

Metoda punktów przypadków użycia

2009-10-01 / inżynieria oprogramowania, szacowanie oprogramowania

Metoda punktów przypadków użycia (ang. *Use Case Points*) w skrócie UCP posiada podobny mechanizm do tego jaki zastosowano w metodzie punktów funkcyjnych (**Metoda punktów funkcyjnych**), jednak zrezygnowano z wykorzystania ekranów i architektury. Dlaczego pominięto ekrany i architekturę? Otóż wyobraźmy sobie sytuację, w której klient zadaje pytanie o czas realizacji projektu na etapie specyfikowania wymagań. Jak się okazuje w tej sytuacji metoda punktów funkcyjnych jest nieużyteczna, gdyż nie posiadamy jeszcze zdefiniowanej architektury, czy też propozycji ekranów. Należy więc zastanowić się czy wymagania systemu mogą być podstawą do estymacji pracochłonności realizacji systemu informatycznego. Twierdząco na tak postawione pytanie odpowiedział w 1993r. Gustaw Karner – twórca metody punktów przypadków użycia. Wymienił on architekturę i ekrany na specyfikację wymagań zapisaną w postaci *specyfikacji przypadków użycia*. Ponadto zasugerował, że należy zwrócić uwagę na tak zwane *czynniki złożoności środowiska* opisujące w dużej mierze organizację, która wytwarza oprogramowanie, *czynniki złożoności technicznej* opisujące własności produktu oraz przyszłych *aktorów systemu*.

Klasyfikacja przypadków użycia jest w metodzie UCP jest podstawą estymacji pracochłonności w wytwarzaniu systemów informatycznych. Tu identyfikujemy wszystkie przypadki użycia, a dokładniej mówiąc specyfikacje przypadków użycia, ponieważ do szacowania potrzebny nam będzie opis scenariusza podstawowego i model klas analitycznych (identyfikacja klas zaangażowanych w realizację danego przypadku użycia). Na tym etapie należy określić poziom złożoności dla każdego przypadku użycia. Metoda wskazuje trzy typy złożoności.

Tabela 1. Klasyfikacja złożoności przypadków użycia

Złożoność przypadku użycia	Definicja	Waga
----------------------------	-----------	------

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

Średnio złożony	<ul style="list-style-type: none"> • Średnio złożony interfejs dla użytkownika. • Operuje na 2 (lub więcej) encjach bazy danych. • Podstawowy scenariusz zawiera 4 ? 7 kroków. • Implementacja obejmuje 5 ? 10 klas . 	10
Złożony	<ul style="list-style-type: none"> • Złożony interfejs dla użytkownika. • Operuje na 3 (lub więcej) encjach bazy danych. • Podstawowy scenariusz zawiera powyżej 7 kroków. • Implementacja obejmuje powyżej 10 klas. 	15

Następnie obliczamy **nieskorygowaną wagę przypadków użycia** w skrócie oznaczoną UUCW (ang . *Unadjusted Use Cases Weight*). W tym celu zliczamy liczbę przypadków użycia zakwalifikowanych do każdej ze wskazanych kategorii (prosty, średnio złożony, złożony). Następnie liczbę przypadków użycia w danej kategorii mnożymy przez wartość wagi dla danej kategorii złożoności przypadku użycia. Wartość nieskorygowanej wagi przypadków użycia stanowić będzie suma wcześniej zdefiniowanych iloczynów po każdej kategorii złożoności przypadku użycia. Podsumowując, formuła obliczania nieskorygowanej wagi przypadków użycia jest następująca:

$$UUCW = \sum_{i=1}^n n_i \cdot w_i$$

gdzie: n – liczba przypadków użycia zakwalifikowanych do i -tej kategorii złożoności przypadku użycia,
 w – wartość wagi dla i -tej kategorii złożoności przypadku użycia.

Czynniki złożoności środowiska (ang. *Environmental Complexity Factors*) charakteryzują dostawcę oprogramowania. Metoda wskazuje osiem czynników złożoności środowiska i każdemu z nich przypisuje wagę, która mówi jak bardzo dany czynnik wpływa na ostateczny wynik estymacji. Im waga czynnika jest większa tym czynnik ma większy wpływ na zmniejszenie prędkości wytworzenia systemu informatycznego. Warto zwrócić uwagę, że większość czynników posiada dodatnie wagi. Oznacza to, że wzrost ich wpływu redukuje prędkość. Z kolei wzrost wpływu czynników z ujemnymi wagami spowoduje zwiększenie wartości estymacji prędkości wytworzenia systemu informatycznego.

