

Rafał MICHALSKI*

JAKOŚĆ UŻYTKOWA W PROCESIE WYTWARZANIA OPROGRAMOWANIA

W pracy przedstawiono przegląd podejść do zapewniania jakości użytkowej systemów informatycznych. Zaprezentowano historyczny rozwój modeli wytwarzania oprogramowania. Wyróżniono dwie grupy: klasyczną, wywodzącą się z inżynierii oprogramowania oraz zorientowaną na końcowego użytkownika. Przedstawiono różne typologie metod oceny oprogramowania oraz nakreślono aktualne problemy i potencjalne kierunki badań dotyczących przyjaznych systemów informatycznych.

Słowa kluczowe: *ergonomia produktów informatycznych, interakcja człowieka z komputerem, cykle życia oprogramowania, systemy informatyczne*

1. Wprowadzenie

Wciąż zwiększająca się liczba komputerów osobistych i ich wszechobecność powoduje, że coraz częściej już nie wysokiej klasy specjalista-informatyk, ale przeciętny człowiek staje przed koniecznością komunikacji z maszyną cyfrową. W dobie powszechnej informatyzacji wzrasta również uzależnienie podmiotów gospodarczych od różnego rodzaju oprogramowania, począwszy od zwykłych przeglądarek internetowych i programów do obsługi poczty elektronicznej, a skończywszy na złożonych systemach informatycznych zarządzania klasy MRP II czy ERP. Liczne problemy występujące podczas wdrożeń systemów komputerowych, a także wykorzystywanie przez użytkowników tylko w niewielkiej części nierzadko olbrzymich możliwości funkcjonalnych pokazują, że pomimo zapewnienia wszystkich technicznych wymagań, proponowane rozwiązania informatyczne nie zawsze są akceptowane. W takich warunkach zapewnianie

* Instytut Organizacji i Zarządzania, Politechnika Wroclawska, Wybrzeże Wyspiańskiego 27, 50-372 Wrocław, e-mail: Rafal.Michalski@pwr.wroc.pl

wysokiego poziomu jakości użytkowej podczas wytwarzania oprogramowania może być czynnikiem decydującym o przetrwaniu i sukcesie firmy.

W dalszej części artykułu przedstawiono przegląd podejść do zapewniania jakości użytkowej systemów informatycznych. Omawiane podejścia podzielono na dwie grupy: klasyczną i obejmującą modele zorientowane na końcowego użytkownika systemu informatycznego. Wytwarzanie wysokiej jakości oprogramowania nierozdzielnie wiąże się z oceną gotowych programów komputerowych lub ich prototypów, w związku z tym omówiono także różne typologie metod oceny projektowanych systemów informatycznych. W części końcowej wskazano na aktualne problemy i przyszłe kierunki badań w obszarze jakości użytkowej systemów interakcyjnych.

2. Jakość użytkowa

Całość zagadnień związanych z jakością użytkową wytwarzania oprogramowania jest dziedziną stosunkowo młodą i w związku z tym używane pojęcia są mało uporządkowane. Szerokie omówienie różnych wymiarów jakości systemów informatycznych oraz niuansów znaczeniowych stosowanej terminologii w publikacjach i wielu międzynarodowych standardach zamieszczono np. w pracach [2], [37], [44]. W polskiej literaturze angielskie słowo *usability* jest tłumaczone w różny sposób, np. *przydatność* [4], *użyteczność* [13], *użytkowość* [7], *jakość ergonomiczna* [28]. Próbę ujednolicenia nomenklatury stosowanej w tej dziedzinie w Polsce podjął Sikorski [42]. Zaproponował on termin *jakość użytkowa* jako odpowiednik angielskiego pojęcia *usability*. W niniejszej pracy przyjęto właśnie to tłumaczenie.

W literaturze i międzynarodowych standardach istnieje wiele definicji jakości użytkowej, spośród których najlepiej istotę zagadnienia oddają następujące:

- Według Jakoba Nielsena „jakość użytkowa to suma cech charakterystycznych produktu, które sprawiają, że oprogramowanie jest wydajne, zrozumiałe oraz łatwe do nauczenia się i użycia” (ang. *Usability is the sum of product characteristics which make software productive and easy to understand, learn and use*) [29]. Nielsen zauważa także, że jakość użytkowa oprogramowania zależy od łatwości nauczenia się obsługi (ang. *learnability*), wydajności (ang. *efficiency*), łatwości zapamiętywania obsługi (ang. *memorability*), liczby popełnianych błędów (ang. *errors*) i od satysfakcji użytkownika (ang. *satisfaction*) [30].

- W międzynarodowej normie ISO 9241 przyjęto, że jakość użytkowa to „stopień, w jakim dany produkt pozwala określonym użytkownikom osiągać cele w sposób skuteczny, efektywny (wydajny) i z satysfakcją (zadowoleniem)” (ang. *usability: the extent to which product is used by specified users to achieve specific goals with effectiveness, efficiency and satisfaction*) [23].

- Dla Preece i in. jakość użytkowa związana jest z „łatwością uczenia się, skutecznością używania i dostarczaniem przyjemnych doznań użytkownikowi” (ang. *easy to learn, effective to use, and provide an enjoyable user experience*) [33].

Przegląd i omówienie licznych definicji jakości użytkowej, z uwzględnieniem pojęć stosowanych w różnych normach międzynarodowych, można znaleźć w pracach Reeda i in. [37], Bevana [2], International standards for HCI and usability [22], Gasika [13], Sikorskiego [42]. Zestawienie definicji z różnych źródeł, odnoszących się do jakości użytkowej podają także O’Neill [31], Flowers [10], Folmer i Bosch [11]. Przedstawione w tej części artykułu definicje jakości użytkowej pokazują, że pojęcie to nie jest rozumiane jednoznacznie. Różni autorzy mają nieco odmienne zdanie na temat tego, jakie atrybuty najlepiej opisują jakość użytkową. Używa się różnych nazw odnoszących się do tych samych pojęć i zdarza się, że te same składowe zajmują inne miejsce w hierarchii pojęć określających jakość użytkową. Niemniej jednak istniejące definicje w dużej części pokrywają się, a pewnym konsensusem wydaje się być podejście zaproponowane w normie ISO 9241 [23].

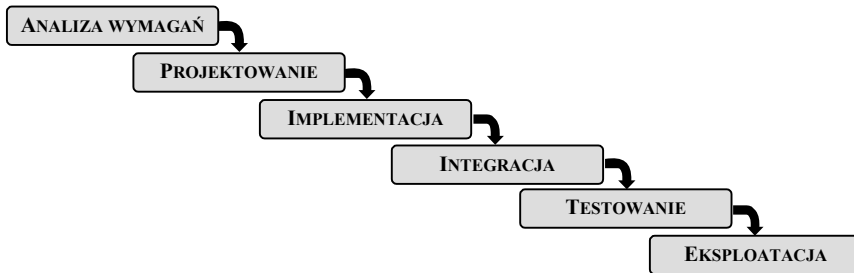
3. Modele wytwarzania oprogramowania

Ostateczna jakość systemów interakcyjnych zależy w dużym stopniu od stosowanego przez producenta podejścia do wytwarzania oprogramowania. Nie ulega też wątpliwości, że odpowiednie wykorzystanie wiedzy oraz wypracowanych metod i technik w obszarze interakcji człowiek–komputer (ang. HCI – *Human–Computer Interaction*) zwiększa prawdopodobieństwo uzyskania programów przyjaznych użytkownikom. Wraz z dynamicznym rozwojem techniki i rozpowszechnieniem komputerów osobistych zmieniała się również filozofia projektowania produktów informatycznych. W dalszych rozdziałach omówiono wybrane podejścia, obejmujące zarówno klasyczne cykle życia systemów informatycznych, jak i te stawiające użytkownika oprogramowania w centrum procesu projektowego.

