Guidelines, principles and Theories

- **1. Guidelines.** Low-level focused advice about good practices and cautions against dangers.
- The practical guidelines prescribe cures for design problems, caution against dangers, and provide helpful reminders based on accumulated wisdom.
- 2. **Principles.** Middle-level strategies or rules to analyze and compare design alternatives.
- 3. **Theories.** High-level widely applicable frameworks to draw on during design and evaluation as well as to support communication and teaching.
- .

Guidelines

- A guidelines document has a shared language and promotes consistency among multiple designers in terminology usage, appearance, and action sequences. It records best practices derived from practical experience or empirical studies, with appropriate examples and counterexamples.

- The creation of a guidelines document engages the design community in lively discussions about input and output formats, action sequences, terminology, and hardware devices

Navigating the Interface: Guidelines

- sample guidelines (**National Cancer Institute, 2006) by US Government** to promote the design of informative webpages

- **Standardize task sequences.** Allow users to perform tasks in the same sequence and manner across similar conditions.
- **Ensure that links are descriptive.** When using links, the link text should accurately describe the link's destination.
- **Use unique and descriptive headings.** Use headings that are distinct from one another and conceptually related to the content they describe.
- **Use radio buttons for mutually exclusive choices.** Provide a radio button control when users need to choose one response from a list of mutually exclusive options.
- **Develop pages that will print properly.** If users are likely to print one or more pages, develop pages with widths that print properly.
- **Use thumbnail images to preview larger images.** When viewing full-size images is not critical, first provide a thumbnail of the image.

Guidelines to promote accessibility for users with disabilities

- **Text alternatives**. Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols, or simpler language.
- **Time-based media**. Provide alternatives for time-based media (e.g., movies or animations). Synchronize equivalent alternatives (such as captions or auditory descriptions of the visual track) with the presentation.
- **Distinguishable.** Make it easier for users to see and hear content, including separating foreground from background. Color is not used as the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.
- **Predictable**. Make Web pages appear and operate in predictable ways

The goal of these guidelines is to have webpage designers use features that permit users with disabilities to employ screen readers or other special technologies to give them access to webpage content.

Organising the Display: Guidelines

Goals for data display

- 1. **Consistency of data display.** During the design process, the terminology, abbreviations, formats, colors, capitalization, and so on should all be standardized and controlled by use of a dictionary of these items.
- 2. **Efficient information assimilation by the user.** The format should be familiar to the operator and should be related to the tasks required to be performed with the data. This objective is served by rules for neat columns of data, left justification for alphanumeric data, right justification of integers, lining up of decimal points, proper spacing, use of comprehensible labels, and appropriate measurement units and numbers of decimal digits.
- 3. **Minimal memory load on the user.** Users should not be required to remember information from one screen for use on another screen. Tasks should be arranged such that completion occurs with few actions, minimizing the chance of forgetting to perform a step. Labels and common formats should be provided for novice or intermittent users.

4. **Compatibility of data display with data entry.** The format of displayed information should be linked clearly to the format of the data entry. Where possible and appropriate, the output fields should also act as editable input fields.

- 5. **Flexibility for user control of data display.** Users should be able to get the information from the display in the form most convenient for the task on which they are working. For example, the order of columns and sorting of rows should be easily changeable by the users.

Getting the users attention: Guidelines

- **Intensity.** Use two levels only, with limited use of high intensity to draw attention.
- **Marking**. Underline the item, enclose it in a box, point to it with an arrow, or use an indicator such as an asterisk, bullet, dash, plussign, or X.
- **Size**. Use up to four sizes, with larger sizes attracting more attention.
- **Choice of fonts.** Use up to three fonts.
- **Blinking.** Use blinking displays (2–4 Hz) or blinking color changes with great care and in limited areas, as it is distracting and can trigger seizures.
- **Color.** Use up to four standard colors, with additional color reserved for occasional use.
- **Audio.** Use soft tones for regular positive feedback and harsh sounds for rare emergency conditions.
- Animation should be used to provide meaningful information, such as for a progress indicator or to show movement of files.

Facilitating data entry: Guidelines

- 1. **Consistency of data-entry transactions.** Similar sequences of actions speed learning.
- 2. **Minimal input actions by user**. Fewer input actions mean greater operator productivity and usually—fewer chances for error. Making a choice by a single mouse selection or finger press, is preferred over typing in a lengthy string of characters. Selecting from a list of choices eliminates the need for memorization, structures the decision-making task, and eliminates the possibility of typographic errors.
- A second aspect of this guideline is that redundant data entry should be avoided. It is annoying for users to enter the same information in two locations, such as entering the billing and shipping addresses when they are the same. Duplicate entry is perceived as a waste of effort and an opportunity for error.

