

**This paper should be cited as:** Bobkowska, A., & Grabiec, K. (2007). Integracja technik użyteczności i technik inżynierii oprogramowania w projekcie informatycznym. *Proceedings of the Conference: Interfejs użytkownika - Kansei w praktyce, Warszawa 2007* (pp. 84–95). Warsaw: Wydawnictwo PJWSTK.

# Anna Bobkowska, Kamila Grabiec

## Integracja technik użyteczności i technik inżynierii oprogramowania w projekcie informatycznym

Politechnika Gdańska,  
Narutowicza 11/12,  
80-952 Gdańsk

- Zagadnienie integracji technik użyteczności i technik inżynierii oprogramowania wynika z obserwacji braku spójności terminologicznej i braku dopasowania metodologicznego technik, które mają zastosowanie do tego samego „bytu” – do oprogramowania i są stosowane w jednym przedsięwzięciu – w projekcie informatycznym.
- W literaturze międzynarodowej można znaleźć wiele przyczyn takiego rozdzielania. Od lat siedemdziesiątych, techniki z obu grup były opracowywane przez niezwiązane ze sobą grupy badaczy, prezentowane były zazwyczaj na niezależnych konferencjach, wypracowały swoje własne, niezależne modele i terminologię. Obecnie nauczane są na osobnych przedmiotach, a w programach nauczania nie ma elementów łączących te dwie dziedziny. Dodatkowo obserwuje się zjawisko niedoceniań przez poszczególne grupy badaczy, wyników i metod wypracowanych przez tę drugą grupę. Wskazywanych

jest wiele różnic pomiędzy metodami tych dyscyplin. Pierwsza różnica dotyczy aspektu jakości, na którym się koncentrują. Przedmiotem zainteresowania inżynierii oprogramowania są aspekty techniczne i zarządzania projektem. Natomiast projektowanie interfejsów użytkownika lub użyteczność zajmuje się zapewnieniem odpowiedniego interfejsu użytkownika w oprogramowaniu oraz kwestiami ergonomii, dopasowania do zadań, wyglądu i satysfakcji z korzystania z systemów. Kolejna różnica dotyczy potrzebnych zdolności i kompetencji do wykonywania zadań z obu grup. Zazwyczaj wśród potrzebnych zdolności przypisywanych do inżynierii oprogramowania wymieniane są umiejętności ścisłego

i abstrakcyjnego myślenia<sup>1</sup>, natomiast wśród wymagań dla projektantów interfejsu użytkownika - zdolności psychologiczne, socjologiczne, estetyczne itd.

- Jednak z drugiej strony, inżynieria oprogramowania oraz projektowanie interfejsów użytkownika reprezentują uzupełniające się podejścia do jakości oprogramowania. Obydwie te dziedziny mają związek z oprogramowaniem, a ich zadania muszą być wykonywane w ramach jednego projektu informatycznego. Zazwyczaj nie jest możliwe wykonanie prac technicznych, a potem 'dodanie' jakości użytkowej lub zaprojektowanie kompletnego interfejsu użytkownika, a następnie tylko dopisanie części implementacyjnej. Dodatkowo można zauważyć ścisłe powiązania pomiędzy nimi. Elementy jakości technicznej są składową jakości użytkowej. Poprawne działanie systemu, jego wydajność wpływają na skuteczność wykonywania zadań i satysfakcję użytkownika. Z drugiej strony, system zoptymalizowany technicznie, którego użytkownik „nie potrafi używać” lub który go „nie satysfakcjonuje” – też nie jest przydatny lub nie spełnia swojej roli. W przypadku możliwości wyboru przez użytkownika innego systemu, system o niskiej użyteczności może nie być w ogóle używany.
- Warto też przyjrzeć się temu zagadnieniu z perspektywy efektywności prowadzenia prac projektowych w ramach uwarunkowań projektu informatycznego, do których zazwyczaj zaliczyć trzeba krótki czas realizacji, nienadmiarowy budżet i wysokie wymagania jakościowe. Nie ma czasu i innych zasobów na wykonywanie nadmiarowych prac. Tutaj warto zauważyć, że są takie zadania w projekcie, które będąc na pograniczu inżynierii oprogramowania i użyteczności, polegają na wykonaniu podobnych lub powiązanych prac, np. opis organizacji, opis zadań, opis sposobów wykonywania zadań etc. Skutkami nieskoordynowanego działania może być wykonywanie tych zadań podwójnie przez różne grupy osób, rozbieżności w otrzymanych opisach i ich konsekwencje ujawniające się podczas integracji systemu oraz marnowanie czasu i denerwowanie przedstawicieli użytkowników. Optymalnym rozwiązaniem jest więc konfiguracja procesu wytwarzania z zastosowaniem odpowiednich technik w konkretnym projekcie oraz odpowiednia koordynacja działań z obu obszarów.
- Celem tego referatu jest przedstawienie zarysu wyników badań międzynarodowych z obszaru integracji technik użyteczności i technik inżynierii oprogramowania oraz prezenta-