3.1. Klasyczne cykle życia systemów informatycznych

W inżynierii oprogramowania wypracowano wiele różnych podejść związanych z zarządzaniem procesem wytwarzania programów. Wśród najbardziej rozpowszechnionych modeli cyklu życia systemów informatycznych znajdują się [8], [33], [36], [42]: model kaskadowy, model spiralny, model RAD (ang. *Rapid Application Development*).

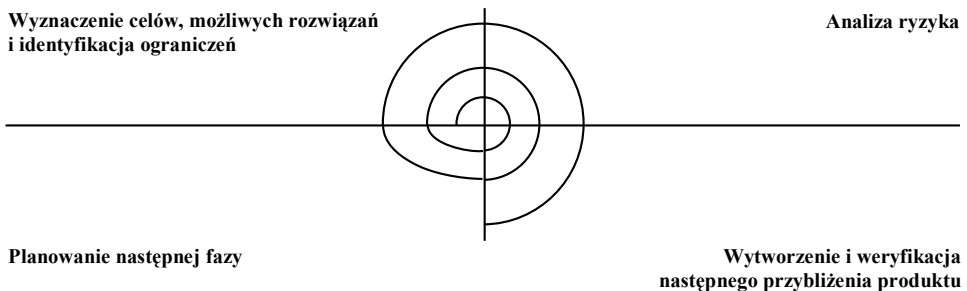
Model kaskadowy zakłada realizowanie poszczególnych etapów wytwarzania oprogramowania w ściśle zdefiniowanej kolejności (rys. 1). Każdy z wymienionych etapów musi zostać zakończony przed rozpoczęciem następnego.



Rys. 1. Kaskadowy cykl życia oprogramowania

Źródło: Opracowanie własne na podstawie prac: [8], [33], [36], [42].

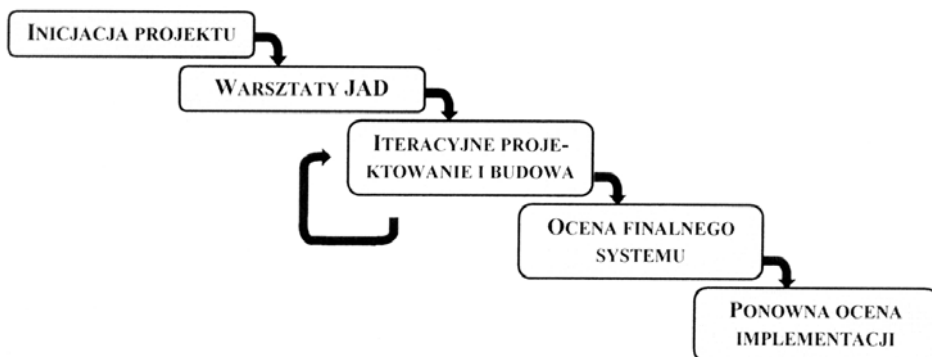
W modelu spiralnym, zaproponowanym przez Barry'ego Boehma [3], postuluje się iteracyjne podejście do wytwarzania produktów informatycznych (rys. 2). Uwzględnia się tutaj działania polegające na wyznaczeniu celów, możliwych rozwiązań i identyfikacji ograniczeń, analizie ryzyka, wytworzeniu i weryfikacji następnego przybliżenia produktu oraz planowaniu następnej fazy.



Rys. 2. Spiralny cykl życia oprogramowania

Źródło: Opracowanie własne na podstawie prac: [33], [36], [42].

W modelu RAD wykorzystuje się podejście ukierunkowane na końcowego użytkownika, w celu zminimalizowania ryzyka zmiany wymagań w trakcie trwania projektu. Kluczowymi elementami modelu RAD są krótkie cykle, na końcu których część lub całość systemu musi zostać wykonana oraz warsztaty JAD (ang. *Joint Application Development*), polegające na określaniu wymagań systemu informatycznego wspólnie przez programistów i przyszłych użytkowników (rys. 3).



Rys. 3. Cykl życia oprogramowania RAD
 Źródło: Opracowanie własne na podstawie pracy [33].

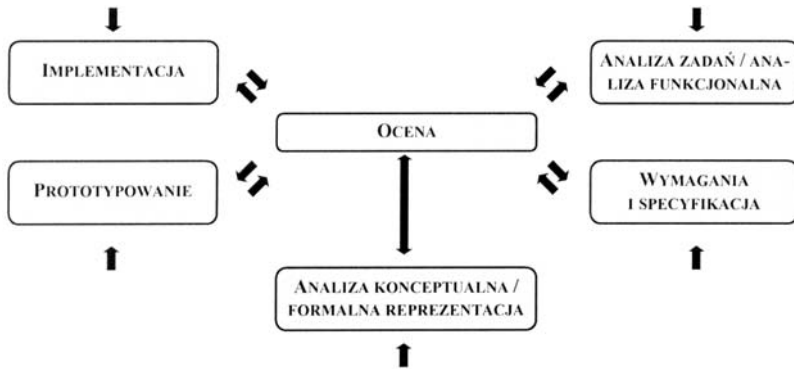
3.2. Wytwarzanie oprogramowania zorientowane na użytkownika

Wymagania rynkowe i coraz powszechniejsze wykorzystanie komputerów spowodowały konieczność większego uwzględnienia potrzeb, możliwości i ograniczeń końcowego odbiorcy produktu informatycznego. Sytuacja ta przyczyniła się do opracowania metod wytwarzania oprogramowania zorientowanych na użytkownika (ang. *user-centered software design*). Wytwarzanie ukierunkowane na użytkownika opiera się na trzech podstawowych elementach ([9], [16], [24], [45]): wczesnym skupieniu się na użytkownikach i zadaniach, empirycznych pomiarach, projektowaniu iteracyjnym.



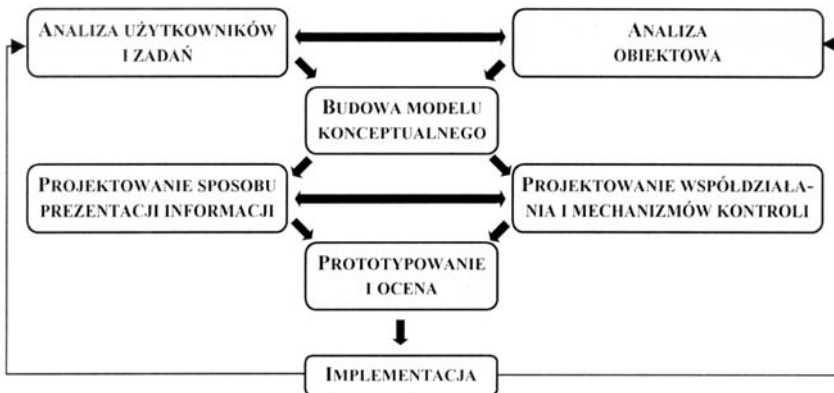
Rys. 4. Działania w zorientowanym na użytkownika procesie wytwarzania oprogramowania
 Źródło: Opracowanie własne na podstawie prac: [1], [42].