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

E1	Znajomość metodyki, języka UML	1,5
E2	Doświadczenie zespołu	0,5
E3	Znajomość technik obiektowych	1
E4	Umiejętności głównego analityka	0,5
E5	Motywacja zespołu	1
E6	Stabilność wymagań	2
E7	Udział pracowników w niepełnym wymiarze czasu	-1
E8	Skomplikowane języki programowania	-1

Przyjrzyjmy się bliżej czynnikom złożoności środowiska:

- E1 – mówi czy zespół jest zna dziedzinę problemu i techniczne aspekty rozwiązania problemu klienta. Zwraca się uwagę na znajomość metodyki w ramach której realizowany jest projekt np. RUP (ang. *Rational Unified Process*) jak również znajomość języków modelowania systemu np. UML (ang. *Unified Modeling Language*).
- E2 – ogólnie rozumiane doświadczenie zespołu w wytwarzaniu oprogramowania.
- E3 – doświadczenie w projektowaniu aplikacji zorientowanych obiektowo związane z umiejętnością projektowania aplikacji obiektowych, oraz umiejętność wykorzystania wspierających narzędzi do projektowania systemów informatycznych.
- E4 – określa zdolność analityka do właściwego pozyskiwania wymagań od klienta oraz posiadanie wiedzy z dziedziny rozwiązywanego problemu.

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

Ustawienia ciasteczek

Akceptuję

W metodzie punktów przypadków użycia wpływ każdego czynnika oceniamy w skali od 0 do 5 (zakres liczb naturalnych), przy czym:

- 0 ? oznacza, że czynnik jest nieistotny dla projektu
- 3 ? oznacza średni wpływ
- 5 ? oznacza silny wpływ

Następnie obliczamy wartość czynnika złożoności środowiska w skrócie oznaczonego ECF (ang. *Environmental Complexity Factors*) na podstawie następującej formuły:

$$ECF = \sum_{i=1}^n w_i \cdot impact_i$$

gdzie: w – wartość wagi i -tego czynnika, $impact$ – ocena wpływu i -tego czynnika.

Czynniki złożoności technicznej (ang. *Technical Complexity Factor*) charakteryzują przyszły system informatyczny. Metoda definiuje precyzyjnie chce określić specyfikę produktu wskazując trzynaście czynników. Podobnie jak czynniki środowiskowe, każdemu z nich przypisano wagę oddającą stopień w jakim czynnik wpływa na wartość końcowej estymacji. Wzrost wpływu każdego współczynnika złożoności technicznej powoduje zawsze wzrost końcowej wartości szacowanej pracochłonności.

Tabela 3. Czynniki złożoności technicznej

Symbol	Opis	Waga
T1	Rozproszenie systemu	2
T2	Wydajność systemu	1
T3	Wydajność dla użytkownika końcowego	1
T4	Złożone przetwarzanie wewnętrzne	1

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

T7	Łatwość użycia	0,5
T8	Przenośność	2
T9	Łatwość wprowadzania zmian	1
T10	Współbieżność	1
T11	Specjalne mechanizmy ochrony dostępu	1
T12	Udostępnianie użytkownikom zewnętrznym	1
T13	Dodatkowe szkolenia użytkowników	1

Przyjrzyjmy się również czynnikom złożoności technicznej:

- T1 – mówi czy w systemie wymagane jest rozproszone przetwarzanie danych.
- T2 – określa wydajność systemu , w odniesieniu do czasu reakcji na zdarzenia, przepustowość, itp .
- T3 – definiuje wydajność dla końcowego użytkownika w kontekście jego percepcji.
- T4 – określa czy wymagane są skomplikowane operacje przetwarzania danych, korzystanie z zaawansowanych algorytmów.
- T5 – mówi czy elementy lub kod wytwarzanego systemu, będzie wykorzystywany powtórnie.
- T6 – określa sposób instalacji łatwość i przyjazność instalacji systemu, wskazuje czy wymagany będzie udział specjalistów do instalacji i konfiguracji początkowej ze strony dostawcy oprogramowania.
- T7 – określa dostosowanie interfejsu użytkownika do jego potrzeb, wygodę w korzystaniu oraz łatwość nauczania się korzystania z systemu.