---

<sup>1</sup> Jest to tradycyjny sposób myślenia, w rzeczywistości wiele zadań inżynierii oprogramowania wymaga zdolności o charakterze społecznym. W szczególności zadania analityków i kierowników projektów wymagają znajomości elementów psychologii stosowanej, np. umiejętności komunikacji. Dla wszystkich informatyków istotna jest umiejętność pracy grupowej.

cja wyników badań ankietowych, których celem było zbadanie, jak wygląda pod tym względem sytuacja w Polsce. W kolejnych sekcjach opisany został zarys nurtu integracji dyscyplin inżynierii oprogramowania i użyteczności, opis badań ankietowych oraz wyniki tych badań.

#### > Nurt integracji dyscyplin inżynierii oprogramowania i użyteczności

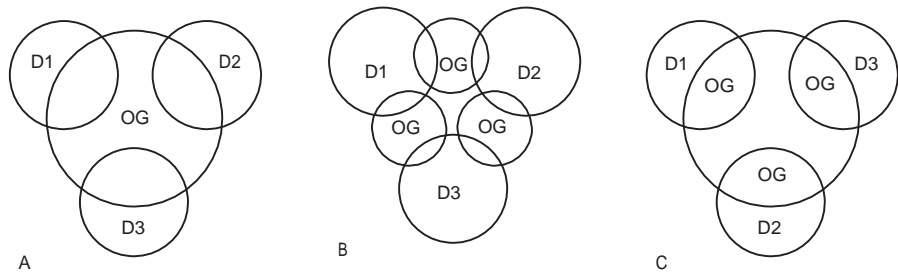
> Zintensyfikowane działania w nurcie badań dotyczących integracji dyscyplin inżynierii oprogramowania i użyteczności (ang. Bridging the SE & HCI Communities) [3] odbywały się w latach 2003-2005. Polegały one na zorganizowaniu serii sesji poświęconych tym zagadnieniom na konferencjach zarówno inżynierii oprogramowania, jak i projektowania interfejsów użytkownika. Uwieńczono je wydaniem książki. Organizatorzy rozpoczęli prace od identyfikacji różnic i problemów na pograniczu obu dyscyplin z udziałem przedstawicieli środowisk naukowych oraz reprezentantów firm informatycznych. Celem była integracja na poziomie procesu, celów technicznych i „miękkich”, architektury, notacji i środowisk. W późniejszym czasie większość prac dotyczyła obiektów granicznych oraz integracji technik z obu grup w procesie wytwarzania oprogramowania.

> Wyróżnione zostały następujące aspekty tego problemu [2]:

- Różnice na poziomie dyscypliny – wynikające z tego, że poszczególne jednostki rozwijają swoją wiedzę bez komunikacji z przedstawicielami innych dyscyplin i z biegiem czasu utwierdzają się w swoich przekonaniach i zamykają na wiedzę spoza swojej dyscypliny;
- Różnice na poziomie praktycznym – pojawiają się w projekcie, gdy występuje potrzeba przekazywania informacji pomiędzy osobami posiadającymi różne specjalizacje i są problemy w komunikacji.