W podejściu tym głównym celem jest zapewnienie jakości użytkowej podczas wszystkich etapów cyklu życia oprogramowania [4], [31], [38], [42]. Uważa się, że podejście takie wprawdzie nie gwarantuje, że zaprojektowany system informatyczny będzie charakteryzował się wysoką jakością użytkową, ale zdecydowanie zwiększa prawdopodobieństwo sukcesu [14]. Ukierunkowany na użytkownika proces projektowania systemów interakcyjnych został zdefiniowany w międzynarodowej normie ISO/DIS 13407 [24]. Działania związane z tym podejściem zilustrowano na rysunku 4.



Rys. 5. Model gwiazdy cyklu życia oprogramowania

Źródło: Opracowanie własne na podstawie prac: [18], [33].



Rys. 6. Model cyklu życia oprogramowania Collinsa

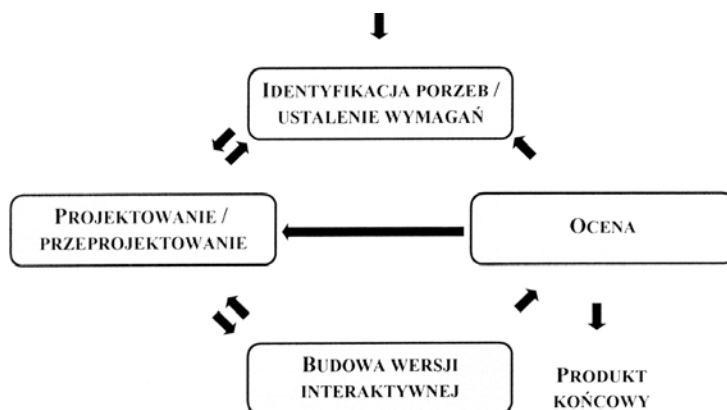
Źródło: Opracowanie własne na podstawie prac: [4], [6].

Wśród modeli cyklu życia oprogramowania, uwzględniających konieczność zapewnienia jakości użytkowej w każdej fazie wytwarzania oprogramowania, na uwagę zasługują modele: gwiazdy, Collinsa, uproszczony model Preece, Rogers i Sharp oraz model inżynierii jakości użytkowej.

Zaproponowany przez Hartsona i Hixa [18] model w postaci gwiazdy (rys. 5) powstał na podstawie obserwacji pracy projektantów interfejsów. Centralną jego częścią jest ewaluacja. Zakończenie jakiegokolwiek czynności wymusza jednocześnie ocenę jej rezultatów.

W modelu Collinsa [6] zwraca się uwagę na opracowanie odpowiedniego modelu konceptualnego, wynikającego z analizy potrzeb użytkowników oraz na iteracyjny charakter działań związanych z tworzeniem prototypów i ich oceną przez użytkowników (rys. 6).

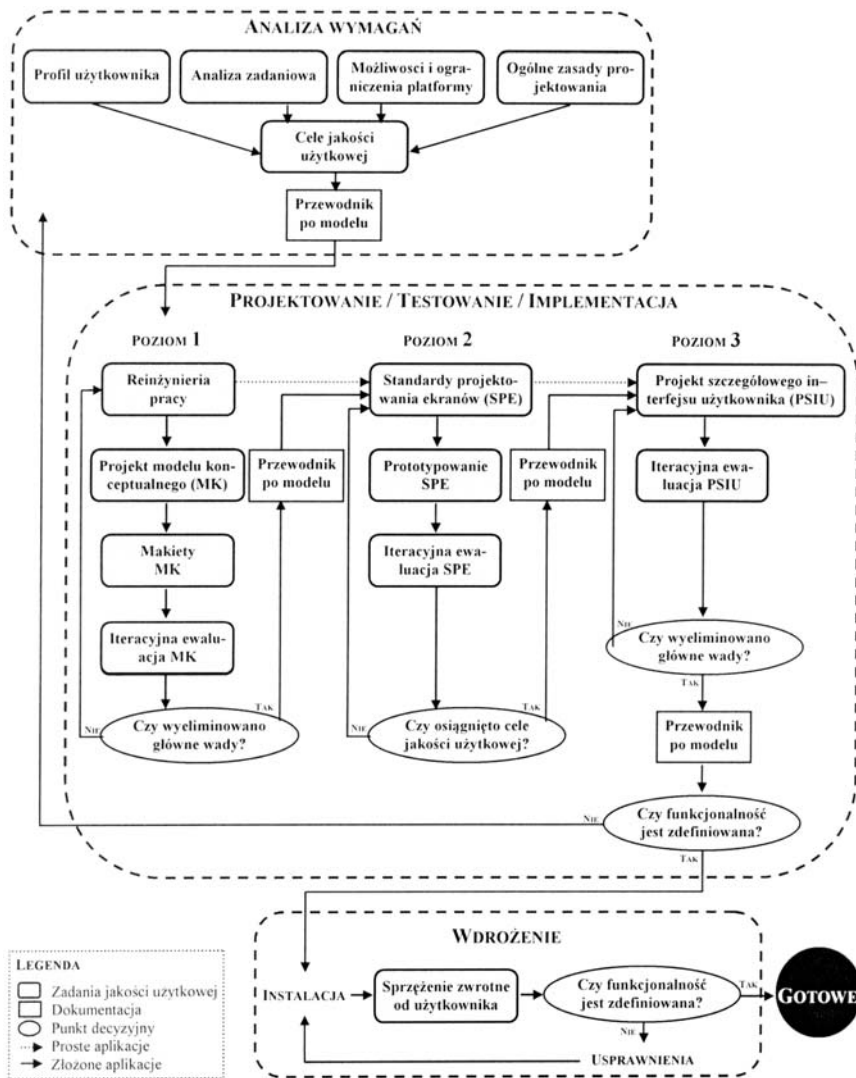
Preece, Rogers i Sharp [33] zaproponowały uproszczony, ogólny model wytwarzania dowolnych produktów o charakterze interakcyjnym (rys. 7). Autorki zwracają uwagę na cztery podstawowe grupy czynności: (1) określenie potrzeb użytkowników i ustalenie wymagań, (2) opracowanie kilku propozycji spełniających zidentyfikowane wymagania, (3) budowę interaktywnych wersji prototypów oraz (4) ocenianie wszystkiego, co zostało zbudowane podczas procesu projektowania.



Rys. 7. Uproszczony model projektowania interakcji opracowany przez Preece, Rogers i Sharp
Źródło: Opracowanie własne na podstawie pracy [33].

