

# Projektowanie ergonomicznych i użytecznych aplikacji komputerowych (część 1)

---

dr inż. Walery Susłow  
mgr inż. Michał Statkiewicz

# Użyteczność i ergonomia



- Użyteczność (cecha produktu) – opisuje jego przydatność dla użytkownika i efektywność wykonywania zadań, do których jest przeznaczony.
- Ergonomia (cecha) – stopień dopasowania produktu do cech użytkownika (fizycznych i psychicznych), mającego na celu minimalizację szkodliwości pracy i maksymalizację jej efektywności.
- Norma ISO 9241-11 (1998)



# Hasła pokrewne



- Przyjazność (dla użytkownika)
- Efektywność, wydajność (pracy)
- Łatwość uczenia się (obsługi produktu)
- Łatwość używania produktu
- Dostępność
- Funkcjonalność
- Satysfakcja z użytkowania
- Niezawodność

# Oprogramowanie – istotne elementy



- Interfejs użytkownika (UI) – wygląd i zachowanie aplikacji
- GUI (80% informacji jest przekazywanych wizualnie)
- Paradygmat WIMP
- Elementy UI – kontrolki, widgety, kontenery
- Programowanie zdarzeniowe
- Funkcje dopasowane do zastosowań

# Przyczyny niedopasowania UI



- Przyzwyczajenia i wiedza programisty / projektanta
- Zróżnicowanie cech, umiejętności i przyzwyczajeń użytkowników
- Ograniczona dostępność specjalistycznej wiedzy nt. projektowania GUI
- Stopniowa zmiana technologii i standardów ograniczająca przydatność części ustalonej wiedzy

# Metody projektowania UI



- User-centered design – projektowanie skierowane na użytkownika
- Wiedza specjalistyczna
- Przewodniki stylu, zalecenia projektowe
- Analiza zadaniowa, KLM
- Prototypowanie

# Pytania



- Kim są użytkownicy, co umieją, czego mogą się nauczyć?
- Czego użytkownicy potrzebują i co chcą osiągnąć?
- W jakim otoczeniu i w jakiej dziedzinie użytkownicy pracują?
- Co ma wykonywać użytkownik, a co komputer?
- Czego użytkownicy się spodziewają i w jaki sposób pracują?

# Model użytkownika, model aplikacji

- Model użytkownika – zbiór przyzwyczajeń, cech i umiejętności użytkownika, oraz jego wyobrażeń na temat zachowania aplikacji.
- Model aplikacji (zachowania) – zbiór algorytmów opisujących działania programu generowane w trakcie interakcji z użytkownikiem.



# Użyteczność aplikacji



- Aplikacja jest użyteczna, kiedy robi to, czego spodziewa się po niej użytkownik.
- Więc: aplikacja jest użyteczna, kiedy model zachowania aplikacji jest dopasowany do modelu użytkownika.
- Więc: nowe aplikacje są użyteczne, jeśli działają tak jak stare?



# Dostępna wiedza



- Ergonomia (oprogramowania)
- Psychologia
- Kognitywistyka
- HCI (ang. human-computer interaction)

# Dziedzina HCI



- HCI – ang. human-computer interaction (wcześniej computer-human interaction) – interdyscyplinarna dziedzina wiedzy zajmująca się badaniem oddziaływań (interakcji) między użytkownikami a komputerami, mająca na celu usprawnienie tych oddziaływań.

# Kognitywistyka



- Kognitywistyka jest dziedziną nauki zajmującą się zjawiskami dotyczącymi umysłu, w szczególności ich modelowaniem. Na m. Na jej określenie używane są też pojęcia nauki kognitywne (ang. Cognitive Science), ce), bądź nauki o poznaniu.

oznaniu.

narną, na pograniczu dziedzin psychologii znawczej, neurologii, filozofii umysłu, sztucznej eligencji oraz lingwistyki.



# Testowanie użytkowników



- Informacje potrzebne do zaprojektowania GUI można zdobyć obserwując użytkowników wykonujących zadania zbliżone do planowanych w projektowanej aplikacji.
- Pomiar reakcji uczestników testu dostarcza dużej ilości danych przydatnych dla optymalizacji aplikacji.