#### > Różnice na poziomie dyscypliny powodują różnice na poziomie praktycznym.

> Najciekawszym osiągnięciem tego nurtu jest wprowadzenie pojęcia obiektu granicznego. Jest to pewna abstrakcja wspólnej przestrzeni pojęć, wspólnej wiedzy, wspólnych produktów i założeń, które mogą służyć do integracji i koordynacji działań osób lub zespołów posiadających różne specjalizacje. Można więc zadać pytanie o naturę tych obiektów granicznych [1]. Na rysunku 1 przedstawiono teoretyczne modele obiektów granicznych. W przypadku A jest to jedna wspólna przestrzeń pojęć dzielona przez ekspertów z różnych dziedzin, w przypadku B – wspólna wiedza par ekspertów, natomiast przypadek C wskazuje na istnienie pewnej wspólnej wiedzy dla wszystkich, np. wiedzy dziedzinowej, jak również pewnych elementów wiedzy specyficznej tylko dla poszczególnych specjalizacji.



Rysunek 1. Modele obiektów granicznych, oznaczenia: OG – obiekt graniczny, D1, D2, D3 – różne dyscypliny, np. inżynieria oprogramowania, użyteczność, wiedza na temat dziedziny problemu, bezpieczeństwo, aspekty biznesowe, wydajność itd.

- Zdaniem praktyków najważniejszym obiektem granicznym są kierownicy zespołów i stosowane przez nich podejście. Zwrócono uwagę na pozytywne skutki postawy ułatwiającej wykonywanie zadań poszczególnym grupom bez zbytej ingerencji w sprawy merytoryczne oraz na to, że odpowiedni dobór osób na tym stanowisku może wyeliminować potrzebę innych obiektów granicznych. W grupie ludzi wymieniane były także osoby przeszkolone w zakresie obu dziedzin. Zwracano szczególną uwagę na następujące umiejętności: umiejętność zrozumienia perspektywy innych, umiejętność wykorzystania wykonywanych przez innych produktów, umiejętność utworzenia płaszczyzny porozumienia oraz wspólnej przestrzeni działania.
- Drugim rodzajem istotnych obiektów granicznych jest zintegrowany proces wytwarzania oprogramowania, przy czym szczególną rolę pełni proces komunikacji i podejmowania decyzji. W ramach tego procesu można wykorzystać spotkania, dyskusje idei, sposobów postępowania i zawartości dokumentów, „burze mózgów”, e-maile itd. Na kolejnym miejscu znalazły się różne specyfikacje i dokumenty: prototypy interfejsów, przypadki użycia i scenariusze, specyfikacje, raporty, prezentacje oraz dokumentacja przeglądów funkcjonalności i użyteczności.
- Zaproponowano również kilka procesów integrujących, np.:
  - integrację użyteczności i inżynierii wymagań [5], w którym wyodrębniono decyzje oraz opis na poziomie zadań, dziedziny, interakcji oraz systemu;
  - metodę włączania technik użyteczności do procesu wytwarzania oprogramowania [4] z dopasowaniem terminologii do słownictwa używanego w inżynierii oprogramowania;
- próbę rozszerzenia RUP (Rational Unified Process) o opis ról, zadań i artefaktów związanych z użytecznością [7];

- podejście do koordynacji działań z obu dyscyplin z przekazywaniem odpowiednich specyfikacji [6].
- Zawierają one wiele ciekawych idei, jednak zazwyczaj są to rozwiązania fragmentaryczne i trudne do ujęcia w jeden zintegrowany proces. Warto też wspomnieć o włączeniu informacji o projektowaniu ukierunkowanym na użytkownika (ang. User-centered design) do specyfikacji RUP) [8].