- 3. **Minimal memory load on users**. When doing data entry, users should not be required to remember lengthy lists of codes.
- 4. **Compatibility of data entry with data display**. The format of data-entry information should be linked closely to the format of displayed information, such asdashes in telephone numbers.
- 5. **Flexibility for user control of data entry**. Experienced users prefer to enter information in a sequence that they can control, such as selecting the color first or size first, when clothes shopping

- Principles

- Know your user/Determining Users skills : First Principle

- Learning about the users is a simple idea but a difficult and often undervalued goal. .
- Successful designers are aware that people learn, think, and solve problems in different ways.
- Some users prefer to deal with tables rather than graphs, with words instead of numbers, or with rigid structures rather than open-ended forms.

- All design should begin with an understanding of the intended users.

- The process of getting to know the users is never-ending because there is so much to know and because the users keep changing.
- However, every step toward understanding the users and recognizing them as individuals with outlooks different from the designer's own is likely to be a step closer to a successful design.

- All design should begin with an understanding of the intended users This includes,
  - population profiles that reflect their age, gender, physical and cognitive abilities, education, cultural or ethnic backgrounds, training, motivation, goals, and personality.
  - Other variables that characterize user personas include location (for example, urban versus rural), economic profile, disabilities, and attitudes toward using technology.
  - Users with poor reading skills, limited education, and low motivation require special attention.

Design goals for novice or first-time, knowledgeable intermittent, and expert frequent users

- Novice or First Time Users

- Novice users—for example, bank customers making their first cellphone check deposit— are assumed to know little of the task or interface concepts.

- first-time users are often professionals who know the task concepts well but have shallow knowledge of the interface concepts (for example, a business traveler using a new rental car's navigation system).
- Both groups of users may have learning-inhibiting anxiety about using computers.

Design Goals

1. Overcoming user uncertainties via instructions or dialog boxes.
2. Restricting vocabulary to a small number of familiar, consistently used concept terms is essential.
3. The number of actions should also be small so that novice and first-time users can carry out simple tasks successfully, which reduces anxiety, builds confidence, and provides positive reinforcement
4. Informative feedback about the accomplishment of each task is helpful, and constructive
5. Specific error messages should be provided when users make mistakes.
6. Carefully designed video demonstrations and online tutorials may be effective

- **Knowledgeable intermittent users.**

- Many people are knowledgeable but intermittent users of a variety of systems (for example, business travelers filing for travel reimbursements). They have stable task concepts and broad knowledge of interface concepts, but they may have difficulty retaining the structure of menus or the location of features.
- The burden on their memories is lightened by orderly structure in the menus, consistent terminology, and interfaces that emphasize recognition rather than recall.
- Consistent sequences of actions, meaningful messages, and guides to frequent patterns of usage help knowledgeable intermittent users to rediscover how to perform their tasks properly.

Expert frequent users.

- Expert “power” users are thoroughly familiar with the task and interface concepts and seek to get their work done quickly.
- They demand rapid response times, brief and non-distracting feedback, and the shortcuts to carry out actions with just a few clicks or gestures.
- M*acro of sequence of three or four actions are preffered* or other abbreviated form to reduce the number of steps are also preffered. Strings of commands, shortcuts through menus, abbreviations, and other accelerators are requirements.

multi-layer/ level-structured or spiral approach to learning

- When multiple usage classes must be accommodated in one system, the basic strategy is to permit a *multi-layer* (sometimes called *level-structured* or *spiral)* approach to learning.
- Novices can be taught a minimal subset of objects and actions with which to get started. They are most likely to make correct choices when they have only a few options and are protected from making mistakes . This is called *training-wheels* interface.
- After gaining confidence from hands-on experience, these users can choose to progress to ever-greater levels of task concepts and the accompanying interface concepts.

- For example, novice users of a cell phone can quickly learn to make/receive calls first, then to use the menus, and later to store numbers for frequent callees.
- Their progress is governed by the task domain, rather than by an alphabetical list of commands that are difficult to relate to the tasks.
- The multi-layer approach must be carried out in the design of not only the software, but also the user manuals, help screens, error messages, and tutorials
- Multi-layer designs is the most promising approach to promoting universal usability.

- Another component of accommodating different usage classes is to permit user control of **the density of informative feedback that the system provides.**
- Novices want more informative feedback to confirm their actions, whereas frequent users want less distracting feedback. Similarly frequent users like displays to be more densely packed than do novices.
- The pace of interaction may be varied from slow for novices to fast for frequent use users.