# Przewodniki stylu



- Object-Oriented Interface - Design: IBM Common User Access Guidelines
- Common Desktop Environment 1.0 Style Guide and Certification Checklist
- OSF/Motif Style Guide
- Macintosh Human Interface Guidelines
- The Windows Interface Guidelines for Software Design
- Taligent Style Guide
- JMAPI User Interface Style Guide Java Management API. - Sun Microsystems
- Design Guide for Multiplatform Graphical User Interfaces - Bellcore

# Analiza zadaniowa



- Analiza działań, które użytkownik będzie wykonywał podczas pracy z aplikacją, pozwala zaprojektować rozmieszczenie dostępnych w programie funkcji.
- Metoda KLM (ang. Keystroke-level) pozwala optymalizować GUI, oceniając czas potrzebny użytkownikowi na wykonanie wszystkich elementarnych czynności w trakcie pracy.

# Heurystyka



- Z powodu różnic między użytkownikami trudne jest zaprojektowanie GUI użytecznego dla wszystkich.
- Zespół ekspertów może określić rozwiązania optymalne dla większości użytkowników, skracając proces projektowania aplikacji
- Alternatywnie można przy pomocy ankiet i testów wybrać rozwiązanie odpowiadające jak największej części planowanej populacji użytkowników.



# Testowanie aplikacji



- Nowotworzone aplikacje zwykle nie są doskonałe.
- Już kilka pierwszych osób testujących aplikację wyłapie większość istotnych błędów i niedociągnięć (Nielsen).

# Nieco wskazówek



- Konsekwencja!
- Dostępność najczęściej używanych funkcji
- Krótkie i przyjazne komunikaty
- Ograniczenie zadań użytkownika do tego, co istotne
- Ograniczenie liczby i pogrupowanie logiczne dostępnych funkcji
- Występowanie czytelnego sprzężenia zwrotnego
- Wzorowanie się na istniejących, popularnych i uznanych rozwiązaniach