#### ➤ Badania ankietowe w polskich firmach informatycznych

- Celem badań ankietowych jest stwierdzenie, jak w polskich firmach informatycznych wygląda sytuacja zastosowania metod użyteczności oraz ich integracji z metodami inżynierii oprogramowania. W szczególności interesujące dla nas są odpowiedzi na następujące pytania: W jakim stopniu stosowane są metody użyteczności? Czy występuje problem integracji metod użyteczności i metod inżynierii oprogramowania? Jakie sposoby integracji są stosowane? Jaka jest efektywność i skuteczność stosowanych rozwiązań? Czy występuje potrzeba dalszych badań w tym obszarze? Czy występuje potrzeba szkolenia studentów w tym obszarze i jakie kompetencje powinni oni posiadać?
- Badanie ma charakter eksploracyjny. Opracowana ankieta zawiera część wstępną, badającą kontekst zastosowania, oraz ankietę właściwą, która została opracowana w czterech wariantach, w zależności od stopnia zastosowania metod użyteczności i metod inżynierii oprogramowania.
- Cechą charakterystyczną badania rzeczywistości projektowej w inżynierii oprogramowania pod względem występujących problemów i skuteczności stosowanych technik jest problem pozyskania odpowiednio dużej ilości danych do analiz. Wynika on, przede wszystkim, z niepowtarzalności projektów informatycznych, dużej liczby charakterystyk i uwarunkowań wpływających na realizację projektu oraz wpływu przyjętej konfiguracji procesu wytwarzania na skuteczność poszczególnych technik. Dlatego wśród zaleceń empirycznej inżynierii oprogramowania, pojawia się postulat opisu kontekstu pozyskiwania danych empirycznych. W części wstępnej ankiety zbierane były następujące dane na temat kontekstu: wielkość firmy, wielkość realizowanych projektów, rodzaj wytwarzanych systemów oraz rola interfejsu użytkownika (mała, gdy oprogramowanie nie posiada interfejsu użytkownika albo jest on bardzo ubogi - do dużej, gdy interfejs użytkownika jest istotny). Ta część zawiera również pytania dotyczące zastosowania technik inżynierii oprogramowania i technik użyteczności (fakt zastosowania poszczególnych technik i stopień formalności) oraz pytanie o współpracę z grafikami komputerowymi.

- W zależności od odpowiedzi na pytania z części wstępnej, respondenci wypełniali jeden z czterech wariantów ankiety właściwej:
  - z zastosowaniem zarówno technik użyteczności, jak i technik inżynierii oprogramowania – wariant najbardziej dla nas interesujący, gdyż tylko wtedy można mówić o integracji, gdy są stosowane techniki z obu grup;
  - z zastosowaniem technik inżynierii oprogramowania i bez zastosowania technik użyteczności – wariant, który miał na celu stwierdzenie, czy występują wymieniane w literaturze problemy wynikające z braku zastosowania metod użyteczności i jakie są przyczyny braku zastosowania tych technik;
  - z zastosowaniem technik użyteczności i bez zastosowania technik inżynierii oprogramowania – wariant, który miał na celu stwierdzenie, czy występują w tych firmach problemy, które mają być rozwiązywane przez zastosowanie technik inżynierii oprogramowania;
  - bez zastosowania jakichkolwiek technik z obu tych grup – wariant, który miał na celu stwierdzenie, czy istnieją takie firmy i jakie w nich występują problemy.
- Ankiety zawierają pytania „otwarte” dotyczące stosowanego podejścia, oceny skuteczności poszczególnych rozwiązań itp. oraz pytania z możliwością wyboru z listy potencjalnych odpowiedzi, które zostały sformułowane na podstawie literatury oraz ogólnej wiedzy na temat uwarunkowań projektu informatycznego. Wariant ankiety z zastosowaniem zarówno metod użyteczności jak i metod inżynierii oprogramowania pokrywa szeroki zakres zagadnień: od stosowanych technik, celów ich wprowadzenia, ich umiejscowienia w procesie wytwarzania oprogramowania, poprzez ocenę ich skuteczności, sposoby integracji, tzw. obiekty graniczne, oraz powiązania w procesie wytwarzania oprogramowania, aż po zagadnienia wdrażania takich technik. Wszystkie warianty ankiety zawierają pytania o prace nad wdrażaniem nowych technik użyteczności, potrzebę badań i opracowań naukowych oraz przydatne kompetencje absolwentów informatyki w tym obszarze.

