

Obiekty graniczne pomiędzy technikami inżynierii oprogramowania a technikami użyteczności i Kansei

ANNA BOBKOWSKA, RAFAŁ KAŹMIEROWSKI, AGNIESZKA WÓJCIK

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

Idea obiektów granicznych została spopularyzowana przez badania w nurcie integracji dyscyplin inżynierii oprogramowania i użyteczności (ang. Bridging the SE & HCI Communities) [5]. Obiekt graniczny jest to pewna abstrakcja wspólnej przestrzeni pojęć, podstawowych założeń i wspólnej wiedzy, wspólnych produktów, zadań i odpowiedzialności, które mogą służyć do integracji i koordynacji działań osób lub zespołów posiadających różne specjalizacje.

Pierwsza część referatu ma na celu podsumowanie zagadnień integracji inżynierii oprogramowania i użyteczności. W sekcji drugiej zawarto krótkie przypomnienie badań międzynarodowych oraz badań ankietowych przeprowadzonych w Polsce. W sekcji trzeciej podjęto próbę usystematyzowania korzyści z wymiany wiedzy pomiędzy dyscyplinami w następujących obszarach:

- wykorzystanie technik inżynierii oprogramowania w użyteczności,
- wykorzystanie technik użyteczności w inżynierii oprogramowania,
- integracja technik z obu grup w projekcie informatycznym.

Wymiana wiedzy pomiędzy dyscyplinami przynosi korzyści obu dyscyplinom. Zastosowanie technik inżynierii oprogramowania umożliwia poprawę precyzji opisu modeli i procesów stosowanych w użyteczności, a zastosowanie technik użyteczności daje możliwość poprawy jakości użytkowej narzędzi stosowanych w inżynierii oprogramowania (narzędzi CASE).

Druga część referatu ma na celu prezentację praktycznego rozwiązania integracyjnego w odpowiedzi na potrzebę 'spojenia technik z obu grup w jeden proces bez znaczącego przyrostu czasu realizacji'. Wybrano podejście bazujące na konfiguracji procesu i przedstawiono propozycję połączenia inżynierii Kansei oraz wybranej techniki użyteczności z IBM RUP (Rational Unified Process) [7]. W tym celu, w kolejnych sekcjach, techniki podlegające połączeniu zostały opisane zgodnie ze standardem SPEM (Software and Systems Process Engineering Metamodel) [11], który jest stosowany w opisie procesu RUP oraz omówione zostały obiekty graniczne łączące te techniki z RUP.

Zagadnienia integracji inżynierii oprogramowania i użyteczności

Nurt integracji dyscyplin inżynierii oprogramowania i użyteczności (ang. Bridging the SE & HCI Communities) [5] powstał w odpowiedzi na problemy związane z rozdzieleniem tych dwóch dyscyplin spowodowane ich niezależnym rozwojem, trudnościami w integracji wyników osiągniętych przez osobne grupy badaczy oraz problemami we współpracy na poziomie projektu informatycznego, którego realizacja powinna uwzględniać zarówno aspekty techniczne powiązane z inżynierią oprogramowania jak i aspekty związane z zapewnieniem jakości użytkowej. Jakość techniczna oprogramowania i jakość użytkowa są ze sobą ściśle powiązane, a odpowiednie skoordynowanie zadań z obu dyscyplin w procesie wytwarzania oprogramowania wpływa na efektywność prowadzenia projektu.

Sposobem na rozwiązanie tych problemów było wprowadzenie obiektów granicznych. Obiekt graniczny to pewna abstrakcja wspólnej przestrzeni pojęć, podstawowych założeń i wspólnej wiedzy, wspólnych produktów, zadań i odpowiedzialności, które mogą służyć do integracji i koordynacji działań osób lub zespołów posiadających różne specjalizacje. Obiekty graniczne mogą powstać na poziomie dyscyplin albo na poziomie projektów w konkretnych firmach, przy czym obiekty graniczne na poziomie dyscyplin ułatwiają zastosowanie w projektach informatycznych. W praktyce występują trzy rodzaje obiektów granicznych:

- ludzie, którzy posiadają umiejętności zrozumienia perspektywy innych, wykorzystania wykonywanych przez innych produktów, utworzenia płaszczyzny porozumienia oraz wspólnej przestrzeni działania, np. kierownicy projektów, osoby przeszkolone w zakresie obu dyscyplin;
- procesy, które definiują, jakie zadania, kiedy, jak i przez kogo powinny być wykonywane, np. zintegrowany proces wytwarzania oprogramowania, proces komunikacji i podejmowania decyzji;
- produkty, które zawierają opis sposobu wykonywania zadań, szablony i specyfikacje projektowe, np. specyfikacja wymagań, dokument planu projektu, szablon opisu kontekstu użycia, prototypy interfejsów użytkownika, raport z przeglądu.

