

**SIEMENS**

*Ingenuity for life*

## Siedem wniosków dotyczących zwinnego tworzenia oprogramowania

Przeniesienie zasad zwinności na poziom  
całego przedsiębiorstwa

Największe błędy popełniane w dziedzinie zwinnego tworzenia oprogramowania wiążą się z niemożnością ciągłego usprawniania procesów — głównie dlatego, że większość rzekomo zwinnych organizacji nie jest w stanie zmienić swojego sposobu myślenia. Nawet te, którym lepiej wychodzi wprowadzanie rewolucyjnych zmian w działaniach, zderzają się z brakiem rozwoju wykorzystywanych narzędzi, niechęcią do zmian lub niemożnością przeniesienia usprawnień osiągniętych w ramach procesu na łańcuch narzędzi. Dzięki zastosowaniu zasad zwinności na poziomie całego przedsiębiorstwa korporacje i inne duże organizacje dostrzegają nareszcie możliwość pełnego wykorzystania potencjału podejścia zwinnego.

# Spis treści

<b>Bolesne wnioski dotyczące zwinnego tworzenia oprogramowania.....</b>	<b>3</b>
Lekcja 1: Wielkie (wygórowane) nadzieje.....	3
Lekcja 2: Zwinność przeważnie tkwi w głowie.....	4
Lekcja 3: Zwinność z różnych punktów widzenia.....	4
Lekcja 4: Zestawienie (błędnie pojmowanych) praw .....	5
Lekcja 5: Proces, narzędzia i zmiany.....	5
Lekcja 6: Najlepsze plany.....	6
Lekcja 7: Zarzuty .....	6
<b>Przeniesienie zasad zwinności na poziom całego przedsiębiorstwa .....</b>	<b>8</b>
<b>Pięć zaleceń dotyczących przeniesienia zasad zwinności na poziom całego przedsiębiorstwa.....</b>	<b>9</b>

# Bolesne wnioski dotyczące zwinnego tworzenia oprogramowania

Wiele organizacji tworzących oprogramowanie rozważa rozszerzenie zasad zwinności na całe przedsiębiorstwo w nadziei, że problemy związane z podejściem zwinnym, które występują na poziomie zespołu uda się rozwiązać dzięki przeniesieniu tych zasad na poziom przedsiębiorstwa. Prawdopodobnie się to nie uda – problemy tylko się nasilą.

Poniżej przedstawiono niektóre wnioski wyciągnięte z typowych błędów popełnianych podczas wprowadzania zasad zwinnego tworzenia oprogramowania i ich wykorzystywania w zespołach programistów.

## Lekcja 1: Wielkie (wygórowane) nadzieje

„Od dziś wdrażamy zasady zwinnego tworzenia oprogramowania”. To proste stwierdzenie ma sprawić, że ludzie przejdą całkowitą przemianę — począwszy od zespołu programistów, przez kierowników, a skończywszy na klientach. Zespół natychmiast zapomni o obowiązujących terminach i pozbedzie się wszelkich specyfikacji. Kierownictwo dokona cięć w inwestycjach, mając

poczucie, że projekt zostanie ukończony na czas i zmieści się w zadanym budżecie. Klienci co chwilę zaczną zmieniać swoje życzenia i wymagania.

Metodologia zwinnego tworzenia oprogramowania zaszczerpa w zespole programistów między innymi takie oczekiwania:

- „Dzięki zwinnemu podejściu możemy zapomnieć o specyfikacjach i dokumentacji”.
- „Dzięki zwinnemu tworzeniu oprogramowania mamy mniej pracy”.
- „Zwinne tworzenie oprogramowania zdejmuje z nas odpowiedzialność”.

A takie wygórowane oczekiwania pojawiają się za sprawą zwinnego podejścia w zespole kierowniczym lub komitecie wykonawczym:

- „Metody zwinne pozwalają na dokonanie cięć w budżetach przeznaczonych na programowanie, zapewnienie jakości i narzędzia”.
- „Dzięki zwinnemu podejściu będziemy w stanie zapewnić krótki czas realizacji i niską cenę”.
- „Zwinne tworzenie oprogramowania uczyni głupców geniuszami”.