#### ➤ Wyniki badań ankietowych

- Prezentowane wyniki stanowią uwieńczenie badań pilotowych i zostały opracowane na podstawie siedemnastu ankiet pozyskanych w firmach o bardzo zróżnicowanej charakterystyce. W związku z tym wstrzymujemy się z podawaniem szczegółowych statystyk. Jednak już na podstawie tej próbki zauważalne są dość wyraźne tendencje. W dalszej części opracowania słowa „często”, „zdecydowana większość”, „najczęściej” oznaczają orientacyjnie 75% lub więcej wystąpień w badanej grupie, osobną kategorię w klasyfika-

cji stanowi „około połowa badanych przypadków”, natomiast słowa „rzadko”, „czasami” należy interpretować jako 25% lub mniej badanych przypadków.

- W dużych firmach informatycznych najczęściej wypełniany był wariant ankiety z zastosowaniem zarówno technik inżynierii oprogramowania jak i technik użyteczności, przy czym stopień zaawansowania i formalności metod inżynierii oprogramowania był większy niż metod użyteczności. Rola interfejsu w badanych przypadkach określana była jako duża lub średnia. Rzadko w dużych firmach są stosowane techniki inżynierii oprogramowania i nie są stosowane techniki użyteczności. W kilku firmach, w których realizowane są zróżnicowane projekty, w zależności od rodzaju projektu dobierane są odpowiednie techniki inżynierii oprogramowania i techniki użyteczności.
- Z badań wynika, że często są stosowane następujące techniki użyteczności:
  - prototypowanie,
  - testy z udziałem użytkowników,
  - analiza zadań użytkowników.
- Są to techniki, które są popularne w inżynierii oprogramowania i znajdują się w podręcznikach i programach nauczania inżynierii oprogramowania. Prototypowanie interfejsu użytkownika jest znane jako uniwersalny sposób na projektowanie i weryfikację wymagań użytkownika dotyczących interfejsu użytkownika oraz funkcjonalności oprogramowania. Testy z udziałem użytkowników są jednym z rodzajów testów, np. testy akceptacyjne, testy „beta”, a analiza zadań użytkowników jest związana z rozwijającym się nurtem modelowania biznesowego.
- W około połowie badanych dużych firm stosowane były również:
  - wywiady i/lub wywiady grupowe,
  - scenariusze użycia,
  - przewodniki stylu,
  - zalecenia projektowe,
  - planowanie użyteczności.
- Również w tej grupie można zauważyć powiązania z technikami inżynierii oprogramowania. Wywiady z klientem/użytkownikiem są stosowane w celu pozyskania wymagań funkcjonalnych, scenariusze użycia są powiązane z modelowaniem dynamiki systemu. Znane są techniki zaleceń projektowych i przewodniki stylu, np. dla programistów, a wszystkie zadania w projekcie podlegają planowaniu. Należy jednak pamiętać, że te same techniki mogą być stosowane w różnym celu. Wywiady, zalecenia, czy też testy mogą być



ukierunkowane na pozyskiwanie, spełnianie i weryfikację wymagań funkcjonalnych lub wymagań użytkowych.