Identify the tasks

- After carefully drawing the user profile, the developers must identify the tasks to be carried out.
- Too often the task analysis is done informally or implicitly .but successful strategies usually involve long hours of observing and interviewing users.

- This helps designers to understand task frequencies and sequences and make the tough decisions about what tasks to support.
- Some implementers prefer to include all possible actions in the hope that some users will find them helpful, but this can cause unfortunate clutter.

- High-level task actions can be decomposed into multiple middle-level task actions, which can be further refined into atomic actions that users execute with a single command, menu selection, and so on.
- Choosing the most appropriate set of atomic actions is a difficult task.
- If the atomic actions are too small, the users will become frustrated by the large number of actions necessary to accomplish a higher-level task.

- If the atomic actions are too large and elaborate, the users will need many such actions with special options, or they will not be able to get exactly what they want from the system

- The relative task frequencies are important in shaping a set of commands or a menu tree.
- Relative frequency of use is one of the bases for making architectural design decisions.
    - Frequent tasks should be simple and quick to carry out, even at the expense of lengthening some infrequent tasks.
    - For example, in a word processor Frequent actions might be performed by special keys, such as the four cursor arrows, Insert, and Delete.
- Less frequent actions might be performed by a single letter plus the ctrl key, or by a selection from a pull-down menu-examples include underscore, bold, or save.
- Infrequent actions or complex actions might require going through a sequence of menu selections or form fillins-for example, to change the printing format or to revise network-protocol parameters.

- A matrix of users and tasks can help designers sort out these issues .
- In each box, the designer can put a check mark to indicate that this user carries out this task.
- A more precise analysis would include frequencies instead of just simple check marks.
- Such user-needs assessment clarifies what tasks are essential for the design and which ones could be left out to preserve system simplicity and ease of learning.

FREQUENCY OFTASK BY JOBTITLE

Hypothetical frequency-af-use data for a medical clinic information system. Answering queries from appointments personnel about individual patients is the highest frequency task.

	TASK				
Job title	Query by Patient	Update Data	Query across Patients	Add Relations	Evaluate System
Nurse	0.14	*0.11*			
Physician	0.06	0.04			
Supervisor	0.01	0.01	0.04		
Appointment personnel	0.26				
Medical-record maintainer	0.07	0.04	0.04	0.01	
Clinical researcher			0.08		
Database programmer			0.02	0.02	0.05

	TASK				
Job title	Query by Patient	Update Data	Query across Patients	Add Relations	Evaluate System
Nurse	0.14	*0.11*			
Physician	0.06	0.04			
Supervisor	0.01	0.01	0.04		
Appointment personnel	0.26				
Medical-record maintainer	0.07	0.04	0.04	0.01	
Clinical researcher			0.08		
Database programmer			0.02	0.02	0.05

Choose an interaction style

- When the task analysis is complete and the task objects and actions have been identified, the next step is to choose the interaction style. The designer can choose from these primary interaction styles:
- Direct manipulation,
- Menu selection,
- form fillin,
- Command language, and
- Natural language

Direct Manipulation

- **The system is portrayed as an extension of the real world**.
    - It is assumed that a person is already familiar with the objects and actions in his or her environment of interest. The system simply replicates them and portrays them on a different medium, the screen.
    - A person has the power to access and modify these objects, including windows.
    - A person is allowed to work in a familiar environment and in a familiar way, focusing on the data, not the application and tools.

- **Objects and actions are continuously visible**. Like one's desktop, objects are continuously visible. Reminders of actions to be performed are also obvious, where labeled buttons replace complex syntax and command names.
- Cursor action and motion occurs in physically obvious and intuitively natural ways.

- This concept as described as *virtual reality*, a representation of reality that can be manipulated.
- It is also called as WYSIWYG (what you see is what you get).

- **Actions are rapid and incremental with visible display of results**. The results of actions are immediately displayed visually on the screen in their new and current form.
- Auditory feedback may also be provided.
- The impact of a previous action is quickly seen, and the evolution of tasks is continuous and effortless.
- **Incremental actions are easily reversible**. actions, if discovered to be incorrect or not desired, can be easily undone

Menu Selection

- In menu-selection systems, users read a list of items, select the one most appropriate to their task, and observe the effect.
- If the terminology and meaning of the items are understandable and distinct, users can accomplish their tasks with little learning or memorization and just a few actions.
- The greatest benefit is that there is a clear structure to decision making, since all possible choices are presented at one time.
- This interaction style is appropriate for novice and intermittent users and can be appealing to frequent users if the display and selection mechanisms are rapid.
- For designers, menu-selection systems require careful task analysis to ensure that all functions are supported conveniently and that terminology is chosen carefully and used consistently.