Z badań ankietowych badających sytuację na styku inżynierii oprogramowania i użyteczności w Polsce [1] wynika, że w firmach informatycznych 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, np. prototypowanie, wywiady, przewodniki styku, scenariusze użycia, planowanie. Wśród czynników wpływających na wybór technik użyteczności najczęściej wymieniane były łatwość wprowadzenia i analiza przewidywanych korzyści. Jako model integracji najczęściej występuje włączenie poszczególnych technik użyteczności do procesu wytwarzania zdefiniowanego według zasad z inżynierii oprogramowania. W Polsce obecnie występuje tendencja wprowadzania coraz większej liczby technik użyteczności. Podczas wprowadzania technik użyteczności bardziej odległych od terminologii i metod inżynierii oprogramowania pojawiają się problemy opisywane w literaturze międzynarodowej i wówczas stosowane są obiekty graniczne pochodzące ze wszystkich trzech grup. Wśród

istotnych tematów badań na pograniczu inżynierii oprogramowania i użyteczności znalazł się problem spojenia technik z obu grup w jeden proces bez znaczącego przyrostu czasu realizacji.

Z przedstawionego podsumowania powiązanych z tematem rezultatów badania ankietowego wynika, że występuje potrzeba badań w zakresie obiektów granicznych. Dodatkowo, słusznym podejściem do integracji procesu wydaje się opis technik użyteczności zgodnie z zasadami opisu procesu w inżynierii oprogramowania, gdyż w ten sposób ułatwione jest włączenie ich do procesu wytwarzania.

Korzyści z wymiany wiedzy pomiędzy dyscypliną inżynierii oprogramowania a dyscypliną użyteczności

Większość badań z nurtu integracji użyteczności i inżynierii oprogramowania dotyczy zastosowania technik z obu grup w projekcie informatycznym. Jednak wymiana wiedzy pomiędzy tymi dyscyplinami może dostarczać korzyści także tym dyscyplinom. Osiągnięcia jednej dyscypliny są przydatne dla drugiej ze względu na koncentrację na innych aspektach lub ze względu na to, że w jednej dyscyplinie występowały bardziej skomplikowane problemy i stąd wypracowane zostały bardziej zaawansowane rozwiązania niż w drugiej. W dalszej części tej sekcji podsumowano badania w następujących trzech obszarach:

- wykorzystanie technik inżynierii oprogramowania w użyteczności,
- wykorzystanie technik użyteczności w inżynierii oprogramowania,
- integracja technik z obu grup w projekcie informatycznym.

Podstawą wykorzystania technik inżynierii oprogramowania w użyteczności jest obserwacja, że w inżynierii oprogramowania jest większa złożoność problemu, z którą musi sobie radzić wykonawca oprogramowania oraz z tego, że wymagany jest większy poziom precyzji. Można więc zastosować wypracowane techniki do poprawy precyzji opisu technik użyteczności. Jednym z zastosowań jest zastosowanie meta-modeli do opisu modeli zadaniowych [2], dzięki któremu uzyskano poprawę dokładności opisu i większą zwięzłość opisu. Dzięki temu łatwiejsze było zrozumienie i porównywanie modeli. Zaprezentowana w drugiej części tego referatu propozycja zastosowania metody opisu procesu wypracowanej w inżynierii oprogramowania (SPEM) do opisu procesu wykonywania zadań użyteczności i Kansei umożliwi dokładniejszy opis, ułatwia integrację z procesem inżynierii oprogramowania oraz daje szereg innych korzyści wynikających z konfiguracji procesu wytwarzania. Inżynieria oprogramowania posiada także bardziej zaawansowane techniki planowania projektu w porównaniu z planowaniem użyteczności, uwzględniając priorytety, cykle życia, harmonogramy itp. Podsumowując, dyscyplina użyteczności może wykorzystać z inżynierii oprogramowania wiedzę na temat opisu modeli, procesów oraz projektu.