- Według danych z ankiet, rzadko stosowane są w dużych firmach analiza konkurencji, badanie opinii, inspekcje użyteczności i analiza kontekstu użycia. Pozostałe techniki użyteczności nie znajdują zastosowania.
- Współpraca z grafikami występuje często. Graficy projektują wybrane elementy interfejsu użytkownika, dobierają kolorystykę produktów, czasami projektują interfejs użytkownika lub dbają o zgodność z wewnętrznym standardem wizualizacji. Nie zajmują się całościowo zapewnieniem jakości użytkowej.
- Celem stosowania technik użyteczności jest najczęściej poprawa jakości użytkowej. Wśród wymienianych celów pojawiają się również: ułatwienie projektowania interfejsów, zwiększenie intuicyjności interfejsu, zwiększenie atrakcyjności produktu, zwiększenie konkurencyjności oraz zmniejszenie liczby negatywnych ocen użytkowników i zgłoszeń problemów serwisowych. Natomiast wśród czynników wpływających na wybór technik użyteczności najczęściej wymieniane były łatwość wprowadzenia i analiza przewidywanych korzyści.
- W kwestii modelu integracji, najczęściej występuje włączenie poszczególnych technik użyteczności do procesu wytwarzania zdefiniowanego według zasad z inżynierii oprogramowania. Wśród odpowiedzi pojawiały się następujące sformułowania: „techniki użyteczności są stosowane jako elementy procesu wytwórczego”, „planowaniu podlegają wszystkie zadania, więc również zadania związane z zapewnieniem użyteczności”, czy też „analiza użyteczności jest częścią analizy wymagań, jedną z faz testowania są testy z udziałem użytkownika”. Techniki są włączane we wszystkich fazach, najczęściej jednak w fazach analizy i testowania. Ponadto zwrócono uwagę na potrzebę spójnego planowania komunikacji z klientem.
- Za najbardziej efektywną metodę uznawane było prototypowanie. Skuteczne okazały się także wywiady, analiza zadań i scenariusze użycia. Doceniane, chociaż mniej skuteczne są przewodniki stylów. Zdecydowana większość firm pracuje nad udoskonaleniem i wprowadzaniem nowych technik użyteczności do procesu wytwórczego.
- Podczas próby wprowadzenia technik użyteczności do procesu wytwórczego często pojawiały się następujące problemy:
  - problem czasu na wykonywanie zadań związanych z poprawą użyteczności;
  - brak porozumienia pomiędzy grupami osób.

- W około połowie badanych przypadków występował brak zrozumienia potrzeby przez kierownictwo. Natomiast rzadko występowały następujące problemy:
- brak jednolitej terminologii;
  - problem wprowadzenia zmian;
  - problem infrastruktury;
  - problem kosztów wdrożenia;
  - problem komunikacji z klientem lub użytkownikiem.
- Firmy radzą sobie z nimi zawsze poprzez szkolenie personelu, a czasami także sformułowanie formalnego procesu wytwarzania z uwzględnieniem użyteczności, sformułowanie zaleceń i wprowadzenie obiektów granicznych. Naszym zdaniem nasilenie problemów, które występują rzadko oraz dodatkowe problemy mogą się pojawiać w firmach, gdy wprowadzane będą techniki użyteczności mniej popularne w inżynierii oprogramowania i mniej podobne do innych technik z tej dyscypliny.
- Odpowiedniki obiektów granicznych występowały w około połowie badanych dużych firm, które stosowały zarówno techniki użyteczności, jak i techniki inżynierii oprogramowania. Zastosowanie takich nowoczesnych i zaawansowanych rozwiązań w polskich firmach informatycznych jest dobrym znakiem dojrzałości procesów wytwórczych. Potwierdzone zostało występowanie obiektów granicznych we wszystkich trzech grupach. Poniżej w każdej grupie wymieniani oni są według liczby wystąpień (najbardziej popularne w pierwszej kolejności). W grupie ludzi obiektami granicznymi są osoby przeszkolone w zakresie obu dziedzin, kierownik zespołu i programiści. W grupie produktów obiektami granicznymi są prototypy ekranów z funkcjami, specyfikacja wymagań, szkice ekranów, ustne omówienie, jak system ma działać i opisy użytkowników. Natomiast w grupie procesów wykorzystywane są obiekty graniczne takie jak planowanie działań w poszczególnych fazach, sprawdzanie wykonania w punktach kontrolnych i sprawdzanie skutków w czasie inspekcji.
- W przypadku dużych firm, które nie stosują technik użyteczności występują następujące problemy: defekty w interfejsie użytkownika i defekty w funkcjonalności, uwagi serwisowe, że użytkownicy nie potrafili się posłużyć funkcjonalnością, problemy z nauką systemu oraz rozwiązywaniem zadań krytycznych spowodowanych przez błędy w systemie. Ciekawym zjawiskiem, które wystąpiło w tych firmach i wymaga dokładniejszego zbadania, jest współwystępowanie problemów użyteczności i problemów inżynierii oprogramowania. Wśród przyczyn braku zastosowania metod użyteczności wymieniane były: brak (ziden-

tyfikowanej) potrzeby wprowadzenia technik, brak ustalonego procesu wytwarzania z takimi technikami, brak wsparcia ze strony kierownictwa.