Jeden z najbardziej kompleksowych modeli cykli życia oprogramowania, uwzględniający wymiary jakości użytkowej, zaproponowała Mayhew [26]. Model ten łączy metody zapewniania jakości użytkowej z tradycyjnymi cyklami życia programów komputerowych, opracowanymi w ramach inżynierii oprogramowania. Wyróżniono trzy zasadnicze obszary działalności (rys. 8): analizę wymagań, projektowanie/testowanie/implementację oraz wdrożenie.

Zadania związane z jakością użytkową określane są już na etapie analizy wymagań. Na podstawie ustalonych w pierwszej fazie celów postuluje się stworzenie *przewodnika po modelu*, w którym zamieszczone są jasno i szczegółowo sprecyzowane kryteria jakości użytkowej i ich mierniki. Przewodnik ten jest weryfikowany i uzupełniany w trakcie trwania całego cyklu życia oprogramowania. Przedstawiony przez autorkę model uwzględnia wytwarzanie zarówno prostych, jak i złożonych systemów informatycznych.



Rys. 8. Cykl życia inżynierii zapewniania jakości użytkowej
 Źródło: Opracowanie własne na podstawie pracy [26].

4. Metody oceny jakości użytkowej

Na podstawie zaprezentowanych cykli życia oprogramowania nietrudno za-
 uważać, że zapewnianie jakości użytkowej w procesie wytwarzania systemów in-

formatycznych jest nierozzerwalnie związane z koniecznością oceny produktów kolejnych faz procesu. Widać to we wszystkich zaprezentowanych cyklach życia produktu informatycznego, wywodzących się zarówno z klasycznej inżynierii oprogramowania, jak i opartych na podejściu wytwarzania zorientowanego na użytkownika. O ważności oceny jakości użytkowej przekonani są także Preece i in. [33], którzy twierdzą, że ewaluacja wszystkiego tego, co zostało stworzone, znajduje się w samym sercu projektowania każdego systemu interakcyjnego, oczywiście włączając w to również programy komputerowe. Dix i in. [8] podkreślają natomiast konieczność uwzględnienia oceny projektowanych produktów informatycznych, z uwzględnieniem końcowych adresatów, na możliwie wczesnym etapie wytwarzania. W związku z powyższym w dziedzinie interakcji człowieka z komputerem stosuje się wiele różnorodnych metod oceny; dość często wywodzą się one z innych dziedzin nauki. W dalszej części tego rozdziału przedstawiono wybrane klasyfikacje i typologie tych metod.

Preece i in. [33] dokonali podziału wszystkich metod oceny jakości użytkowej, uwzględniając cztery paradygmaty: szybko i na brudno (ang. *quick and dirty*), testowanie jakości użytkowej (ang. *usability testing*), studiowanie w warunkach naturalnych (ang. *field studies*) oraz ewaluacja predykcyjna (ang. *predictive evaluation*). Wyróżnili oni również pięć technik oceny jakości użytkowej oprogramowania: obserwację użytkowników (ang. *observing users*), pytanie użytkowników o ich opinie (ang. *asking users*), pytanie ekspertów o ich zdanie (ang. *asking experts*), testowanie działań użytkowników (ang. *user testing*), modelowanie wykonywania zadania przez użytkownika tak, aby przewidzieć efektywność interfejsu użytkownika (ang. *modeling users' task performance*).

Ogólną klasyfikację metod oceny jakości użytkowej zaprezentowali Whitefield i in. [46]. Wyróżniono cztery klasy metod: (1) metody analityczne np. GOMS (ang. *Goals, Operators, Methods, and Selection rules*) czy KLM (ang. *Key-stroke Level Model*) – umożliwiające prognozowanie czasu wykonania zadania; (2) raporty użytkowników np. wywiady, kwestionariusze; (3) raporty ekspertów np. listy kontrolne, ocena heurystyczna; (4) metody obserwacyjne np. nieformalna obserwacja, eksperymenty w laboratoriach (rys. 9).

		UŻYTKOWNIK	
		Reprezentacyjny	Rzeczywisty
KOMPUTER	Reprezentacyjny	METODY ANALITYCZNE	RAPORTY UŻYTKOWNIKÓW
	Rzeczywisty	RAPORTY EKSPERTÓW	METODY OBSERWACYJNE

Rys. 9. Ogólna klasyfikacja metod oceny jakości użytkowej wg Whitefielda i in.

Źródło: Opracowano na podstawie prac: [12], [46].

Hom [19] podzielił metody oceny jakości ergonomicznej na trzy zasadnicze kategorie: (1) testowanie (ang. *testing*), np. protokół głośnego myślenia, śledzenie ruchów gałek ocznych; (2) inspekcja (ang. *inspection*) – oceny heurystyczne, listy kontrolne, wędrówka kognitywna; (3) dociekanie (ang. *inquiry*) – wywiady, ankiety, studia etnograficzne.

Z punktu widzenia rodzaju ocenianych cech jakości ergonomicznej można wyróżnić [8], [32]: (1) metody oceny cech obiektywnych, np. analiza danych zarejestrowanych podczas eksperymentów laboratoryjnych, KLM czy modele matematyczne symulujące funkcjonowanie oczu i rąk – SimEye i SimHand; (2) metody oceny wymiaru subiektywnego jakości użytkowej, przykładowo: AHP (ang. *Analityc Hierarchy Process*) – metoda analitycznego wyznaczania hierarchii porównywanych parami wariantów, wywiady, ankiety, listy kontrolne, (3) metody uwzględniające oceny obu wymiarów np. analizy conjoint.

Jedno z najnowszych i najbardziej kompleksowych podejść do pomiaru jakości użytkowej systemów interakcyjnych zostało zaproponowane przez Hornbæka [20]. Uwzględnia on generalnie dwa zasadnicze wymiary (rys. 10). Pierwszy wywodzi się z normy ISO 9241 [23] i uwzględnia zdefiniowane tam trzy wymiary: skuteczność, efektywność i satysfakcja. Drugi wymiar dzieli metody pomiaru na subiektywne i obiektywne.

	OBIEKTYWNE MIARY	SUBIEKTYWNE MIARY
WYNIKI (SKUTECZNOŚĆ)	Ocena ekspercka	Postrzeżenie wyników przez użytkownika
PROCES INTERAKCJI (EFEKTYWNOŚĆ)	Czas, wzorce użycia, łatwość uczenia	Subiektywnie odczuwalny czas trwania, obciążenie mentalne, postrzeżenie trudności zadania
NASTAWIENIE UŻYTKOWNIKÓW I DOŚWIADCZENIA (SATYSFAKcja)	Parametry fizjologiczne, reakcje odruchowe	Rzetelne i trafne kwestionariusze

Rys. 10. Ogólna klasyfikacja metod oceny jakości użytkowej wg Hornbæka
Źródło: Opracowano na podstawie pracy [20].

