Course: Human Computer Interface (HCI)

Week 10: Interaction Design Styles

Lecturer: Martha Gichuki

Lecture Learning Outcomes

At the end of this lecture, the learner will be able to:

(i) Define basic Interaction design style concepts
(ii) Describe the typical interaction design styles available for HCI designers
(iii) Describe the industry accepted interface design guidelines and standards

INTERACTION STYLES

Introduction

The concept of Interaction Styles refers to all the ways the user can communicate or otherwise interact with the computer system. Recall from lecture 2 that computers are electronic devices that process data into information and that people use the information for decision making. This means that people and computers have to interact using various computer applications (programs). These programs follow different styles of the interaction and sometimes the same operations follow different styles. Some of the interaction styles available for HCI designers to use include: -

i. *Direct manipulation* - a style that is good for tasks such as drawing and resizing objects [1]

[1] Human-computer interaction (3rd ed.), Dix, A., Finlay, J., Abowd, G., & Russell, B. New Jersey: Prentice Hall. ISBN-10: 0130461091, ISBN-13: 978-0130461094, (2004). Pg. 25, 32

ii. *Conversation* - ideal for children and computer phobic users

iii. *Instructions* - ideal for tasks that are repetitive

iv. *Hybrid conceptual models* - model mixtures used where the same task is carried out differently.

Types of interaction design styles

- ✓ Command language
- ✓ Form filling
- ✓ Menu selection
- ✓ Direct graphical manipulation

A. Command language (or command entry)

Command language is the earliest form of interaction style and is still being used, mainly on Linux, Unix and Microsoft Disk Operating System (MS DOS). Command prompts are used by expert users who type in commands and possibly some parameters that will affect the way the command is executed.

Examples of such commands include “ls”, which stands for 'list' and “al” which displays a detailed list of files.

Command language places a considerable cognitive burden on the user in that the interaction style relies on recall as opposed to recognition memory. Commands as well as their many parameterized options have to be learned by heart and the user is given no help in this task of retrieving command names from memory[2].

Advantages of Command Language

- ✓ Flexible and interesting especially if the user knows how to use the commands.
- ✓ Appeals to expert users.
- ✓ Supports creation of “user-defined scripts”
- ✓ Suitable for interacting with networked computers even with low bandwidth.

[2] Human-computer interaction (3rd ed.), Dix, A., Finlay, J., Abowd, G., & Russell, B. New Jersey: Prentice Hall. ISBN-10: 0130461091, ISBN-13: 978-0130461094, (2004). Pg 200

Drawbacks of Command Language

- Requires user retention of commands in memory which, puts heavy memory loads on users with poor retention.
- Learnability of commands is very poor.
- High error rates
- Error messages and assistance are hard to provide because of the diversity of possibilities plus the complexity of mapping from tasks to interface concepts and syntax
- Unsuitable for novice users.

INDUSTRY-ACCEPTED INTERFACE DESIGN STYLES GUIDELINES AND STANDARDS

The guidelines for *good command line interface* are based on three main goals:

1. Offering maximum flexibility

- ✓ Conduct task analysis to determine the necessary commands
- ✓ Provide a way to combine and execute sets of commands.

2. Facilitating command remembering

- ✓ Use meaningful, descriptive names.
- ✓ Follow "de facto" standards.
- ✓ Use options for small modifications in command's behaviour.
- ✓ If abbreviations are necessary, make them consistent when possible.
- ✓ Use consistent format of the command line.
- ✓ Provide on-line help

3. Facilitating error correction.

- ✓ Provide a way to edit and replay last command.
- ✓ Give feedback on both successful and unsuccessful commands

B. Form fill-ins

The form filling interaction style (also called "*fill in the blanks*") is suitable for novice users. Initially, form filling interfaces were fully form-based, with the screen designed as a form in which data could be entered in the pre-defined form fields. Currently, most of the software programs mix forms with other interaction styles. The ***tab-key*** helps users move between the fields and ***enter*** to ***submit*** the form entries. Thus, there was originally no need for a pointing device such as a mouse and the separation of data in

fields allowed for validation of the input. Form filling interfaces are useful for routine, clerical work or for tasks that require a great deal of data entry. An example of form fill that requires users to fill in their name, address and city would have three fields and many computer programs like video rental software, financial systems, pay roll systems etc. are still purely forms-based[3].

Advantages of Form Fill-ins

- ✓ *Simple* data entry
- ✓ *Easy to learn* since the fields are predefined and need only be recognized
- ✓ *Guides* the user via the *predefined rules*.

Drawbacks of Form Fill-ins

- × Consumes a lot of screen space which can distract users if not well arranged.
- × Brings rigidity of business processes.

C. Menu selection

A menu is a range of options displayed on the screen from which a user can select and execute one or more of the options to change the state of the interface. Menu-selection enables the user to select a command from a pre-defined selection of commands arranged in menus as they observe the effect of the selection. Well grouped labels on the menus/commands are understandable and help users accomplish their tasks with negligible learning or memorization. This is because finding a command or menu item is based on recognition as opposed to recall memory tasks. To save screen space, menu items are often clustered in pull-down or pop-up menus[4].