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

Ustawienia ciasteczek

Akceptuję

Analogicznie jak poprzednio wpływ każdego czynnika oceniamy w skali od 0 do 5 (zakres liczb naturalnych), przy czym:

- 0 ? oznacza, że czynnik jest nieistotny dla projektu
- 3 ? oznacza średni wpływ
- 5 ? oznacza silny wpływ

Następnie obliczamy wartość czynnika złożoności technicznej w skrócie oznaczonego TCF (ang. *Technical Complexity Factor*) na podstawie następującej formuły:

$$TCF = \frac{1}{n} \sum_{i=1}^n w_i \cdot impact_i$$

gdzie: w – wartość wagi i -tego czynnika, $impact$ – ocena wpływu i -tego czynnika.

Klasyfikacja aktorów jest kolejnym etapem związanym z estymacją projektu. Tu identyfikujemy wszystkich aktorów, którzy będą wchodzić w interakcję z przyszłym systemem informatycznym. Na tym etapie należy określić poziom złożoności dla każdego typu aktora. Metoda wskazuje trzy typy złożoności.

Tabela 5. Klasyfikacja złożoności aktorów

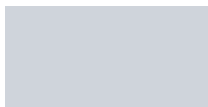
Złożoność aktora	Definicja	Waga
Prosty	System zewnętrzny, komunikujący się z docelowym przez interfejs programistyczny (API).	1
Średnio złożony	System zewnętrzny komunikujący się przez bardziej złożony protokół, np. http. Człowiek wchodzący w interakcję z systemem poprzez terminal tekstowy.	2

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

Ustawienia ciasteczek

Akceptuję

kategorii (prosty, średnio złożony, złożony). Następnie liczbę aktorów w danej kategorii mnożymy przez wartość wagi dla danej kategorii. Wartość nieskorygowanej wagi aktorów stanowić będzie suma wcześniej zdefiniowanych iloczynów po każdej kategorii złożoności aktora. Podsumowując, formuła obliczania nieskorygowanej wagi aktorów jest następująca:



gdzie: n – liczba aktorów zakwalifikowanych do i -tej kategorii, w – wartość wagi dla i -tej kategorii.

Działania metody punktów przypadków użycia dobiegają powoli końca. Wyznaczyliśmy już kilka współczynników, przyszedł w końcu czas na wykorzystanie ich wartości. Po obliczeniu nieskorygowanych wag aktorów UAW i nieskorygowanych przypadków użycia UUCW, możemy obliczyć nieskorygowane punkty przypadków użycia (ang. *Unadjusted Use Case Points*) w skrócie UUCP, poprzez zwykłe sumowanie dwóch powyższych wartości:



Natomiast interesujące nas szczególnie punkty przypadków użycia UCP wyznaczymy za pomocą formuły, która wykorzystuje wcześniej wyznaczone wartości czynnika złożoności technicznej TCF i czynnika złożoności środowiska ECF:



Ostatnim etapem w metodzie UCP jest przeliczenie punktów przypadków użycia na konkretne wartości pracochłonności liczone w osobogodzinach. Dokonuje się to poprzez pomnożenie UCP przez tak zwany współczynnik produktywności PF (ang. *Productivity Factor*). Współczynnik produktywności przekształca nam jeden punkt przypadku użycia na ilość godzin pracy człowieka. Historycznie wartość współczynnika PF wahała się od 15 do 30 roboczogodzin na jeden punkt przypadków użycia. Z badań Karnera, twórcy metody, wynika, że realizacja jednego UCP wymaga około 20 roboczogodzin.

Więcej na temat szacowania można znaleźć:

- [Metoda punktów funkcyjnych](#)
- [Metoda punktów przypadków użycia](#)

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

Podobne wpisy

Zintegrowane środowisko wytwarzania aplikacji web'owych na platformie .NET

W artykule przedstawiono opis pakietu narzędziowego VS.NET (Microsoft) z Rational XDE (IBM) do wytwarzania aplikacji webowych pracujących w środowisku urządzeń [więcej](#)

Rational Software Architect Pierwszy Krok

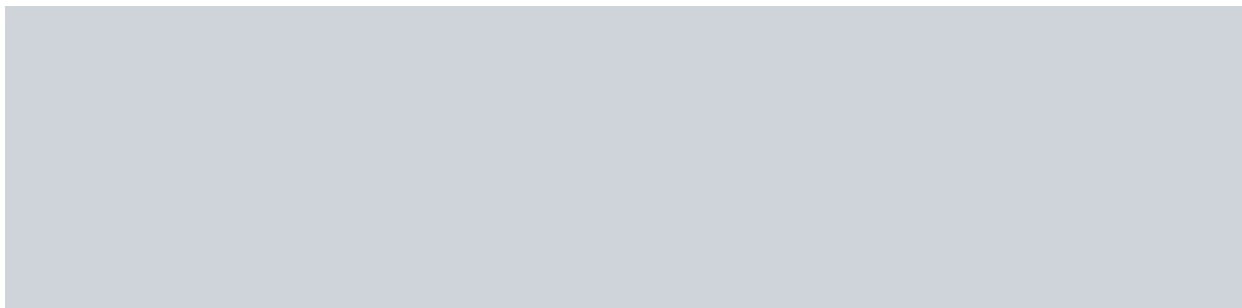
Technorati Tagi: Rational Software Architect, inżynieria oprogramowania W artykule zaprezentowano jak rozpocząć pracę z i opis elementów tego narzędzia CASE. Środowisko [więcej](#)