# Nieco więcej wskazówek



- Apple Computer, Inc. (1992), Macintosh Human Interface Guidelines, Addison-Wesley Publishing Co., Reading, MA.
- Bajwa, S.(1995), "Effects of inconsistencies on users performance and subjective satisfaction", Unpublished report, Department of Computer Science, University of Maryland.
- Barnard, P., Hammond, J. , Morton, J., Long, J. and Clark, I. (1981), "Consistency and compatibility in human-computer dialogue", International Journal of Man-Machine Studies 15, 87-134.
- Bodart, F., Hennebert, A.-M., Leheureau, J.-M. and Vanderdonckt, J. (1994), Towards a dynamic strategy for computer-aided visual placement, In Catarci, T., Costabile, M., Levialdi, S., and Santucci, G. (Editors), Proc. Advanced Visual Interfaces Conference '94, ACM Press, New York, 78-87.
- Chimera, R. and Shneiderman, B. (1993), User interface consistency: An evaluation of original and revised interfaces for a videodisk library, In Shneiderman, B. (Editor), Sparks of Innovation in Human-Computer Interaction, Ablex Publishers, Norwood, NJ, 259-273.
- Chin, J.P., Diehl, V.A. and Norman, K. (1988), Development of an instrument measuring user satisfaction of the human-computer interface. Proc. of the Conference on Human Factors in Computing Systems, CHI'90, ACM, New York, 213-218.
- Coll, R., and Wingertsman, A. (1990), The effect of screen complexity on user preference and performance, International Journal of Human-Computer Interaction 2,3, 255-265.
- Comber, T. and Maltby, J. (1996), Investigating layout complexity, In Vanderdonckt, Jean (Editor), Computer-Aided Design of User Interfaces, Press Universitaires de Namur, Namur, Belgium, 209-227.
- Frederiksen, N., Grudin, J. and Laursen, B. (1995), Inseparability of design and use: An experimental study of design consistency, Proceedings of Computers in Context'95, Aarhus, Aarhus University, 83-89.
- Grudin, J. (1989), The case against user interface consistency, Communications of the ACM 32,10, ACM Press, New York, 1164-1173.
- Grudin, J., and Norman, D. (1991), Language evolution and human-computer interaction, Proc. of the Thirteenth Annual Conference of the Cognitive Science Society, Hillsdale, New Jersey, 611-616.
- Harrison, M.D. and Thimbleby, H.W. (1985), Formalizing guidelines for the design of interactive systems. In Proc. of BCS HCI Specialist Group Conference HCI85, 161-171.
- Jeffries, R., Miller, J., Wharton, C. and Uyeda, K. (1991), User interface evaluation in the real world: A comparison of four techniques, Proc. of CHI 1991, ACM, New York, 119-127.
- Karat, C.-M., (1992), Cost-justifying human factors support in development projects, Human Factors Society Bulletin 35, 8.
- Kellogg, W. (1987), Conceptual consistency in the user interface: Effects on user performance, In Proc. of Interact '87 Conference on Human-Computer Interaction, Stuttgart.
- Lynch, P. (1994), Visual design for the user interface, pt. 1 design fundamentals, Journal of Biocommunications 21, 1, 22-30.
- MacIntyre, F., Estep, K.W. and Sieburth, J.M. (1990), Cost of user-friendly programming, Journal of Fourth Application and Research 6, 2, 103-115.
- Mahajan, R. and Shneiderman, B. (1995), A family of user interface consistency checking tools: Design analysis of SHERLOCK, Proc. of NASA Twentieth Annual Software Engineering Workshop, 169-188.
- Microsoft, Inc. (1992), The Windows Interface: An Application Design Guide, Microsoft Press, Redmond, WA.
- Mullet, K.(1995), Organizing information spatially, Interactions, July 95, 15-20.
- Nielsen, H.(1989), Coordinating User Interfaces for Consistency Checking, Ed. Nielsen, J. Academic Press Inc., London.
- Polson, P., Muncher, E. and Engelbeck, G. (1986), A test of a common elements theory of transfer, Proc. of CHI'86, ACM, New York, 78-83.
- Reisner, P. (1990), What is consistency?, Proc. of the IFIP Third International Conference on Human-Computer Interaction, Interact '90, Elsevier Science Publishers, B.V., North-Holland, 175-181.
- Rosenberg, D. (1989), Cost benefit analysis for corporate user interface standards: What price to pay for consistent look and feel?, Coordinating User Interfaces for Consistency Checking, Ed. Nielsen, J. Academic Press Inc., London, 21-34.
- Sears, A. (1993), Layout Appropriateness: A metric for evaluating user interface widget layouts, IEEE Transactions on Software Engineering 19, 7, 707-719.
- Sears, A. (1994), AIDE: A step towards metric-based interface development tools, Proc. of UIST '95, ACM, New York, 101-110.
- Shneiderman, B. (1998), Designing the user interface: Strategies for Effective Human-Computer Interaction: Third Edition, Addison-Wesley Publ. Co., Reading, MA.
- Shneiderman, B., Chimera, R., Jog, N., Stimart, R. and White, D. (1997), Evaluating spatial and textual style of displays, In MacDonald, L. W. and Lowe, A. C. (Editors) Display Systems: Design and Applications, John Wiley & Sons, Chichester, UK, 83-96.
- Smith, D.C., Irby, C., Kimball, R., Verplank, B. and Harslem, E. (1982), Designing the star user interface, Byte 7, 4, 242-282.
- Streveler, D. and Wasserman, A. (1987), Quantitative measures of the spatial properties of screen designs, Proc. of INTERACT '87, Elsevier Science, Amsterdam, 125-133.
- Tullis, T.S. (1983), The formatting of alphanumeric displays: A review and analysis, Human Factors 25, 657-682.
- Tullis, T. S. (1988), A system for evaluating screen formats: Research and application, In Hartson, H. Rex and Hix, Hartson (Ed.), Advances in Human-Computer Interaction: Volume 2, Ablex Publishing Corp., Norwood, NJ, 214-286.
- Tullis, T. S. (1997), Screen design, In Helander, M. Landauer, T., and Prabhu, P (Editors), Handbook of Human-Computer Interaction: Second Edition, Elsevier Science, Amsterdam, The Netherlands, 503-531.
- Vanderdonckt, J. and Gillo, X. (1994), Visual techniques for traditional and multimedia layouts, In Catarci, T., Costabile, M., Levialdi, S. and Santucci, G. (Editors), Proc. Advanced Visual Interfaces Conference '94, ACM Press, New York, 95-104.
- Wiecha, C., Bennett, W., Boies, S. and Gould, J. (1989), Generating Highly Interactive User Interface, Proc. of CHI'89, ACM, New York, 277- 282.
- Wolf, R. (1989), Consistency as process, Coordinating User Interfaces for Consistency Checking, Ed. Nielsen, J. Academic Press Inc., London, 89-92.