Inne klasyfikacje stosowanych metod oraz ich omówienie można znaleźć np. w pracach: [8], [12], [19], [25], [32]–[35], [42], [45], [46]. Zaprezentowane w tej części typologie i klasyfikacje bardzo się różnią. Ta różnorodność wynika zapewne w dużej mierze z braku jednorodnej definicji jakości użytkowej. W takim kontekście wybór odpowiednich metod oceny jakości użytkowej i wykorzystanie ich w poszczególnych etapach wytwarzania oprogramowania stanowi wciąż wyzwanie dla badaczy i praktyków. Opisane i cytowane powyżej podziały metod i technik oceny jakości użytkowej produktów informatycznych przedstawiają i porządkują oferowane przez HCI narzędzia, ułatwiając w ten sposób podejmowanie decyzji w tym zakresie.

5. Aktualne problemy i potencjalne kierunki badań

W oparciu o zaprezentowane materiały i cytowane prace, ta część artykułu została poświęcona identyfikacji zasadniczych problemów związanych z zapewnianiem wysokiej jakości oprogramowania. Przedstawione problemy stały się podstawą określenia możliwych kierunków działań, zmierzających do dalszego zwiększenia znaczenia jakości użytkowej w wytwarzaniu programów komputerowych. Wskazano również potencjalne obszary wymagające dalszych badań i szczególnego zainteresowania zarówno naukowców, jak i producentów oprogramowania.

Omówione w początkowej części tej pracy różnorodne definicje jednego z kluczowych pojęć w HCI – jakości użytkowej – pokazują, że dalsze prace nad ujednoczeniem nazewnictwa wydają się konieczne. Brak powszechnie akceptowanych terminów może z jednej strony utrudniać prowadzenie dyskursu naukowego i rozwój badań, a z drugiej stanowić poważną przeszkodę w porównywaniu poziomu jakości użytkowej konkurujących ze sobą programów komputerowych. Trwające obecnie prace nad opracowaniem kolejnych części norm ISO 9241, ISO 9126, ISO/DIS 13407 i pokrewnych, próby konsolidacji treści w nich zawartych oraz przygotowanie przewodników, pokazujących jakie są zależności między standardami i jak je stosować w praktyce, przypuszczalnie przyczynią się do poprawy sytuacji w tym zakresie.

Obserwowany w ostatnich latach dość spory wzrost liczby prac badawczych w zakresie jakości użytkowej oprogramowania wymaga również zwiększenia liczby opracowań przeglądowych, obejmujących i porządkujących coraz większą liczbę prac naukowych. Celowe także może okazać się tworzenie grup badawczych, ukierunkowanych na poszukiwanie kompleksowych rozwiązań wybranych kluczowych zagadnień i integrujących wyniki badań z różnych obszarów (podobny postulat został przedstawiony w pracy [39]). Do wyłonienia takich strategicznych kierunków należy również zaangażować przedstawicieli producentów oprogramowania, którzy pomogliby ustalić priorytety badań uwzględniające, w większym niż dotychczas stopniu, ich potrzeby. Wydaje się, że do takich priorytetowych kierunków można zaliczyć ocenę jakości użytkowej systemów informatycznych w długim okresie oraz badania relacji pomiędzy różnymi aspektami jakości użytkowej.

Jest rzeczą dość oczywistą, że jakość systemów interakcyjnych w sporym stopniu może zależeć od sposobu interakcji, który określa typ dialogu człowieka z komputerem [27]. W ostatnich latach oprócz podstawowych stylów obejmujących [4], [33], [45]: pytania i odpowiedzi, języki poleceń, wypełnianie formularzy, menu, manipulację bezpośrednią, język naturalny, pojawiły się zupełnie nowatorskie rozwiązania. Należą do nich między innymi [5], [17], [33], [47]: wszechobecne przetwarzanie (ang. *ubiquitous computing*), przenikające przetwarzanie (ang. *pervasive computing*), przetwarzanie zintegrowane z odzieżą (ang. *wearable computing* lub *wearables*), wirtualna rzeczywistość, wirtualne środowisko (ang. *virtual reality*, *virtual environment*), poszerzona rzeczywistość (ang. *augmented reality*), przetwarzanie wyrażające

emocje (ang. *affective computing*) itp. Okazuje się, że pomimo ogromnego postępu technicznego, umożliwiającego tworzenie coraz bardziej złożonych i zaawansowanych technologicznie systemów, przydatność nowych stylów interakcji jest w dużym stopniu ograniczona [17]. Uważa się także [47], że większość publikacji naukowych w dziedzinie współdziałania człowieka z komputerem dotyczy właśnie tych nowych problemów i sposobów ich rozwiązywania. Ignoruje się jednocześnie wiele potencjalnych obszarów badawczych i możliwości usprawnień modeli, metod i narzędzi związanych z tradycyjnymi sposobami komunikacji człowieka z maszyną cyfrową. Tymczasem dzisiejsze systemy informatyczne, wciąż jeszcze w większości zaopatrzone są w typowe graficzne interfejsy użytkownika, gdzie bardzo często użytkownik komunikuje się z systemem za pomocą stosunkowo prostych metod *manipulacji bezpośredniej* [39], [41], [43] (szczegółowe omówienie metody zamieszczono też w pracach [15], [48]). Nastawienie producentów i badaczy na nowości techniczne w znacznym stopniu ogranicza zainteresowanie badaniami o charakterze podstawowym, które mają ogromne znaczenie dla przeciętnych użytkowników popularnych programów komputerowych. Powyższe postulaty nie oznaczają naturalnie konieczności porzucenia badań nad nowymi sposobami interakcji. W kontekście wciąż szybkiego rozwoju techniki komputerowej kontynuacja badań w tym zakresie jest niewątpliwie niezbędna.

Wydaje się, że aktualnie prowadzone badania w zbyt małym stopniu uwzględniają zdobycze z innych, dynamicznie rozwijających się dziedzin nauki. Wykorzystanie w kolejnych pracach dorobku, uzyskanego na gruncie na przykład neurofizjologii czy neuropsychologii, może dać odpowiedź na wiele pytań nurtujących zarówno badaczy, jak i praktyków poruszających się w obszarze HCI. Podobne rezultaty może dać weryfikacja i ewentualna modyfikacja na gruncie HCI znanych w innych dziedzinach, szczególnie psychologii, modeli zachowania się człowieka. Analizując prace związane z badaniem jakości użytkowej systemów interakcyjnych, można odczuć także pewien niedosyt, związany z wykorzystaniem w niewystarczającym stopniu nowoczesnych metod i narzędzi badawczych, takich jak funkcjonalny rezonans magnetyczny (fMRI), pozytonowa emisyjna tomografia komputerowa (PET), urządzenia do śledzenia ruchów gałek ocznych (ang. *eye trackers*) itp. Zastosowanie tych narzędzi wiąże się najczęściej z dużą czasochłonnością prowadzenia badań i zaangażowaniem sporych środków finansowych, co zapewne ogranicza ich wykorzystanie w procesie produkcji oprogramowania. Niemniej jednak szersze stosowanie takiego sprzętu w celach naukowych z pewnością przyczyniłoby się do rozwoju badań nad zagadnieniami podstawowymi (np. mechanizmem wizualnego przetwarzania preatentywnego). Uzyskana w ten sposób wiedza o fundamentach zachowania się użytkownika w interakcji z maszynami cyfrowymi w konsekwencji mogłaby mieć wymierne znaczenie również dla praktyki wytwarzania systemów informatycznych.