Wstęp do projektowania aplikacji w Rational Software Architect

Rational Software Architect jest kolejną po Rational Rose i Rational XDE generacją narzędzi wspierających twórców oprogramowania w czasie projektowania. RSA [więcej](#)

Rational Unified Process – Wstęp

Rational Unified Process jest zunifikowanym procesem wytwórczym oprogramowania dostarczającym praktycznych wskazówek, wzorców dokumentów i narzędzi, szablonów dokumentów oraz przykładów postępowania [więcej](#)



[← Poprzedni Wpis](#)

[Następny Wpis →](#)

3 komentarze dla "Metoda punktów przypadków użycia"

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

[Ustawienia ciasteczek](#)

[Akceptuję](#)

po czemu to i w tym pierwszym wzorze???? copy right wchodzi na wzór, rozumiem, że bezpieczeństwo przed kradzieżą tego drogiego wzoru jest ważniejsze niż czytelność?



MICHAŁ WOLSKI

2022-06-21 O 09:41

Obrazek ma ponad 10 lat, jakiś automat to kiedyś robił. Od wielu lat już nie stosuję tego typu mechanizmów. Mam nadzieję, że mimo tych trudności jest szansa, by skorzystać ze wzoru.

Dodawanie komentarzy zostało zablokowane.

Moje internetowe kursy



Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

[Ustawienia ciasteczek](#)

[Akceptuję](#)

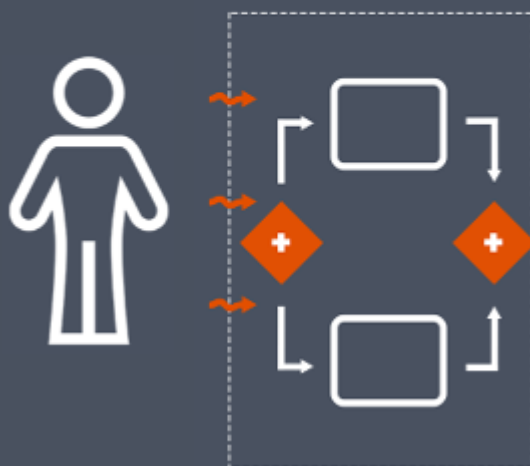
UML praktycznie

internetowy kurs z zadaniami



ENTERPRISE ARCHITECT

Kurs internetowy



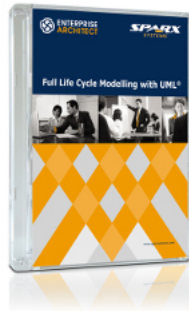
Polecam

Do modelowania używam Enterprise Architect:

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

[Ustawienia ciasteczek](#)

[Akceptuję](#)



Kupując **licencje Enterprise Architect** wspierasz rozwój tego bloga. Wpisując w treści kuponu: **michalwolski** otrzymasz rabat na licencje Enterprise Architect.

Zapisz się na newsletter od Michała Wolskiego

Email

- ☐ Potwierdzam zapisanie się na newsletter, i tym samym przetwarzanie mojego adresu e-mail, w celu otrzymywania powiadomień o planowanych webinarach, kursach, wpisach na blogu i innych inicjatywach Michała Wolskiego. (Wiem, że mogę wypisać w dowolnym momencie)

Zapisz mnie

[Strona główna](#)[Regulamin](#)[Polityka prywatności](#)[Regulamin newsletter](#)[Mapa witryny](#)

Strona wykorzystuje pliki cookies w celu jej prawidłowego działania, korzystania z narzędzi analitycznych i marketingowych. Dane zebrane za pomocą tych mechanizmów mogą być przesłane poza UE. Jeżeli wyrażasz zgodę na wykorzystywanie plików cookies, kliknij przycisk "Akceptuję". Przycisk "Ustawienia ciasteczek" pozwoli Ci na ustawienie swoich własnych preferencji. Więcej informacji znajdziesz w polityce prywatności.

[Ustawienia ciasteczek](#)[Akceptuję](#)