Podstawą wykorzystania technik użyteczności w inżynierii oprogramowania jest obserwacja, że narzędzia CASE (ang. Computer Aided Software Engineering) są także oprogramowaniem, a jakość użytkowa tych narzędzi wpływa na efektywność i satysfakcję pracy informatyków. Niestety większość narzędzi CASE jest wytwarzana głównie z myślą o dostarczeniu określonej funkcjonalności. W celu oceny lub poprawy jakości użytkowej narzędzi CASE można zastosować techniki użyteczności. Badania oceny jakości użytkowej pokazały, że różnica w produktywności wykonywania tych samych zadań modelowania z zastosowaniem różnych narzędzi UML może dochodzić do 40% [3]. Za pomocą badań dostosowania narzędzi UML do różnych profili użytkowników można pokazać, że są znaczne różnice we wsparciu narzędzi UML dla poszczególnych ról związanych z wytwarzaniem oprogramowania [4]. Tradycyjne listy heurystyk użyteczności mogą być zastosowane (po drobnym dostosowaniu) do badania ogólnych cech jakości użytkowej narzędzi UML [13]. Z punktu widzenia firmy stosującej narzędzia UML, ocena ogólnej jakości użytkowej oraz dopasowania do konkretnych profili pracowników może mieć wpływ na wybór narzędzia. Natomiast firmy wytwarzające narzędzia UML mogą wykorzystywać te techniki do poprawy użyteczności swoich produktów. Przykładem narzędzia UML, którego producent zainteresował się aspektami użyteczności, jest Visual Paradigm [15]. Skutkiem tego jest większa możliwość dostosowania wyglądu narzędzia oraz diagramów do preferencji użytkownika; unikalny system uchwytów wokół elementów diagramu, który przyspiesza modelowanie powiązanych elementów oraz rozpoznawanie gestów wykonywanych myszką na podstawie których wstawiane są odpowiednie symbole. We wspomnianych badaniach produktywności Visual Paradigm zdobył najlepsze wyniki. Podsumowując, inżynieria oprogramowania może wykorzystać wiedzę z zakresu użyteczności do poprawy jakości użytkowej narzędzi CASE.

Integracja technik inżynierii oprogramowania i użyteczności w projekcie ma służyć zwiększeniu efektywności prowadzenia prac projektowych dzięki usunięciu nadmiarowości wykonywania podobnych prac przez zespoły z inną specjalizacją oraz poprawie komunikacji pomiędzy tymi zespołami. W ramach nurtu integracji inżynierii oprogramowania i użyteczności zaproponowano kilka sposobów integracji, np.:

- integrację użyteczności i inżynierii wymagań [8], w którym wyodrębniono decyzje oraz opis na poziomie zadań, dziedziny, interakcji oraz systemu;
- metodę włączania technik użyteczności do procesu wytwarzania oprogramowania [6] z dopasowaniem terminologii do słownictwa używanego w inżynierii oprogramowania;
- podejście do koordynacji działań z obu dyscyplin z przekazywaniem odpowiednich specyfikacji [9];
- próbę rozszerzenia RUP (Rational Unified Process) o opis ról, zadań i artefaktów związanych z użytecznością [12];
- włączenie do RUP opisu wytwarzania ukierunkowanego na użytkownika (ang. user centered design) i inżynierii użyteczności, jednak mają one status jedynie koncepcji i zaleceń.

W dalszej części tego referatu przyjęto podejście polegające na integracji z RUP. Jednak w odróżnieniu od podejścia opisanego w [12] polegającego na dopisaniu elementów użyteczności na niskim poziomie abstrakcji, w zaproponowanym podejściu wyróżniane są osobne dyscypliny Kansei i użyteczności oraz wskazywane są w jawny sposób obiekty graniczne, które służą ich połączeniu. Takie podejście jest bardziej elastyczne i zgodne z ideą konfigurowalności procesu. W odróżnieniu od ostatniego podejścia, zastosowane zostanie modelowanie zadań użyteczności zgodnie ze SPEM.