Advantages of Menu Selection

- ✓ Ideal for novice (new) users or intermittent users.
- ✓ Can appeal to expert users if display and selection mechanisms are rapid and if appropriate "shortcuts" are implemented.

[3] Designing for Effective Human/Computer Interaction (4th ed.), Schneiderman, B., Plaisant, C.: Pearson Education, Inc.: ISBN 0-321-19786-0, (2005). Pg. 313

[4] Designing for Effective Human/Computer Interaction (4th ed.), Schneiderman, B., Plaisant, C.: Pearson Education, Inc.: ISBN 0-321-19786-0, (2005). Pg 321

- ✓ Users can explore and look around in the menus for the appropriate command, unlike having to remember the name of a command *and* its spelling when using command language.
- ✓ Structures decision making.
- ✓ Allows easy support of error handling as the user's input does not have to be parsed (as with command language).

Drawbacks of Menu Selection

- × Too many menus may lead to information overload or complexity of discouraging proportions.
- × May be slow for frequent users.
- × May not be suited for small graphic displays.

D. Graphical direct manipulation

Direct manipulation applications represent data as graphical objects that can be manipulated directly using pointing devices on the screen. This allows users perform operations on the application's data and the applications are normally implemented as window systems.

Graphical metaphors are used to represent the objects on the screen and operations on these objects are performed by manipulating them directly using pointing, dragging and dropping or selecting actions. The system responds immediately to the user actions by changing appearance of the objects, for example recycle bin becomes full, when a document is put into it[5].

Advantages of Direct Manipulation

- ✓ Visually presents task concepts.
- ✓ Easy to learn.
- ✓ Errors can be avoided more easily.
- ✓ Encourages exploration.
- ✓ High subjective satisfaction.
- ✓ Recognition memory as opposed to recall memory

[5] Designing for Effective Human/Computer Interaction (4th ed.), Schneiderman, B., Plaisant, C.: Pearson Education, Inc.: ISBN 0-321-19786-0, (2005). Pg, 248-250

Drawbacks of Direct Manipulation

- May be more difficult to programme
- Not suitable for small graphic displays
- Spatial and visual representation is not always preferable.

INDUSTRY-ACCEPTED INTERFACE DESIGN STYLES GUIDELINES AND STANDARDS

The guidelines for *good form filling, menu selection and direct graphical manipulation* interface design are based on three main goals:

i. User Support
ii. Interaction Design
iii. Screen design

a) **User support**

- ✓ Provide both context-sensitive and object-sensitive help to the users[6].

b) **Interaction design**

- ✓ Make interactions as direct as possible by using selecting, dragging, etc.
- ✓ Make operations reversible when possible.
- ✓ Issue a "warning" message before any destructive operation.
- ✓ Always display clearly marked object for exiting program
- ✓ Provide keyboard shortcuts for most often used commands[7].

c) Screen design

- ✓ Use relatively less arbitrary metaphors to represent objects
- ✓ Display only objects which can be manipulated at the given time
- ✓ Represent the state of the object too, possibly by color coding.
- ✓ Keep consistency by putting common objects at the same place on all screens[8].

---

[6] Human-computer interaction (3rd ed.), Dix, A., Finlay, J., Abowd, G., & Russell, B. New Jersey: Prentice Hall. ISBN-10: 0130461091, ISBN-13: 978-0130461094, (2004). Pg 53, 56

[7] Human-computer interaction (3rd ed.), Dix, A., Finlay, J., Abowd, G., & Russell, B. New Jersey: Prentice Hall. ISBN-10: 0130461091, ISBN-13: 978-0130461094, (2004). Pg 35

[8] Human-computer interaction (3rd ed.), Dix, A., Finlay, J., Abowd, G., & Russell, B. New Jersey: Prentice Hall. ISBN-10: 0130461091, ISBN-13: 978-0130461094, (200 4). Pg. 37

Content Covered in Week 10: Interaction Design Styles

(i) We have defined the basic Interaction design style concepts

(ii) We have described the typical interaction design styles available for HCI designers

(iii) We have described the industry accepted interface design guidelines and standards

References

1. Human-computer interaction (3rd ed.), Dix, A., Finlay, J., Abowd, G., & Russell, B. New Jersey: Prentice Hall. ISBN-10: 0130461091, ISBN-13: 978-0130461094, (2004).
2. Designing for Effective Human/Computer Interaction (4th ed.), Schneiderman, B., Plaisant, C.: Pearson Education, Inc.: ISBN 0-321-19786-0, (2005).
3. The design of everyday things, Norman, D. A. New York: Basic Books. ISBN-10: 0465067107, ISBN-13: 978-046506710, (2002).
4. Designing the user interface: Strategies for effective human-computer interaction (5th ed.), Shneiderman, B., Plaisant, C., Cohen, M., & Jacobs, S. New Jersey: Prentice Hall. ISBN-10: 0321537351, ISBN-13: 978-0321537355, (2009).