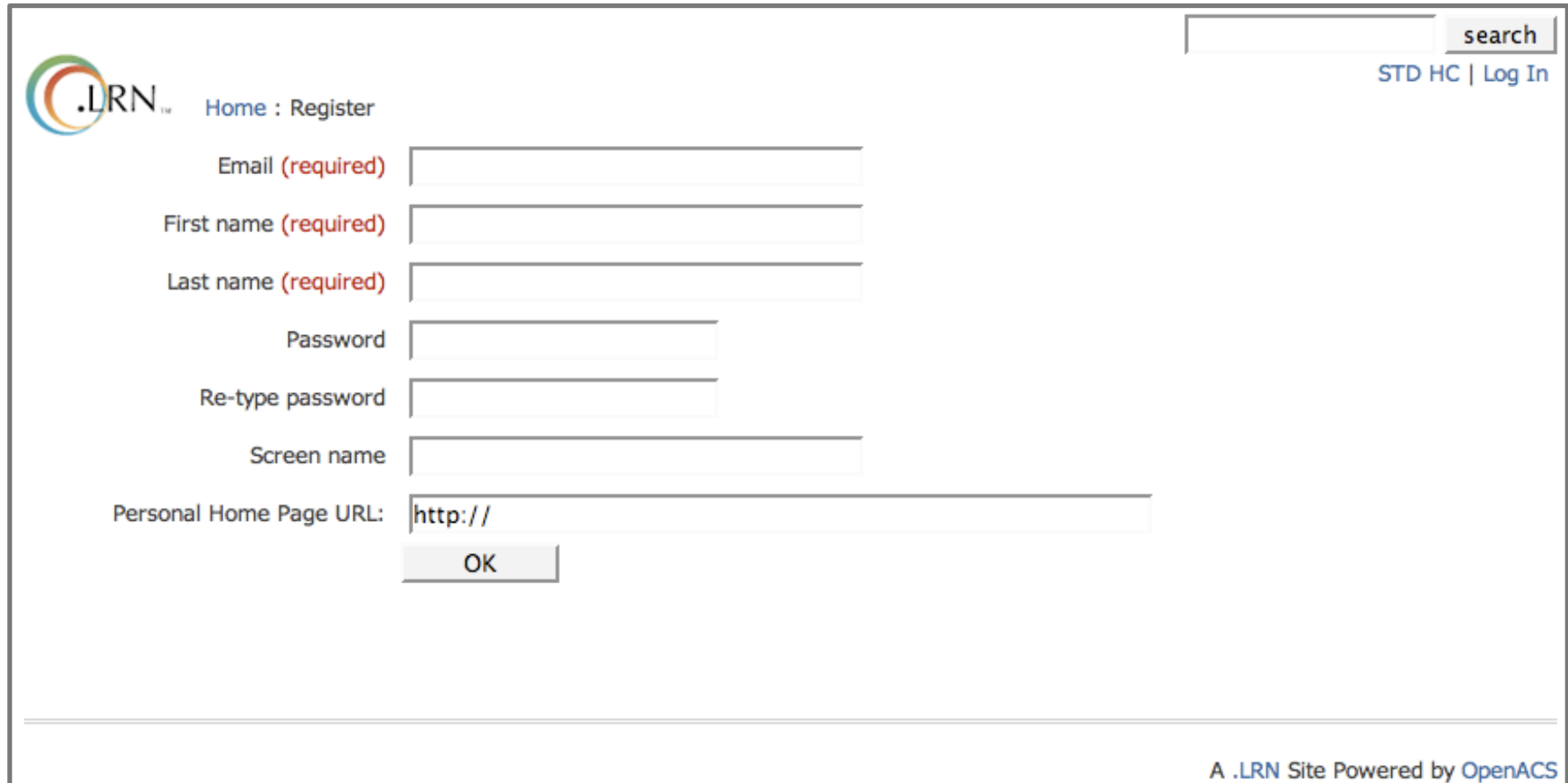
References: [11] [12]

IMPLEMENTATION

Offer tips

Registration - BEFORE

IMPLEMENTATION

Offer tips

Registration - AFTER : persistent inline tips

IMPLEMENTATION

Offer tips

Registration - AFTER : Bubble when roll over

IMPLEMENTATION

Offer tips

Registration - AFTER : Tip appears on click

The tip appears only when the user clicks the field and disappear when click on another field .
Only one tip at a time on the screen

▶ **To make your password more secure**
- Use letters and numbers
- Use special characters (e.g.,@)
- Mix lower and uppercase

IMPLEMENTATION

Offer tips

File upload - BEFORE

IMPLEMENTATION

Offer tips

File upload - AFTER : Show example

IMPLEMENTATION

Welcome screens / descriptive help blocks

- Welcome screens for the main landing page, also called descriptive help blocks for other pages, should help the user situate himself within the interface and guide him through the next steps.

