Teaching the Design and Evaluation of User-Computer Interfaces

James D. Foley

School of Engineering and Applied Science
The George Washington University
Washington, D.C. 20052

In this paper we describe a graduate Computer Science course, "The Design of User-Computer Interfaces", which the author developed and has taught since 1979. Students taking the course are normally in their second year of graduate study, and have thus been exposed to several different interactive computing environments.

The purposes of the course are to:

1. Sensitize Computer Science students to the user-oriented issues involved in designing user-computer interfaces (in the long-term, this in itself may be the most important contribution from the course).

2. Teach a structured top-down methodology for designing user-computer interfaces.

3. Introduce important fundamental ideas from the disciplines of cognitive psychology, perceptual psychology, and human motor performance; and show their application to the design of user-computer interfaces.

4. Teach students how to conduct experiments involving user-computer interfaces; and to evaluate experiments reported in the literature.

The course begins with presentation of the top-down design approach, described in [FOLE80, FOLE82], consisting of conceptual, semantic, syntactic, and lexical design. Also described are design guidelines of the type found in [CHER76, ENGE75, FOLE74, FOLE82, and HANS71] and summarized in [SHNE79]. We show how and suggest that the guidelines apply at different stages in the design process, the guidelines have psychological/physiological foundations and should be subjected to experimental confirmation. Three weeks are typically devoted to this material. Two homework assignments are given during this initial portion of the course. The first is to analyze and critique an existing user-computer interface -- typically that of WIDJET, the Waterloo Interactive Debugging Job Entry Terminal system used at GWU by beginning students to edit and submit programs and to browse through results. The students describe the WIDJET conceptual model, and use state diagrams to specify the syntactic and lexical designs. The design guidelines are used to critique the interface (which improves each year, as our Computing Center incorporates many of the class' comments). A typical report is 10 to 15 pages long, and includes a one-page "executive summary" which is given to those responsible for the system which has been studied.

The second homework is to completely design the interface to an interactive system. It is important that this and the preceding assignment involve systems with simple semantics: otherwise not enough attention can be paid to the syntactic and lexical issues. The result is a detailed design document (typically 15-25 pages in length) which could be used by a programmer to implement the system. A user's conceptual model is detailed, all semantics (including errors) are defined, screen formats are given, and the syntactic and lexical input designs are specified using BNF or state diagrams. The system design is one which maintains an individual's list of phone numbers normally found in a "flip-top" telephone directory.

While the second assignment is underway, the lectures cover concepts from cognitive psychology, perceptual psychology, and human factors, with one week devoted to each area. Relevance to user interface design is stressed, and the basis for some of the design guidelines,

discussed earlier, are made clear. The purpose of these three lectures is to both solidify the design guidelines and to provide a foundation for the discussion of experimentation, which forms the second major portion of the course.

One to two weeks are next spent discussing the methodological and, to a lesser extent, the statistical issues involved in designing and conducting a controlled experiment. Various sources are used, including [SHNE80]. Then another one to two weeks are devoted to discussing several experiments from the literature, such as [CARD78; CARD80, ENGL67, LEDG80].

Because interaction devices are important for experiments and user interfaces, the general ideas of interaction devices, interaction techniques, and interaction tasks are introduced, using [FOLE81] as a reading plus slides and a specially-edited 20-minute video tape showing a variety of interaction techniques.

Two assignments concerning experiments are given, and they parallel those used with the design phase of the course: in the first assignment the student analyzes and critiques two experiments as reported in the literature, one of which is well-done, the other of which is poorly-done. In the second assignment the students synthesize: they design, carry out, and report on an experiment. The report submitted follows the same general form as that described in the paper by B. Shneiderman in these proceedings.

The experiment is performed by teams of two to three students: it is best if one of the students on each team has had a statistics course and/or knows how to use SPSS, SAS, or a similar statistical package. The students present their results to the class at the end of the semester (in 1981, the presentation was to the Software Psychology Society).

Toward the end of the semester, as the students devote their major efforts to the experiment, lectures are given on:

user-interface issues in the design of programming languages

perception of time and experiments dealing with time

software tools for the implementation of user interfaces, such as screen and dialogue managers [HANA80, HEIN75, HOLM81, JOHN78; ROGE81].

During the semester students are expected to attend at least one meeting of the Software Psychology Society, to gain direct exposure to the work of others and to others interested in the interface between computing and psychology. Students are evaluated on the basis of their homework projects (which are quite substantial) and participation in class discussions. No exam is given.

References

CARD78 Card, S.K., English, W. K. and Burr, B. J., "Evaluation of Mouse, Rate-Controlled Isometric Joystick, Step Keys, and Text Keys for Text Selection on a CRT," Ergonomics, 21, 8(1978), pp. 601-613.

CARD80 Card, S., T. Moran and A. Newell, "The Keystroke-level Model for User Performance Time with Interactive Systems," Communications of the ACM, 23(7), July 1980, pp. 396-410.

CHER76 Cheriton, D. R., "Man-Machine Interface Design for Timesharing Systems," Proc. ACM 1976 Conference, pp. 362-365.

ENGE75 Engel, S. E. & Granda, R., "Guidelines for Man-Display Interfaces," IBM Technical Report, TR00 2720 (1975).

ENGL67 English, W. K., Engelbart, D. C. & Berman, M. L., "Display-Selection Techniques for Text Manipulation," IEEE Trans. on Human Factors in Electronics, HFE-8, 1 (March 1976), pp. 5-15.

FOLE74 Foley, J. D. and V. L. Wallace, "The Art of Natural Graphic Man-Machine Conversation," Proc. IEEE 62(4), April 1974, pp. 462-470.

FOLE80 Foley, J.D., "The Structure of Command Languages," in R. A. Guedj, et al., eds., Methodology of Interaction, North-Holland, Amsterdam, 1980, pp. 227-234.

FOLE82 Foley, J. and A. van Dam, Fundamentals of Interactive Computer Graphics, Addison-Wesley Pub. Co., Reading, Ma., 1982.

HANA80 Hanau, P. and D. Lenorowitz, "Prototype and Simulation Tools for User/Computer Dialogue Design", Proceedings 1980 SIGGRAPH Conference, published as Computer Graphics 14, 3(July 1980), pp. 271-278.

HANS71 Hansen, W. J., "User Engineering Principles for Interactive Systems," Proc. 1971 Fall Joint Computer Conf., pp. 523-532.

HEIN75 Heindel and Roberto. LANGPAK - An Interactive Language Design System, American Elsiver, 1975.

HOLM81 Holmes, H., J. Thomas, and G. Rogers, "A Clause-Structured Interaction Handler". The Gepore GWU-IIST-81-20, Wasington, D.C., 1981.

JOHN78 Johnson and Lesk, "Language Development Tools", Bell System Technical Journal 57(6).

LEDG80 Ledgard, H. et. al., "The Natural Language of Interactive Systems," Communications of the ACM 23(10), October 1980, pp. 556-563.

ROGE81 Rogers, G. and M. Feldman, "An Intermediate Language and Interpreter for Style - Independent Interactive Systems," The George Washington University, Institute for Information Science and Technology Technical Report GWU-IIST-81-21, 1981.

SHNE79 Shneiderman, B., "Human Factors Experiments in Designing Interactive Systems," Computer 12 (12), December 1979, pp. 9-19.

SHNE80 Shneiderman, B., "Software Psychology", Winthrop, 1980.