Opis procesu wytwarzania zgodnie ze standardem SPEM

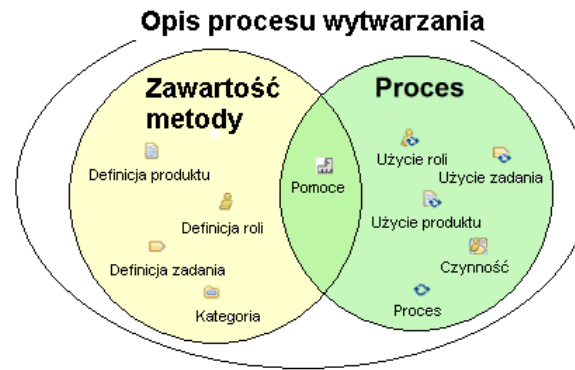
Software and Systems Process Engineering Meta-model (SPEM) [11] jest standardem służącym do opisu procesu wytwarzania oprogramowania. SPEM wyróżnia następujące poziomy opisu:

- zawartość metod (ang. method content), która zawiera definicję zadań, produktów oraz ról osób odpowiedzialnych za ich wykonanie w wymiarze statycznym;
- proces (ang. process), który wykorzystuje zdefiniowane zadania, produkty i role, łącząc je w większe fragmenty procesu realizowane podczas wytwarzania określonych typów systemów (wymiar dynamiczny).

Zawartość poszczególnych metod może być wykorzystana w różnych procesach lub w wielu miejscach tego samego procesu.

Opis procesu wytwarzania systematyzuje w firmie wiedzę na temat procesów wytwórczych, ułatwia konfigurację procesu i powoduje poprawę precyzji jego opisu. Dzięki dołączonym pomocom (ang. guidance), do których zaliczyć można szablony, listy kontrolne, opis dobrych praktyk, podręczniki, artykuły o powiązanej tematyce, procedury itp., stanowi pomoc podczas realizacji projektu oraz wprowadzenie dla nowych pracowników. Precyzyjny opis procesu ułatwia planowanie konkretnych projektów danego typu poprzez udostępnienie szablonu ich realizacji.

SPEM dostarcza notacji graficznej do modelowania zawartości metod oraz procesu. Najważniejsze symbole zostały pokazane na rysunku 1. Dla zawartości metod stosowane są głównie diagramy pokazujące powiązania ról z zadaniami i produktami, natomiast dla procesu – diagramy aktywności. Pierwszym procesem, który został opisany zgodnie ze SPEM, jest IBM RUP.

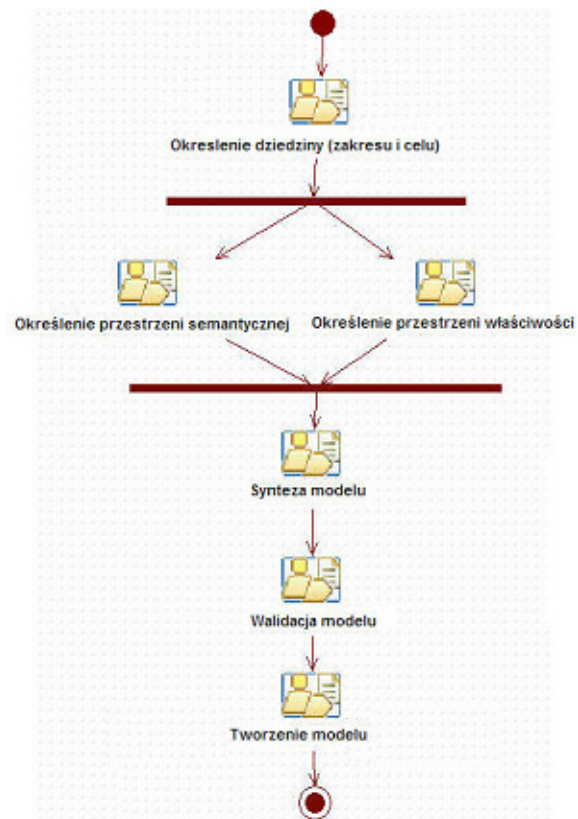


Rys. 1. Symbole stosowane podczas opisu zawartości metod i procesu, źródło: [11]

Zaprezentowana w tym referacie próba opisu technik Kansei oraz użyteczności zgodnie ze SPEM ma na celu zwiększenie precyzji opisu tych technik, łatwiejszą integrację tych technik z technikami inżynierii oprogramowania zawartymi w RUP (co wynika z zastosowania wspólnego języka opisu) oraz umożliwienie odniesienia szeregu korzyści, które są osiągane w inżynierii oprogramowania na skutek precyzyjnego opisu procesu wytwarzania.