- For example, we can designate an area at the top of an application screen to display information about what to do on that page and what to do next.

- Applied to dotLRN, it could become Welcome "help boxes" that appear by default and that user can choose to hide

- Based on current dotLRN structure, it could apply to the following spaces
    > Welcome to your space (see example)
    > Welcome to the community "Name of the community" space
    > Welcome to the class "Name of the class" space
    > Welcome to the control panel (see example)

References: [11]

IMPLEMENTATION

Welcome screens / descriptive help blocks

My Space - BEFORE

IMPLEMENTATION

Welcome screens / descriptive help blocks

My Space - AFTER

IMPLEMENTATION

Welcome screens / descriptive help blocks

Control Panel - BEFORE

IMPLEMENTATION

Welcome screens / descriptive help blocks

Control Panel - AFTER

IMPLEMENTATION

Fill the blank slate with something useful

- Whenever some screens are blank or offer few content or options, always anticipate the user next tasks and offer links and guiding cues accordingly.

- For users just getting to know a Web application, a blank slate can be a barrier in the learning process. Instead of knowing exactly what to do first, users can become stalled when faced with the empty canvas.
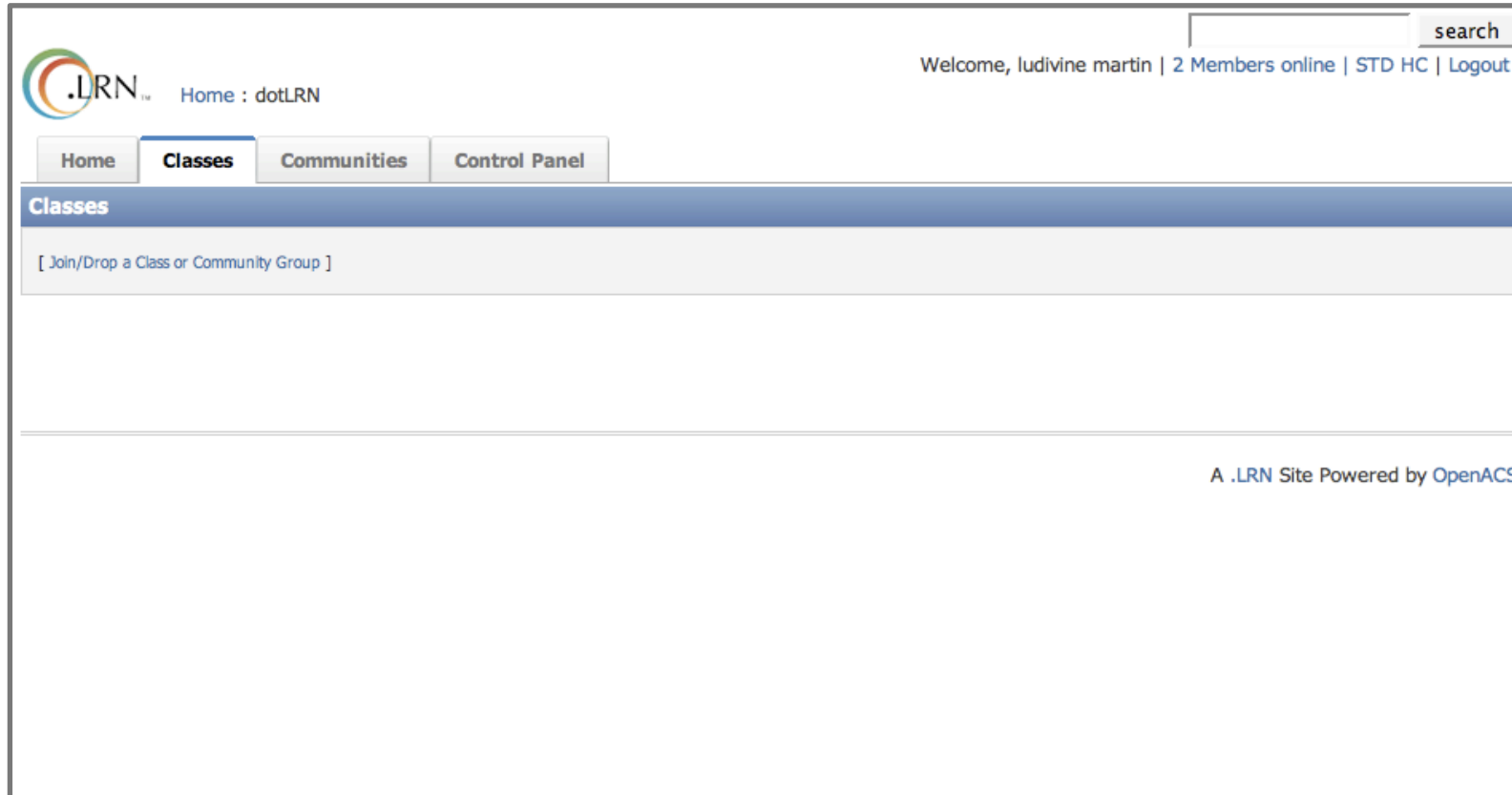In addition to helping users get up to speed, it can be a great way to compel users to jump in and start using the application, which can be the hardest thing to do otherwise.