W wielu publikacjach, głównie o charakterze empirycznym, zamieszczanych nawet w renomowanych czasopismach międzynarodowych, daje się zauważyć nie zawsze dostateczną dbałość o formalny warsztat metodologiczny. Jest to dość zaskakujące, biorąc pod uwagę łatwy dostęp do komputerowych pakietów statystycznych

zarówno komercyjnych, jak i darmowych. Zaniedbania w tym zakresie mogą prowadzić do wyciągania nieuprawnionych bądź fałszywych wniosków z uzyskanych danych eksperymentalnych. Dodatkowo, pomimo często wykorzystywanych metod subiektywnych do oceny jakości użytkowej, zdziwienie może budzić także niewielka liczba prac poświęconych rozwijaniu trafnych i rzetelnych kwestionariuszy oraz małe zainteresowanie ich standaryzacją (zostało to również zauważone w pracy [20]). W tym obszarze badacze i praktycy HCI powinni brać przykład z psychologii, gdzie odpowiednie standardy zostały już wypracowane.

Ewolucja poglądów naukowców na jakość systemów informatycznych przedstawiona w tej pracy powinna mieć swoje odzwierciedlenie praktyczne. Pomimo tego, że HCI należy zaliczyć do stosunkowo młodych dziedzin nauki, istnieje sporo metod, technik i narzędzi pozwalających na zwiększanie jakości użytkowej systemów interakcyjnych. Choć metody te dość często są niedoskonałe, to jednak stanowią one dość pokaźny arsenał walki z błędami projektowymi, które pojawiają się na wszystkich etapach wytwarzania oprogramowania.

Z punktu widzenia praktyki rozwój badań naukowych powinien doprowadzić do powstania (przygotowania) powszechnie akceptowanych narzędzi testowych, takich jak listy kontrolne, heurystyki czy ujednoczone procedury postępowania przy produkcji oprogramowania. Podobnie jak dla naukowców, również dla producentów systemów informatycznych istotne jest stosowanie wysokiej klasy kwestionariuszy subiektywnych, odpowiednio zweryfikowanych procedurami statystycznymi. Obecnie niestety trudno się doszukać tego rodzaju kwestionariuszy w języku polskim, a w języku angielskim istnieje zaledwie kilka takich pozycji. Ważne dla podniesienia jakości produkowanych programów jest też zachęcanie producentów do powszechnego korzystania z istniejących norm międzynarodowych z serii ISO 9241 i 9126 i wkomponowanie zaleceń z nich wynikających w cykle życia oprogramowania. Działania takie mogą być prowadzone w ramach wdrażania zaawansowanych, całościowych modeli inżynierii jakości użytkowej zorientowanych na końcowego użytkownika, na przykład zgodnie z podejściem zaproponowanym przez Mayhew [26].

Wśród problemów związanych z wykorzystaniem w praktyce badań z dziedziny HCI duże znaczenie ma niewątpliwie sposób wykorzystania metod oceny jakości użytkowej na różnych etapach wytwarzania oprogramowania. Kwestie integracji stosowanych metod ewaluacji jakości użytkowej z odpowiednimi fazami tworzenia produktów interakcyjnych (porównaj też [21]) mogą powodować poważne problemy. Nierzadko zdarzają się sytuacje, w których proces projektowania interakcji i przeprowadzanie ocen jakości użytkowej są w dużej mierze niezależne od siebie. W konsekwencji prowadzi to do tego, że wyniki przeprowadzonych analiz jakości użytkowej nie mają większego wpływu na proces wytwarzania oprogramowania. Coraz większa presja na jak najszybsze wprowadzanie produktów na rynek, a także *outsourcing* w zakresie tworzenia kodów źródłowych nie sprzyjają też integracji ocen jakości użytkowej z etapami cykli życia systemów informatycznych.

Tabela 1. Potencjalne kierunki badań i działań związane z poprawą jakości użytkowej systemów informatycznych

Perspektywa naukowa	Perspektywa użytkowa
<ul style="list-style-type: none"> • Tworzenie nowych i udoskonalanie istniejących metod oceny jakości użytkowej. • Kontynuacja badań nad nowymi rozwiązaniami technologicznymi i nowymi sposobami interakcji włączając w to interakcje w przestrzeni trójwymiarowej oraz interakcje wykorzystujące nowe urządzenia wejścia-wyjścia. • Prace nad ulepszeniem istniejących rozwiązań w zakresie jakości systemów interakcyjnych. • Zwiększenie liczby prac przeglądowych, obejmujących i porządkujących coraz większą liczbę prac naukowych z dziedziny HCI. • Prace nad ujednoczeniem nazewnictwa w dziedzinie HCI. Brak powszechnie akceptowanych terminów może utrudniać porozumiewanie się i rozwój badań. • Prace nad przygotowaniem wspólnej normy międzynarodowej, stanowiącej kompromis między zapisami w normach 9241 i 9126. • Weryfikacja na gruncie HCI znanych w innych dziedzinach, szczególnie psychologii, modeli zachowania się człowieka. • Uwzględnianie w kolejnych badaniach w większym stopniu dorobku z innych dziedzin nauki, w szczególności z dynamicznie rozwijających się działów medycyny związanych z fizjologią funkcjonowania mózgu. • Szersze korzystanie z nowoczesnych metod i narzędzi badawczych, takich jak fMRI, PET, urządzeń do śledzenia ruchów gałek ocznych. • Rozwój badań nad zagadnieniami podstawowymi, np. mechanizmem przetwarzania preatentywnego. • Tworzenie grup badawczych ukierunkowanych na poszukiwanie kompleksowych rozwiązań wybranych zagadnień i integrujących wyniki badań z różnych obszarów. • Zwiększenie zaangażowania w badaniach naukowych doświadczonych praktyków z firm produkujących i wdrażających systemy informatyczne. • Poszukiwanie sposobów integracji metod oceny jakości użytkowej z odpowiednimi fazami tworzenia oprogramowania. 	<ul style="list-style-type: none"> • Działania propagujące korzystanie z istniejących standardów międzynarodowych, związanych z zapewnianiem jakości użytkowej (np. ISO 9241 i 9126). <ul style="list-style-type: none"> – Przetłumaczenie norm na język polski. – Udostępnienie opracowań opisujących rozwiązania sugerowane przez normy. – Wkomponowanie zaleceń z nich wynikających w cykle życia oprogramowania. • Przygotowanie podręczników przedstawiających zarówno zalecenia zwiększające jakość użytkową systemów informatycznych, jak i podstawowe metody ocen. • Przetłumaczenie na język polski podstawowych i uznanych na świecie podręczników z dziedziny HCI. • Opracowanie w języku polskim powszechnie znanych i wykorzystywanych list kontrolnych, heurystyk. • Wdrażanie zaawansowanych, całościowych modeli inżynierii jakości użytkowej, zorientowanych na końcowego użytkownika • Uczestnictwo naukowców w komercyjnych projektach wykorzystujących w praktyce zdobycze nauki. • Zachęcanie do brania udziału praktyków w badaniach o charakterze naukowym. • Uczestnictwo przedstawicieli firm informatycznych w szkoleniach i warsztatach. • Zwiększenie zajęć z interakcji człowieka z komputerem na studiach o profilu informatycznym. • Stworzenie bazy danych ośrodków i naukowców zajmujących się tematyką i udostępnienie oraz rozpowszechnianie jej w przedsiębiorstwach z branży technologii informatycznych. • Popularyzowanie wiedzy o jakości użytkowej, np. udział w festiwalach nauki, wypowiedzi do prasy, organizacja szkoleń i warsztatów, organizacja konferencji. • Wykorzystanie środków Unii Europejskiej na wspólne inicjatywy naukowców i przedsiębiorstw, mające na celu podnoszenie jakości użytkowej oprogramowania.