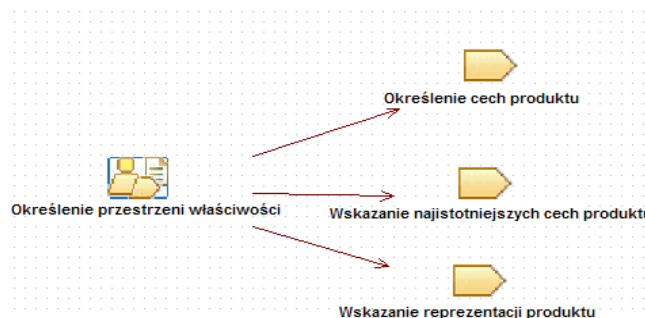
Opis inżynierii Kansei według SPEM

Na rysunku 2 pokazano czynności występujące w ramach inżynierii Kansei [10] opisywanej tu jako dyscyplina Kansei, a w tabelicy 1 przedstawiono opis tych czynności.



Rys. 2. Czynności występujące podczas realizacji procesu Kansei.

W skład czynności wchodzi zadania, a w opisie zadań występują kroki. Zadania wykonywane na etapie określania przestrzeni właściwości przedstawiono na rysunku 3.



Rys. 3. Zadania związane z określaniem przestrzeni właściwości.

Nazwa czynności	Opis czynności
Określenie dziedziny (zakresu i celu)	Definiowanie dziedziny produktu, czyli określenie zakresu produktu i jego celu, określenie środowiska docelowego produktu (produkty o podobnej cenie lub jakości, grupa wiekowa przyszłych użytkowników) oraz wyznaczenie najważniejszych cech produktu.
Określenie przestrzeni semantycznej	Określenie kolekcji słów opisujących właściwości produktu w celu wyodrębnienia słów najlepiej opisujących szukane cechy produktu. Przestrzeń określeń składa się z etapów: <ol style="list-style-type: none"> wyznaczenia zbioru słów opisujących typ wybranego produktu i jego dziedzinę; wskazaniu słów, które mają największy wpływ na umysł użytkownika; wskazanie 'słów Kansei', które mają największe znaczenie strategiczne dla tworzonego produktu.

Określenie przestrzeni właściwości	Określenie zbioru określeń technicznych lub inżynierskich produktu, które można wyrazić w wartościach liczbowych np. średnica, waga, czas (zob. rys. 3.) W ramach określania przestrzeni właściwości wyróżnione są następujące zadania: 1. Określenie cech produktu - wyszukiwane są słowa określające cechy produktu z różnych źródeł np. z technicznych specyfikacji produktu czy opinii ekspertów; 2. Selekcja najistotniejszych cech produktu; 3. Wskazanie reprezentacji produktu - tworzenie modeli produktów posiadających określone cechy.
Synteza modelu	Połączenie przestrzeni semantycznej z przestrzenią właściwości. Pierwszym krokiem jest ocena produktu przez użytkowników według skali semantycznej powiązanej ze 'słowa Kansei'. W ramach analizy wyników następuje powiązanie słów Kansei z przestrzenią semantyczną z określonymi właściwościami produktów z zastosowaniem metod statystycznych. Wynikiem syntezy obu zbiorów jest wskazanie kluczowych cech wytwarzanego produktu.
Walidacja modelu	Przewartościowanie 'słów Kansei' i wyznaczenie czynników różnicujących produkty z udziałem zamawiającego produkt, którego zadaniem jest wskazanie kluczowych pojęć określających cechy produktu zgodne z jego wizją produktu.
Tworzenie modelu	Tworzenie modeli matematycznych z uwzględnieniem wyników z poprzednich etapów.