Form Fill In

- The form fill-in style is very useful for collecting information.
- Screen contains a series of controls or fields into which the user either types information or selects an option, or options, from a listing of choices.
- Screen fill-in forms are derived from their antecedents, paper forms.
- An advantage of a form is its familiarity. If it is designed well, a form will aid the user in understanding its purpose and allow fast and easy entry of information.
- A poorly designed screen form can be inefficient and aggravating to complete.
- . Since knowledge of the keyboard, labels, and permissible fields is required, some training may be necessary.
- This interaction style is most appropriate for knowledgeable intermittent users or frequent users.

Command language

- For frequent users, command languages provide a strong feeling of being in control.
- Users learn the syntax and can often express complex possibilities rapidly, without having to read distracting prompts.
- Error rates are typically high, training is necessary, and retention may be poor.
- Error messages and online assistance are hard to provide because of the diversity of possibilities and the complexity of mapping from tasks to interface concepts and syntax.
- Command languages and lengthier query or programming languages are the domain of expert frequent users
- Powerful advantages include easy history keeping and simple macro creation.

Natural language

- Requires computers to respond properly to arbitrary natural-language sentences or phrases.
- Natural-language interaction usually provides little context for issuing the next command,
- .

- EIGHT GOLDEN RULES OF INTERFACE DESIGN

RULE 1- *Strive for consistency*

- . There are many forms of consistency.
- Consistent sequences of actions should be required in similar situations;
- Identical terminology should be used in prompts, menus, and help screens; and consistent color, layout, capitalization, fonts, and so on should be employed throughout.

Rule 2-Cater to Universal usability

- . Recognize the needs of diverse users and design for *plasticity,* facilitating transformation of content.
- Novice-expert differences, age ranges, disabilities, and technology diversity each enrich the spectrum of requirements that guides design.
- Adding features for novices, such as explanations, and features for experts, such as shortcuts and faster pacing, enrich the interface design and improve perceived system quality.

Rule 3- offer informative Feedback

- For frequent and minor actions, the response can be modest, whereas for infrequent and major actions, the response should be more substantial.
- Visual presentation of the objects of interest provides a convenient environment for showing changes explicitly

Rule 4- design Dialogs to Closure

- Sequences of actions should be organized into groups with a beginning, middle, and end.
- Informative feedback at the completion of a group of actions gives operators the satisfaction of accomplishment, a sense of relief, the signal to drop contingency plans from their minds, and a signal to prepare for the next group of actions.
- For example, e-commerce web sites move users from selecting products to the checkout, ending with a clear confirmation page that completes the transaction.

Rule 5- Prevent errors

- As much as possible, design the system such that users cannot make serious errors; for example, grayout menu items that are not appropriate and do not allow alphabetic characters in numeric entry fields.
- If a user makes an error, the interface should detect the error and offer simple, constructive, and specific instructions for recovery.
- For example, users should not have to retype an entire name-address form if they enter an invalid zip code, but rather should be guided to repair only the faulty part. Erroneous actions should leave the system state unchanged, or the interface should give instructions about restoring the state.

Rule 6. Permit easy reversal of actions

- As much as possible actions should be reversible. This feature relieves anxiety, since the user knows that errors can be undone, thus encouraging exploration of unfamiliar options.
- The units of reversibility may be a single action, a data-entry task, or a complete group of actions, such as entry of a name and address block.

Rule 7 Support Internal Locus of Control

- Experienced operators strongly desire the sense that they are in charge of the interface and that the interface responds to their actions.
- Surprising interface actions, tedious sequences of data entries, inability to obtain or difficulty in obtaining necessary information, and inability to produce the action desired all build anxiety and dissatisfaction.

Rule 8 Reduce short term memory Load

- The limitation of human information processingmin short-term memory (the rule of thumb is that humans can remember"seven plus or minus two chunks" of information) requires that displays be kept simple, multiple-page displays be consolidated, window-motion frequency be reduced, and sufficient training time be allotted for codes, mnemonics, and sequences of actions.
- Where appropriate, online access to command-syntax forms, abbreviations, codes, and other information should be provided.

Conclusion

- The guidelines and principles described are devoted to simplifying the users' tasks. Users can then avoid routine, tedious, and error-prone tasks and can concentrate on making critical decisions, coping with unexpected situations, and planning future actions