W tabeli 1 przedstawiono w zwięzłej formie kierunki badań, zasygnalizowane w tej części pracy. Zestawienie to zawiera również potencjalne działania naukowców

i praktyków, które mogą mieć znaczenie przy zapewnianiu jakości użytkowej w procesie wytwarzania oprogramowania.

Poprawa jakości użytkowej oprogramowania wymaga szerokiej współpracy pomiędzy firmami produkującymi i wdrażającymi systemy informatyczne z badaczami. Z jednej strony należy zachęcać praktyków do brania udziału w projektach o charakterze naukowym. Z drugiej natomiast warto uświadamiać naukowcom konieczność uwzględniania w większym stopniu w swoich priorytetach realnych potrzeb producentów i użytkowników oprogramowania.

6. Podsumowanie

W pracy dokonano przeglądu zasadniczych podejść do wytwarzania oprogramowania, zaczynając od klasycznych modeli cyklu życia produktów informatycznych i kończąc na współczesnych koncepcjach silnie zorientowanych na końcowego użytkownika. Choć trudno nazwać ten przegląd wyczerpującym, przedstawiony materiał wyraźnie ilustruje zmianę tendencji w wytwarzaniu oprogramowania. Omówione w tej pracy nowoczesne modele uwzględniają, w dużo większym stopniu niż podejścia klasyczne, konieczność zapewnienia wysokiej jakości użytkowej na różnych etapach projektowania systemów informatycznych.

Na podstawie zaprezentowanego materiału dokonano analizy i zestawienia potencjalnych problemów, wynikających z aktualnego stanu wiedzy i omówiono przyszłe kierunki badań. Rozważania zostały podzielone na dwa obszary: jeden związany z pracą naukową i drugi odnoszący się do praktycznego wykorzystania dorobku z dziedziny interakcji człowieka z komputerem. W kontekście badań naukowych wskazano na konieczność zwiększenia ilości prac o charakterze porządkującym, mających na celu ujednoczenie nomenklatury, systematyzację wyników badań oraz ukierunkowanie grup badawczych na kluczowe dla HCI zagadnienia. Zwrócono również uwagę na konieczność zwiększenia zainteresowania badaniami, których celem jest poprawianie metod i narzędzi z zakresu klasycznych stylów interakcji, ulepszanie subiektywnych kwestionariuszy poprzez ich weryfikację i standaryzację. Odnotowano także potrzebę intensyfikacji prac nad badaniem zależności pomiędzy różnymi wymiarami i atrybutami jakości użytkowej oraz dbania o stosowanie odpowiedniego warsztatu badawczego. Do przyszłościowych, cieszących się obecnie niewielkim zainteresowaniem obszarów badawczych, zaliczono prace podstawowe, wykorzystujące nowoczesne narzędzia badawcze (fMRI, PET, eye trackers itp.), prace mające na celu przenoszenie modeli, narzędzi i technik z innych dziedzin na grunt HCI wraz z weryfikacją ich przydatności oraz ewentualnym dostosowaniem do specyfiki tej dyscypliny.

W aspekcie praktycznym zwrócono uwagę na konieczność opracowania i rozpowszechniania standardowych narzędzi do monitorowania i porównywania poziomu

jakości użytkowej. Wskazano również na konieczność prowadzenia działań promujących korzystanie z norm międzynarodowych i zachęcających do wdrażania kompleksowych podejść w celu zapewniania wysokiej jakości użytkowej na wszystkich etapach wytwarzania produktów informatycznych. Zastosowanie w praktyce modeli wytwarzania oprogramowania zorientowanych na końcowego użytkownika jest możliwe poprzez właściwe wykorzystanie wiedzy i zastosowanie adekwatnych metod oceny oraz dzięki ich odpowiedniemu zintegrowaniu z fazami projektowania interakcji. Powyższe postulaty wymagają oczywiście dwustronnej i daleko idącej współpracy między badaczami i praktykami.

Bibliografia

- [1] BEVAN N., *Quality in use: Meeting user needs for quality*, The Journal of Systems and Software, 1999, 49, s. 89–96.
- [2] BEVAN N., *International standards for HCI and usability*, International Journal of Human-Computer Studies, 2001, 55, s. 533–552.
- [3] BOEHM B., *A spiral model for software development enhancement*, Computer, 1988, 20(5), s. 61–72.
- [4] BURNS J.T., *Współdziałanie człowieka z komputerem* [w:] J. Górski, *Inżynieria oprogramowania w projekcie informatycznym*, MIKOM, Warszawa 2000, s. 155–184.
- [5] CHRISTOU G., JACOB R.J.K., *Evaluating and comparing interaction styles* [w:] J.A. Jorge, N.J. Nunes, J.F. Cunha (eds.), DSV-IS, LNCS, Berlin–Heidelberg, Springer-Verlag, 2003, s. 406–409.
- [6] COLLINS D., *Designing object-oriented user interfaces*, Redwood: Benjamin Cummings, 1994.
- [7] Computerworld, *Power Mac 601 – szansa Apple’a*, Computerworld, 02/1995.
- [8] DIX A., FINLAY J., ABOWD G.D., BEALE R., *Human-Computer Interaction*, 3rd edition, Harlow: Pearson Education, 2004.
- [9] EASON K.D., *Ergonomic perspectives on advances in human-computer interaction*, Ergonomics, 1991, 34(6), s. 721–741.
- [10] FLOWERS J., *Usability*, <http://www.bsu.edu/web/jcflowers1/rlo/usability.htm>, 2004-11-17.
- [11] FOLMER E., BOSCH J., *Architecting for usability: a survey*, The Journal of Systems and Software, 2004, 70, s. 61–78.
- [12] FU L., SCHMIDT K., *Usability Evaluation* [w:] W. Karwowski (red.), *International Encyclopedia of Ergonomics and Human Factors*, Taylor & Francis, London 2001, s. 1187–1190.
- [13] GASIK S., *Jakość w projektach informatycznych*, <http://sgasik.webpark.pl/jakosc.htm>, 2004-11-17.
- [14] GÖRANSSON B., LIF M., GULLIKSEN J., *Usability design – extending rational unified process with a new discipline* [w:] J.A. Jorge, N.J. Nunes, J.F. Cunha (eds.), DSV-IS, LNCS, Springer-Verlag, Berlin–Heidelberg 2003, s. 316–330.
- [15] GÖRNER C., VOSSEN P., ZIEGLER J., *Direct manipulation user interface* [w:] M. Galer, S. Harker, J. Ziegler (red.), *Methods and tools in user-centred design for information technology*, North-Holland, Elsevier Science Publishers B.V., 1992.
- [16] GOULD J.D., BOIES S.J., LEWIS C.H., *Designing for usability – key principles and what designers think*, Communications of the Association for Computing Machinery, 1985, 28(3), s. 300–311.
- [17] HARTSON H.R., *Human-computer interaction: interdisciplinary roots and trends*, The Journal of Systems and Software, 1998, 43, s. 103–118.