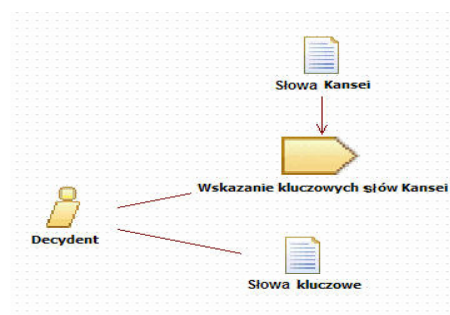
Tablica 1: Opis czynności występujących podczas realizacji procesu Kansei

Do każdego zadania i kroku można dołączyć pomoce zawierające wskazówki, jak ma przebiegać ich wykonanie, listy kontrolne oraz informacje o narzędziach, które można zastosować.

W inżynierii Kansei można wyodrębnić następujące role:

- kierownik procesu Kansei - zajmuje się planowaniem procesu Kansei oraz kontroluje ten proces i jego wyniki;
- analityk dziedziny - w fazie rozpoczęcia jest on odpowiedzialny za przeprowadzenie wywiadów z zamawiającym oraz rozpoznanie jego wizji i potrzeb względem produktu. Przeprowadza on również analizę rynku, w czasie której wyszukuje produkty o podobnych właściwościach estetycznych oraz ekonomicznych i określa grupę odbiorców na podstawie wywiadów z klientem;

- decydent – przedstawiciel zamawiającego, który akceptuje (lub nie akceptuje) wyniki integracji zbioru semantycznego ze zbiorem właściwości; może również wartościować słowa i tworzyć nowe na podstawie istniejących już zestawień;
- badacz - jest odpowiedzialny za określenie zbiorów semantycznych i właściwości, przewartościowanie wyników obu zbiorów oraz za stworzenie ankiet dla ankietowanych. Tworzenie zbiorów słów Kansei odbywa się zazwyczaj metodami heurystycznymi np. burza mózgów, w której mogą brać udział osoby nie związane z projektem, udziałowcy i osoby z określonej grupy odbiorców;
- ankietowany - osoba z określonej wcześniej przez analityka dziedziny grupy odbiorców, która bierze udział w ocenie produktu o określonych właściwościach względem słów Kansei pochodzących ze zbioru przestrzeni semantycznej.



Rys. 4. Rola Decydecnt wraz z jej zadaniami i produktami, za które jest odpowiedzialna

Przykładowy opis struktury odpowiedzialności dla roli decydecnt pokazano na rysunku 4. Wynika z niego, że decydecnt wskazuje kluczowe słowa Kansei na podstawie ‘słów Kansei’, a produktem wykonania tego zadania są ‘słowa kluczowe’.

Opis oceny heurystycznej z dyscypliny użyteczności

Ocena heurystyczna [14] jest jedną z technik zawartych w dyscyplinie użyteczności. Została ona zaklasyfikowana jako czynność w notacji SPEM. Ocena heurystyczna jest formą oceny użyteczności, w czasie której specjaliści użyteczności oceniają różne elementy interfejsu użytkownika na podstawie zdefiniowanego wykazu heurystyk użyteczności. Na rysunku 5 pokazano strukturę zadań i produktów dla oceny heurystycznej, a w tablicach 2 i 3 zawarto opis produktów i zadań występujących na diagramie. Rolę analityka użyteczności pełnią eksperci z dziedziny interakcji pomiędzy człowiekiem a komputerem oraz specjaliści od czynników ludzkich.



Rys. 5. Struktura zadań i produktów dla oceny heurystycznej

Produkt	Opis
System / Prototyp	Produkt, który zostanie poddany analizie użyteczności.
Heurystyki	Zbiór zasad i zaleceń użyteczności, np. heurystyki użyteczności J. Nielsena takie jak 'widoczność stanu systemu', 'estetyczna i minimalistyczna stylistyka'.
Listy problemów	Wynik indywidualnej pracy analityka w postaci listy wykrytych problemów pod względem heurystyk.
Raport z oceny heurystycznej	Raport zawierający ocenę heurystyczną przedmiotu analizy, który powstaje w wyniku scalenia list problemów.