- W przypadku małych i średnich firm sytuacja jest znacznie bardziej zróżnicowana. Wypełniane były wszystkie cztery warianty ankiety właściwej. Wśród przyczyn braku zastosowania metod użyteczności i/lub metod inżynierii oprogramowania wymieniane były: brak wiedzy na ten temat, brak ustalonego procesu twórczego, brak wyszkolonego personelu oraz zbyt duża ilość formalności - dokumentacji w przypadku braku stosowania technik inżynierii oprogramowania. Wskazano na istnienie wielu wad tego rozwiązania, ale stosowanie bardziej sformalizowanych metod wydawało się zbyt czasochłonne. Występujące problemy w tych firmach są dobrze znane w inżynierii oprogramowania, np. przekraczanie terminów, problemy spowodowane przez częstą zmienność wymagań, czy też problemy z wystarczającym przetestowaniem systemu.
- W małych i średnich firmach, które stosują zarówno techniki użyteczności jak i techniki inżynierii oprogramowania zauważalne jest duże podobieństwo do dużych firm, które stosują oba rodzaje technik pod względem celów, sposobów integracji, roli interfejsu użytkownika, niższego stopnia zaawansowania technik użyteczności oraz problemów podczas wprowadzania technik użyteczności. Jednakże występują następujące różnice:
  - rzadko występuje analiza zadań, wywiady, scenariusze użycia i planowanie użyteczności, natomiast w około połowie przypadków wykorzystywana była analiza kontekstu użycia, analiza konkurencji oraz techniki obserwacyjne;
  - mniejsza współpraca z grafikami – tylko w około połowie badanych przypadków;
  - zastosowanie w większości przypadków obiektów granicznych, jednak brak występowania takiego obiektu granicznego jakim są osoby przeszkolone w obu dziedzinach;
  - inne sposoby radzenia sobie z problemami występującymi podczas wprowadzania technik użyteczności – brak występowania szkoleń i brak określania formalnego procesu wytwarzania z zastosowaniem metod użyteczności, natomiast występowanie rozmów.
- W kwestii skuteczności zaznaczono małą skuteczność badania opinii za pomocą ankiet ze względu na małą liczbę odpowiedzi oraz potwierdzono skuteczność forum dyskusyjnego na temat produktów.
- Na pytanie o potrzebę publikacji i opracowań badawczych na pograniczu inżynierii oprogramowania i użyteczności, zdecydowana większość firm, w których były stosowa-

ne metody użyteczności, odpowiedziała pozytywnie. Wyjątkiem był reprezentant firmy, który postrzegał użyteczność jako część inżynierii oprogramowania i zgłosił potrzebę większej ilości prac dotyczących jakości użytkowej.

- Wśród sugestii tematów badawczych występowały następujące zagadnienia:
  - wpływ zastosowania technik z obu grup na efektywność procesu wytwarzania;
  - spojenie technik z obu grup w jeden proces bez znaczącego przyrostu czasu realizacji;
  - porównanie technik, ich skuteczności i korzyści z ich zastosowania;
  - koszty i metody wdrożenia w firmach;
  - wpływ rozmaitych czynników, np. typu systemu, klienta docelowego, ograniczeń sprzętowych, zaawansowania i rozpowszechnienia danej technologii, na zastosowanie poszczególnych metod użyteczności.
- Na pytanie o potrzebę szkolenia studentów w tym obszarze zdecydowana większość respondentów odpowiedziała pozytywnie. Wśród przydatnych kompetencji wymieniona została świadomość istotności zagadnień użyteczności i umiejętność oceny technologii. Szczególną uwagę zwrócono jednak na kompetencje o charakterze psychologicznym: świadomość punktu widzenia użytkownika i zdolność rozmowy z nim, umiejętność wydobywania wiedzy, słuchania i porozumiewania się, współpracy w grupie, umiejętność rozumienia potrzeb biznesowych.