### Lekcja 2: Zwinność przeważnie tkwi w głowie

„Programując od wielu lat, widzieliśmy już setki metod. Zwinne tworzenie oprogramowania to po prostu kolejna z nich”.

Zwinne podejście do tworzenia oprogramowania to sposób myślenia zwyczajowo nazywany zwinnością. Niektórzy twierdzą, że jeśli nie ma się w sobie zwinności, nie można się jej nauczyć.

Jeśli przekonania, że wcale nie trzeba aż tak wiele zwinności, by osiągnąć zwinność nabrać.

Lata doświadczeń wskazują, że ogólny konsensus zmienia się z wersją następującego kierunku: nabranie zwinności jest możliwe, ale

wymaga to czasu, zaangażowania, poświęcenia i odpowiednich szkoleń. Oto niektóre z błędnych wyobrażeń lub wprowadzających w błąd przekonań dotyczących podejścia zwinnego.

- Zwinne tworzenie oprogramowania to pestka.
- Programiści muszą działać zwinnie, ale pozostałe osoby już niekoniecznie.
- Jeśli pracujesz w firmie stosującej metody zwinne, myślisz tymi samymi kategoriami.
- Wprowadzając do procesu dowolne narzędzie z dziedziny zwinnego tworzenia oprogramowania, zwiększasz swoją zwinność.
- Szkolenie z metod zwinnych sprawi, że wszyscy nabiorą zwinności.
- Jestem zagorzałym zwolennikiem metodologii zwinnej. Coś, co nie jest zgodne z moją definicją zwinności, nie ma ze zwinnością absolutnie nic wspólnego (i nadaje się tylko do kosza).



### Lekcja 3: Zwinność z różnych punktów widzenia

Trzeci wniosek dotyczy definicji zwinności i jej możliwych błędnych interpretacji ze strony różnych zainteresowanych osób, które często postrzegają tę metodologię przez pryzmat własnej roli:

(Zniekształcony) punkt widzenia kierownika:

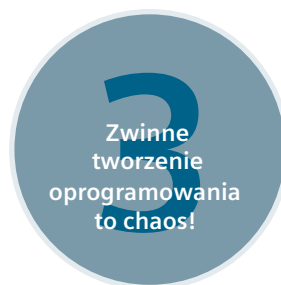
- „Zwinne tworzenie oprogramowania to chaos”.
- „Zwinne tworzenie oprogramowania oznacza brak kontroli”.
- „Zwinne tworzenie oprogramowania oznacza brak zaangażowania”.
- „Zwinne tworzenie oprogramowania oznacza brak ingerencji”.
- „Zwinne tworzenie oprogramowania to rozwiązanie niskobudżetowe”.
- „Każda zmiana wymaga stworzenia planu”.
- „Każda zmiana wymaga sporządzenia umowy”.

(Zniekształcony) punkt widzenia programisty:

- „Zwinne tworzenie oprogramowania jest dla programistów; nie jesteś programistą, więc go nie zrozumiesz”.
- „Zwinne tworzenie oprogramowania oznacza brak zobowiązań”.
- „Zwinne tworzenie oprogramowania oznacza brak kluczowych etapów”.
- „Jestem programistą zajmującym się zwinnym tworzeniem oprogramowania, dlatego wiem, co mam robić”.
- „Jesteśmy zespołem zwinnie tworzącym oprogramowanie, więc... zróbmy sobie urlop”.

(Zniekształcony) punkt widzenia eksperta od zasad zwinności:

- „Jestem ekspertem od zasad zwinności, dlatego to ja zdecyduję, co macie robić”.
- „Nie tak powinno to wyglądać według zasad zwinności”.