Tablica 2: Opis produktów związanych z oceną heurystyczną

Zadanie	Opis
Wykrywanie problemów	Indywidualna praca analityka użyteczności polegająca na wyszukiwaniu defektów na podstawie heurystyk. Składa się z dwóch kroków: zapoznania się z systemem oraz szczegółowej kontroli, która może być powtarzana wiele razy (zob. rysunek 6) Produktem jest lista problemów.
Scalanie wyników	Scalanie wyników indywidualnej pracy kilku analityków użyteczności, analiza i tworzenie rankingu problemów. Produktem jest raport z oceny heurystycznej.

Tablica 3: Opis zadań związanych z oceną heurystyczną



Rys. 6. Wizualizacja kroków podczas wykrywania usterek

Szczegółowe zasady i zalecenia użyteczności mogą być zawarte w pomocach. Przykładowymi pomocami istotnym dla oceny heurystycznej jest zalecenie 'aby w ocenie heurystycznej brało udział od trzech do pięciu analityków', lista heurystyk i szablony list problemów lub raportów.

Obiekty graniczne

W wyniku modelowania oceny heurystycznej i inżynierii Kansei (w osobnych dyscyplinach użyteczności i Kansei) powstał opis tych technik w języku SPEM zgodnym z językiem opisu RUP, ale dotychczas te techniki nie zostały jeszcze powiązane z elementami RUP. Integracja z RUP polega na utworzeniu odpowiednich obiektów granicznych, tzn. na pokazaniu jakie produkty są przekazywane, jakie zadania są wykonywane wspólnie, kto jest odpowiedzialny za wykonanie i koordynację tych zadań itp. Wyodrębnienie obiektów granicznych powinno być poprzedzone analizą nakładania się elementów z integrowanych procesów oraz potencjalnych miejsc styku. Obiekt graniczny może występować na poziomie dyscypliny, jako wiedza na temat nakładania się elementów procesu, potencjalnych miejsc styku oraz propozycji ich połączenia, albo na poziomie procesu projektowego – wówczas reprezentuje konkretną konfigurację procesu.

Dyscypliny użyteczności i Kansei są najściślej związane z projektowaniem interfejsów użytkownika. Projektowanie interfejsu użytkownika w RUP występuje jako zadanie w dyscyplinie RUP o nazwie analiza i projektowanie. Rolą odpowiedzialną za wykonanie tego zadania jest projektant interfejsu, a produktami są mapa nawigacji oraz prototypy ekranów. Powinny one być wykonywane z zastosowaniem zaleceń projektowania interfejsu użytkownika na podstawie specyfikacji wymagań oraz innych dokumentów pomocniczych, np. wizji systemu, przypadków użycia, wymagań udziałowców. W aspekcie zakresu systemu i dopasowania systemu do zadań wykonywanych przez użytkownika, dyscypliny użyteczności i Kansei powiązane są ze specyfikacją wymagań oraz wizją systemu z dyscypliny RUP o nazwie wymagania.

Połączenie oceny heurystycznej z projektowaniem interfejsu użytkownika jest dość łatwe. Zadania te uzupełniają się, więc można je połączyć. Prototypy interfejsu użytkownika wykonane zgodnie z RUP mogą podlegać ocenie heurystycznej, a następnie wyniki tej oceny mogą być podstawą poprawy prototypów. Obiekt graniczny składa się z następujących elementów:

- przekazywane produkty: prototyp interfejsu użytkownika oraz raport z oceny heurystycznej;
- wspólne procesy: dyskusja wyników oceny, opis procesu przekazywania produktów;
- poziom osób: kierownik projektu, który jest koordynatorem działań projektantów interfejsu użytkownika i analityków użyteczności.

Połączenie dyscypliny Kansei z RUP jest już nieco trudniejsze. Inżynieria Kansei jest bardziej skomplikowana, a dodatkowo występuje nakładanie się produktów i zadań z obu dyscyplin. W zakresie produktów określenie dziedziny (zakresu i celu) z inżynierii Kansei zawiera elementy opisywane w wizji systemu (RUP), a prototyp interfejsu użytkownika (RUP) nakłada się na wynik wskazania reprezentacji produktu z inżynierii Kansei.

Nakładanie występuje również na poziomie zadań prowadzących do wykonania tych produktów. W tym przypadku jest możliwych kilka rozwiązań:

- wykonywanie prototypu interfejsu użytkownika (zgodnie z RUP), a następnie przekazanie go wraz ze specyfikacjami do udoskonalania poprzez techniki Kansei;
- zastąpienie projektowania interfejsu użytkownika (RUP) przez cały proces inżynierii Kansei - w tym przypadku występuje problem integracji z zakresem funkcjonalności (część inżynierii oprogramowania), więc obiektami granicznymi z kategorii produktów są specyfikacje wymagań, wizja systemu i przypadki użycia.