References: [11]

IMPLEMENTATION

Fill the blank slate with something useful

BEFORE

IMPLEMENTATION

Fill the blank slate with something useful

AFTER

Provide account information on current status : No current class

Provide link to next logical action : Browse classes

Provide suggestion to learn more about the topic, offer a guided tour: Link to a tutorial. The screenshot presents a strong visual cue

NOTE

Embedded help is great

- Previous examples show the range and the potential efficiency of embedded help and instructive design

- Designer and content producers should consider and focus on embedded help when designing and creating content.

But a help section is always needed

- Studies show current failures of approaches claiming that "transparent user interfaces" eliminate the need for online help.

- The user should never need the TOC (table of content) or Index. However, this is not an excuse for not writing a good TOC or Index.

- An external help system is needed to provide in-depth conceptual information and procedures.

- Furthermore, and ironically, advanced users tend to look more into the help systems than beginners, and their needs should be addressed as well.

References: [2] [10] [11]

▶ **Embedded help should be prioritized, but a help section is always needed**

References

[1] Divide and Conquer: Providing Web-based User Assistance at the Point of Use by Scott DeLoach (2004)
http://www.winwriters.com/articles/embedded/index.html

[2] Online help for the general public: specific design issues and recommendations by A. Capobianco, N. Carbonell (2003)
http://www.loria.fr/~carbo/UAISJ_03.pdf

[3] Design Checklists for Online Help by Michelle Corbin (2004)
http://www.writersua.com/articles/checklist/index.html

[4]Help and Usability by Joan Kalk (2006)
hrrp://www.chi-sa.org.za/CHI-SA2001/Help&Usability.PDF

[5] Contextual Help and its Effects on Human Working Memory Load by Ellen Reitmayr (2006)
http://www.kdedevelopers.org/node/1780

[6] Definition of: context sensitive help (2008)
http://www.pcmag.com/encyclopedia_term/0,2542,t=context+sensitive+help&i=40282,00.asp

[7] Context sensitive definition (2008)
http://en.wikipedia.org/wiki/Context-sensitive_help

[8] A Usability Test of Web-based User Assistance by Matthew Ellison (2004)
http://www.winwriters.com/usability_test_analysis.htm

[9] Evolve Your User Interface To Educate Your Users by Dave Shepard (2008)
http://www.smashingmagazine.com/2008/01/16/evolve-your-user-interface-to-educate-your-users/

[10] Context Sensitive Help Concepts by Don Lammers (2000)
http://www.smountain.com/resource/ContextHelp.pdf

[11] Designing the Obvious: A Common Sense Approach to Web Application Design by Robert Hoekman Jr. (2006)
http://www.rhjr.net/dto

[12] Help is the last resort by Gordon R. Meyer (2007)
http://www.g2meyer.com/usablehelp/lastreso.html