- [18] HARTSON H.R., HIX D., *Human-computer interface development: concepts and systems for its management*, ACM Computing Surveys (CSUR), 1989, 21(1), s. 5–92.
- [19] HOM J., 1998, *The usability methods toolbox*, <http://jthom.best.vwh.net/usability>, 2004-11-19.
- [20] HORNBÆK K., *Current practice in measuring usability: Challenges to usability studies and research*, International Journal of Human – Computer Studies, 2006, 64(2), s. 79–102.
- [21] HORNBÆK K., STAGE J., *The interplay between usability evaluation and user interaction design*, International Journal of Human – Computer Interaction, 2006, 21(2), 117–123.
- [22] *International standards for HCI and usability* (2003), http://www.usabilitynet.org/tools/r_international.htm, 2004-11-17.
- [23] ISO 9241, 1998, *Ergonomic requirements for office work with visual display terminals (VDTs)*, Part 11. Guidance on usability, International Standard.
- [24] ISO/DIS 13407, 1996, *Human-Centred Design Processes for Interactive Systems*, Draft International Standard.
- [25] LEWIS C., RIEMAN J., *Task-centered user interface design. A practical introduction*, http://pages.cpsc.ucalgary.ca/~saul/hci_topics/papers/LewisRiemanBook/, 2004-11-16.
- [26] MAYHEW D.J., *The usability engineering lifecycle*, Academic Press, San Diego 1999.
- [27] MCDUGALL S.J.P., OBORNE D.J., *Human-machine systems: written and symbolic communication* [w:] W. Karwowski (red.), *International Encyclopedia of Ergonomics and Human Factors*, Taylor & Francis, London 2001, s. 125–127.
- [28] MICHALSKI R., *Komputerowe wspomaganie badań jakości ergonomicznej oprogramowania*, praca doktorska, Politechnika Wrocławska, 2005.
- [29] NIELSEN J., *Usability Engineering*, Academic Press, New York 1993.
- [30] NIELSEN J., 2003, *Usability 101: introduction to usability*, <http://www.useit.com/alertbox/20030825.html>, 2004-11-17.
- [31] O'NEILL E., *User-developer cooperation in software development: building common ground and usable systems*, Springer-Verlag, London 2001.
- [32] PARK K.S., LIM C.H., *A structured methodology for comparative evaluation of user interface designs using usability criteria and measures*, International Journal of Industrial Ergonomics, 1999, 23, s. 379–389.
- [33] PREECE J., ROGERS Y., SHARP H., *Interaction design: beyond human-computer interaction*, John Wiley & Sons, New York 2002.
- [34] RAUTENBERG M., 1996, *Usability engineering*, ETH, Swiss Federal Institute of Technology, <http://www.zie.pg.gda.pl/~msik/ergonomia-si/ergonomia-si-materialy.htm>, 2004-11-19.
- [35] RAUTENBERG M., 1998, *Introduction into human-computer interaction*, Zürich: Swiss Federal Institute of Technology, <http://www.zie.pg.gda.pl/~msik/ergonomia-si/ergonomia-si-materialy.htm>, 2004-11-19.
- [36] REDMILL F., *Zarządzanie projektem* [w:] J. Górski, *Inżynieria oprogramowania w projekcie informatycznym*, MIKOM, Warszawa 2000, s. 19–51.
- [37] REED P., HOLDAWAY K., ISENSEE S., BUIE E., FOX J., WILLIAMS J. LUND A., *User interface guidelines and standards: progress, issues, and prospects*, Interacting with Computers, 1999, 12, s. 119–142.
- [38] RILEY V.A., *Functional systems design versus interface design* [w:] W. Karwowski (red.), *International Encyclopedia of Ergonomics and Human Factors*, Taylor & Francis, London 2001, s. 696–697.
- [39] SHACKEL B., *People and computers – some recent highlights*, Applied Ergonomics, 2000, 31, s. 595–608.
- [40] SHNEIDERMAN B., *The future of interactive systems and the emergence of direct manipulation*, Behaviour and Information Technology, 1982, 1, s. 237–256.
- [41] SHNEIDERMAN B., *Direct manipulation. A step beyond programming languages*, IEE Computer, 1983, 16, s. 57–69.

- [42] SIKORSKI M., *Zarządzanie jakością użytkową w przedsiębiorstwach informatycznych*, Wydawnictwo Politechniki Gdańskiej, Gdańsk 2000b.
- [43] SMITH D.C., IRBY C., KIMBALL R., VERPLANK B., HARSLEM E., *Designing the Star user interface* [w:] P. Degano, S. Sandewall (red.), *Integrated Interactive Computing Systems*, North-Holland, Amsterdam 1983.
- [44] STEWART T., STEWART T.F.M., *Human-computer interaction (HCI) standards* [w:] W. Karwowski (red.), *International Encyclopedia of Ergonomics and Human Factors*, Taylor & Francis, London 2001, s. 676–679.
- [45] TORRES R.J., *User interface design and development*, Upper Saddle River, Prentice Hall, PTR, 2002.
- [46] WHITEFIELD A., WILSON F., DOWELL J., *A framework for human factors evaluation*, Behavior and Information Technology, 1991, 10, s. 65–79.
- [47] WHITTAKER S., TERVEEN L., NARDI B.A., *A reference task agenda for HCI*, Human-Computer Interaction, 2000, 15(2/3).
- [48] ZIEGLER J.E., FAHRNICH K.P., *Direct manipulation* [w:] M. Helander (red.), *Handbook of Human-Computer Interaction*, North-Holland, Elsevier Science Publishers B.V., 1988.

Usability in software manufacturing process

The paper presents a review of information system usability assurance approaches. The author describes a historical development of software production models and divides them into two main groups. The first one, named classical, included fundamental computer program life cycles derived from software engineering. Waterfall, spiral and rapid application development (RAD) models were among approaches presented here. The second group comprised software manufacturing models oriented towards end users, in which particular emphasis is placed on usability. Within the confines of this group, the user-centered software design approach, recommended by the ISO/DIS 13407, was described first. Next, the author characterises star, Collins, simplified interaction design model proposed by Preece, Rogers and Sharp as well as comprehensive approach prepared by Mayhev. For the sake of big evaluation significance in individual life cycle stages, the paper contains various classifications and typologies of software assessment methods. Additionally, the author describes current research trends aimed at developing user-friendly computer programs and he identifies existing and possible problems in this area.

Keywords: *informatics products' ergonomics, human-computer interaction, software life cycles, information technology systems*