Podsumowanie

Zaproponowane podejście do integracji technik użyteczności i Kansei z procesem inżynierii oprogramowania (RUP) zawiera modelowanie tych technik zgodnie ze standardem SPEM oraz jawny opis obiektów granicznych. Techniki są modelowane w ramach osobnych dyscyplin: dyscypliny inżynierii Kansei oraz dyscypliny inżynierii użyteczności. Modelowanie zwiększa precyzję i spójność opisu oraz ułatwia połączenie z RUP dzięki zastosowaniu wspólnego języka opisu. Wyodrębnienie obiektów granicznych powinno być poprzedzone analizą miejsc nakładania i potencjalnych miejsc styku. Analiza miejsc nakładania służy wyeliminowaniu nadmiarowości pracy i niespójności pomiędzy różnymi produktami opisującymi te same wymagania lub decyzje. Analiza potencjalnych miejsc styku pokazuje punkty integracji technik z różnych dyscyplin w jeden proces. Miejscami styku mogą być przekazywane produkty; fragmenty procesów, w których łączą się lub są wspólnie wykonywane zadania z kilku dyscyplin oraz przydział odpowiedzialności koordynujących poszczególnym rolom. Gdy występuje nakładanie się zawartości technik, obiekty graniczne na poziomie dyscypliny mogą zawierać kilka propozycji integracji. Obiekty graniczne na poziomie projektu odzwierciedlają konkretną konfigurację procesu.

Literatura

1. A. Bobkowska, K. Grabiec; „Integracja technik użyteczności i technik inżynierii oprogramowania w projekcie informatycznym”, W: „Interfejs użytkownika - Kansei w praktyce”; 2007
2. A. Bobkowska, M. Kosidowska; „Porównanie podejść do opisu kontekstu użycia z wykorzystaniem meta-modeli”, W: „Interfejs użytkownika - Kansei w praktyce”; 2006
3. A. Bobkowska, K. Reszke; „Usability of UML modeling tools” W: Software engineering: evolution and emerging technologies, seria: „Frontiers in Artificial Intelligence and Applications”, IOS Press; 2005
4. A. Bobkowska, M. Weihs; „Verification of the Fit to User Profiles for UML Tools”, W: „Proceedings of 1st International IEEE Conference on Information Technology”; 2008.
5. „Bridging the SE & HCI Communities”, www.se-hci.org/bridging/ (dostęp 6.5.2008)

6. X. Ferre; „Integration of usability techniques into the Software Development Process”; W: Proceedings of the workshop „Bridging the Gaps Between Software Engineering and Human-Computer Interaction” at ICSE; 2003
7. IBM Rational Unified Process, www.ibm.com
8. B. Paech, K. Kohler; „Usability Engineering integrated with Requirements Engineering”; W: Proceedings of the workshop „Bridging the Gaps Between Software Engineering and Human-Computer Interaction” at ICSE; 2003
9. 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”; W: Proceedings of the workshop „Closing the Gap: Software Engineering and Human-Computer Interaction” at INTERACT Conference; 2003
10. S. Schütte, J. Eklund; „Product Development for Heart and Soul. An introduction to Kansei Engineering Methodology”, Linköpings Universitet, Department for Human Systems Engineering, Sweden; 2003
11. „Software and Systems Process Engineering Meta-model” (SPEM) <http://www.omg.org/docs/formal/08-04-01.pdf>
12. K.S. Sousa, E. Furtado; „UPi – A Unified Process for Designing Multiple UIs”; W: Proceedings of the workshop „Bridging the Gaps II: Bridging the Gaps Between Software Engineering and Human-Computer Interaction” at ICSE; 2004
13. M. Staszkiwicz; „Metoda oceny użyteczności narzędzi UML”, praca dyplomowa na wydziale ETI PG; 2007
14. Usability Net, www.usabilitynet.org (dostęp 9.05.2008)
15. Visual Paradigm for UML, www.visual-paradigm.com (dostęp 9.05.2008)