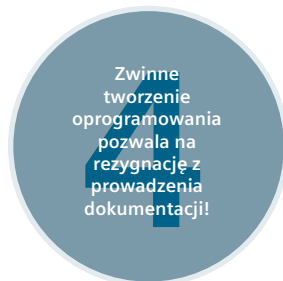
#### Lekcja 4: Zestawienie (błędnie pojmowanych) praw

Gdy zespół lub dział zaczyna działać według zasad zwinności, ludzie uważają, że muszą przygotować się na pewne wyrzeczenia. Programiści zajmujący się zwinnym tworzeniem oprogramowania błędnie zakładają, że mają prawo:

- Używać zwrotu „Cóż, działamy według zasad zwinnego tworzenia oprogramowania” jako odpowiedzi na każde pytanie (zazwyczaj mając na myśli „nie”)
- Utożsamiać zasady zwinnego tworzenia oprogramowania z przyzwoleniem na zrezygnowanie z prowadzenia dokumentacji
- Utożsamiać zasady zwinnego tworzenia oprogramowania z przyzwoleniem na ignorowanie umów
- Używać dowolnego narzędzia, by spełnić wymogi zwinnego tworzenia oprogramowania (bądź całkowicie zrezygnować z wykorzystywania narzędzi)

Po stronie osób zainteresowanych projektem również powstaje bardzo niepokojący zestaw „praw”:

- „Prawo” do opisywania swoich potrzeb w (bardziej niż dotychczas) niejasny sposób
- „Prawo” do ciągłej zmiany zdania
- „Prawo”, by powiedzieć: „Skończcie dyskutować i dostarczcie mi rozwiązanie”.



#### Lekcja 5: Proces, narzędzia i zmiany

Jeden z najsłabszych wniosków, jakie nasuwają się po wprowadzeniu zasad zwinności dotyczy usprawniania procesów. Jest to rzecz niełatwa, ponieważ oznacza zmianę, a ludzie nie lubią zmian. Ciągłe usprawnianie procesów oznacza ciągłe zmiany.

Oto typowe argumenty przeciwko usprawnianiu procesów:

- „Nie sądzę, by ta praktyka była dla nas dobra, dlatego ją pominiemy”.
- „Nasz proces jest stabilny”.
- „W tej chwili nasz proces (mniej więcej) działa. Po co więc mówić o zmianie?”

Z drugiej strony, jednym z największych wrogów usprawniania procesów są narzędzia będące już w użyciu:

- „Nie potrzebujemy narzędzia”.
- „Najlepszym narzędziem do zwinnego tworzenia oprogramowania jest Excel”.
- „Działamy w sposób zwinny, bo mamy narzędzie zapewniające nam tę zwinność”.
- „Nie możemy zmienić procesu, ponieważ nasze narzędzie nie wspiera zmian, które chcemy wprowadzić”.

To ostatnie stwierdzenie niejednokrotnie staje się głównym tematem wielu dyskusji.

Dany proces jest dobry, gdy:

- Staje się kwestią przyzwyczajenia: ludzie nie muszą myśleć o procesie ani stale wracać do podręcznika, który wyjaśni im, jak zrobić krok naprzód.

lub

- Jest zautomatyzowany: wiedza na temat tego, jak zrobić krok naprzód jest zawarta w narzędziu, które nie pozwala na błędną interpretację, w przeciwieństwie do polegania na ludziach, którzy muszą nauczyć się procesu i zapamiętać wszystkie szczegóły.

Przy zmianie procesów co dwa tygodnie pierwsza opcja nie wchodzi w grę: proces nigdy nie stanie się kwestią przyzwyczajenia. Nie wystarczy czasu, by się do niego przyzwyczaić. Dlatego też ciągłe lub częste usprawnianie procesów jest niemożliwe, o ile nie użyje się narzędzia, które prowadzi ludzi przez cały proces i jest przy tym na tyle otwarte, by z łatwością dostosować się do wszelkich zmian w procesie. Wielu dostawców twierdzi, że są w stanie tego dokonać. Jedynym narzędziem, które wykazuje tę zdolność w praktyce, a na dodatek zostało stworzone i opracowane od podstaw w tym celu, jest rozwiązanie Polarion® firmy Siemens PLM Software do zarządzania cyklem życia aplikacji (ALM).

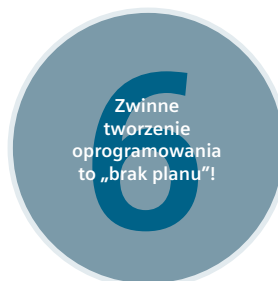


## Lekcja 6: Najlepsze plany

Jedną z najbardziej rewolucyjnych zmian, jakie wprowadziła metodyka zwinnego tworzenia oprogramowania, jest zmiana dotycząca planowania. Właściwie stosowane planowanie iteracyjne daje dużą elastyczność, umożliwiając pozytywne interakcje między programistami i osobami zainteresowanymi projektem, a także ukazuje ciągły i namacalny postęp — rzecz jasna, haczykiem jest tutaj określenie „właściwie stosowane”.