#### ➤ Podsumowanie

- Przegląd publikacji z nurtu integracji dyscyplin inżynierii oprogramowania i użyteczności potwierdził występowanie problemu rozbieżności pomiędzy tymi dwoma dyscyplinami. Jednak w zależności od edukacji w danym kraju oraz kultury (preferencje dla wszechstronności wykształcenia lub wąskiej specjalizacji, naturalny poziom komunikacji) występowały nieco inne problemy i propozycje rozwiązań. Uniwersalnym osiągnięciem tego nurtu było wprowadzenie obiektów granicznych, zidentyfikowanie ich rodzajów oraz zaproponowanie szeregu sposobów na integrację procesu wytwarzania z uwzględnieniem technik z obu dyscyplin.
- Celem badania ankietowego było zbadanie sytuacji pod tym względem w Polsce. Większość zbadanych polskich firm stosuje zarówno techniki inżynierii oprogramowania jak i techniki użyteczności, przy czym stopień zaawansowania i formalności metod inżynierii oprogramowania jest większy niż metod użyteczności. Częściej stosowane są te techniki użyteczności, które są bardziej zbliżone do inżynierii oprogramowania i nauczane w na tym przedmiocie. Taki stan rzeczy jest rezultatem większej popularności inżynierii

oprogramowania w polskim środowisku, wynikającej z większej liczby książek dotyczących tej tematyki, większej liczby konferencji inżynierii oprogramowania i większej liczby przedmiotów specjalistycznych dotyczących inżynierii oprogramowania na kierunku informatyka. Wykształcenie polskich informatyków jest dość wszechstronne, a projektowanie interfejsów użytkownika przez długi czas było jednym z przedmiotów na kierunku informatyka. Dlatego też problem integracji technik użyteczności i inżynierii oprogramowania nie występuje w tak jaskrawy sposób. Czasami wręcz techniki projektowania interfejsów użytkownika traktowane są jako część inżynierii oprogramowania.

➤ W Polsce obecnie występuje mała potrzeba integracji, raczej istnieje tendencja wpro-

wadzania większej liczby technik użyteczności. Jednak podczas wprowadzania technik użyteczności bardziej odległych od terminologii i metod inżynierii oprogramowania mogą pojawiać się problemy opisywane w literaturze. Wówczas warto skorzystać z wiedzy na temat integracji i odpowiednio konfigurować proces wytwarzania z zastosowaniem technik z obu dyscyplin oraz łączących je obiektów granicznych.

- L.D. Bakalis, E. Folmer, J. Bosch; ; „Position Statement”, Proceedings of the Workshop „Identifying Gaps between HCI, Software Engineering, and Design, and Boundary Objects to Bridge them” at CHI Conference; 2004
- S. Berkun; „Position Statement”, Proceedings of the Workshop „Identifying Gaps Between HCI, Software Engineering, and Design, And Boundary Objects to Bridge them” at CHI Conference; 2004
- „Bridging the SE & HCI Communities”, <http://www.se-hci.org/bridging/> (10.5.2007)
- X. Ferre; „Integration of usability techniques into the Software Development Process”; Proceedings of the workshop „Bridging the Gaps Between Software Engineering and Human-Computer Interaction” at ICSE; 2003.
- B. Paech, K. Kohler; „Usability Engineering integrated with Requirements Engineering”; Proceedings of the workshop “Bridging the Gaps Between Software Engineering and Human-Computer Interaction” at ICSE; 2003.
- P.S. Pyla, M.A. Perez-Quinones, J.D. Arthur, H.R. Hartson; „Towards a Model-based Framework for Integrating Usability and Software Engineering Life Cycles”; Proceedings of the workshop „Closing the Gap: Software Engineering and Human-Computer Interaction” at INTERACT Conference; 2003.
- K.S. Sousa, E. Furtado; „UPi – A Unified Process for Designing Multiple UIs”; Proceedings of the workshop „Bridging the Gaps II: Bridging the Gaps Between Software Engineering and Human-Computer Interaction” at ICSE; 2004.

- Rational Unified Process v. 2001A.04.00.13