# Co można zaprojektować w GUI?



- Rozmieszczenie elementów (położenie, grupowanie, dostępność)
- Rodzaj i wygląd elementów (kształty, rozmiary, kolory)
- Przekazywane treści (dialogi, ikony)
- Sposób pracy z programem (kolejność działań, dialogi, makra)
- Rodzaj sprzężenia zwrotnego (tekst, animacja, dźwięk)



# Układ graficzny elementów



- Prawo Fitt'a  $T = a + b \log_2 \left( \frac{D}{W} + 1 \right)$
- Prawo Hick'a  $T = b \log_2(n + 1)$
- Miller:  $7 \pm 2$  elementy?
- grupowanie
- rozmieszczenie przestrzenne



# Kolory i ergonomia



- Interfejs powinien być czytelny w różnych warunkach oświetleniowych
- Interfejs powinien być czytelny dla osób z upośledzonym postrzeganiem kolorów – używane barwy powinny się zauważalnie różnić jasnością
- W każdym przypadku należy zapewnić odpowiednio duży kontrast jasności między tekstem a tłem
- Należy unikać kolorów z przeciwnych krańców widma występujących obok siebie
- Większe obszary jednolitego koloru powinny mieć ograniczone nasycenie i jasność
- Kolory wpływają na emocje użytkownika



# Czcionki i tekst



- Zalecane jest stosowanie czcionek proporcjonalnych do czytania tekstu i czcionek o stałej szerokości do edycji
- W przypadku mniejszych rozmiarów liter należy używać czcionek bezszeryfowych
- Wielkość liter na ekranie nie powinna być mniejsza niż 2,5mm
- Komunikaty muszą być jak najkrótsze – użytkownicy nie lubią dużo czytać na ekranie



# Metafory w GUI



- Nie zawsze jest możliwe dopasowanie się do modelu użytkownika docelowej grupy – należy wtedy wymusić nowy model, łatwo przyswajalny – umożliwia to zastosowanie metafor.
- Metafory to na ogół graficzne odwołanie się do powszechnej wiedzy, skojarzone z obsługą wybranej funkcji w programie.





# Sprzężenie zwrotne



- Użytkownicy są niecierpliwi – już kilkusekundowa zwłoka w działaniu programu może wywołać znaczną irytację.
- Każde działanie użytkownika powinno być natychmiast potwierdzone sygnałem zwrotnym (wizualnym bądź dźwiękowym)
- Jeśli konieczne są dłuższe obliczenia, należy pokazać pasek postępu, i wyraźny sygnał zakończenia obliczeń.



# Obsługa błędów



- Nie da się całkowicie zapobiec występowaniu nieprzewidzianych sytuacji
- Sposób obsługi błędów powinien zależeć od doświadczenia docelowej grupy użytkowników
- Ilość występujących błędów można znacznie ograniczyć ograniczając swobodę użytkownika (sterowanie dostępnością funkcji, okna modalne, kreatory)



# Dynamiczne dopasowanie GUI do użytkownika

- Nie jest możliwe stworzenie niezmiennego interfejsu użytecznego dla wszystkich użytkowników.
- Ręczna personalizacja jest mało efektywna, z powodu ograniczonej wiedzy i możliwości samodzielnej oceny efektywności użytkowników.
- Rozwiązaniem jest pomiar reakcji użytkownika w trakcie pracy i modyfikacja GUI w oparciu o ustalone algorytmy.

# Dodatkowe informacje



- <http://www.uzytecznosc.pl/>
- <http://www.webusability.pl/>
- <http://elsinterakcja.pl/>
- <http://rzeczy.blox.pl/html>
- <http://www.useit.com/>
- <http://www.usability.gov/>
- <http://www.usabilityfirst.com/>
- <http://en.wikipedia.org/wiki/Usability>