Oto niektóre spośród błędnych interpretacji związanych ze zwinnym planowaniem, które zazwyczaj zwiększają ryzyko związane z jego wdrożeniem i/lub powodzeniem.

- Zwinne tworzenie oprogramowania to „brak planu”.
- Zwinne tworzenie oprogramowania to brak kluczowych etapów.
- Zwinne tworzenie oprogramowania oznacza elastyczne podejście do etapów kluczowych.
- Iteracja to ramy czasowe, w których trzeba coś zrobić.
- Iteracja to przedział czasowy między dwoma spotkaniami przeglądownymi.
- Iteracja rozpoczyna się dwa dni przed jej zakończeniem.
- Iteracji nie można anulować.
- Zwinne podejście pomaga w planowaniu.



## Lekcja 7: Zarzuty

Jakakolwiek kombinacja nadużyć i nieporozumień przedstawionych we wnioskach od 1 do 6 tak naprawdę sprowadza się do formułowania zarzutów pod adresem metod zwinnych. Nikt nigdy nie powie: „Nie radzimy sobie z wykorzystywaniem metod zwinnego tworzenia oprogramowania”. Powie raczej, że „zwinne podejście jest do niczego”.

Oto niektóre z najbardziej typowych stwierdzeń:

- „Zasad zwinności nie da się zastosować w dużych grupach/projektach/przedsiębiorstwach”.
- „Nie da się stosować zasad zwinności, jeśli trzeba dotrzymać terminu”.
- „Ze względu na biurokrację zwinne podejście nie ma u nas racji bytu”.
- „Nie da się stosować zasad zwinności, jeśli chce się mieć kontrolę nad działaniami programistów i je monitorować”.
- „W przypadku naszych klientów/programistów/kierowników działanie według zasad zwinności nie ma racji bytu”.



# Przeniesienie zasad zwinności na poziom całego przedsiębiorstwa

Część z przedstawionych w artykule błędnych interpretacji i zniekształconych praktyk, które negatywnie wpływały na próby wykorzystania zasad zwinności w ciągu ostatnich 15 lat, to wyolbrzymienia. Podejście zwinne już od dłuższego czasu znajduje się w centrum zainteresowania, a zespoły programistów uczą się i doskonalą swoje umiejętności z roku na rok, projekt po projekcie. Może nas korcić, aby zignorować wspomniane wnioski jako niedotyczące nas samych lub jako lekcje z przeszłości.

Ryzyko polega na tym, że w opartym na zasadach zwinności środowisku programistycznym zapewniającym rozwój produktu Polarion firmy Siemens PLM Software możemy nie dostrzec już niektórych koncepcji i zachowań omówionych w powyższych bolesnych wnioskach (bądź zdążyliśmy się już do nich przyzwyczaić lub po prostu je zaakceptowaliśmy). Może tak być dlatego, że teraz nie mają już one aż takiego wpływu na nasz sukces we wdrażaniu metod zwinnego tworzenia oprogramowania.



# Pięć zaleceń dotyczących przeniesienia zasad zwinności na poziom całego przedsiębiorstwa

1. Zastanów się nad decyzją o przeniesieniu zasad zwinności na poziom całego przedsiębiorstwa. Zdecyduj się na to tylko wtedy, gdy naprawdę satysfakcjonuje Cię dotychczasowe wdrożenie zasad zwinnego tworzenia oprogramowania i potrzebujesz czegoś, co przeniknie na kolejne poziomy Twojej organizacji. Nie rób tego, jeśli Twoją jedyną motywacją jest niezadowolenie z dotychczasowego wdrożenia zasad zwinności lub jeśli uważasz, że przeniesienie tych zasad na poziom całego przedsiębiorstwa rozwiąże napotkane wcześniej problemy na poziomie zespołu/projektu.
2. Jeśli uważasz, że Twoja organizacja nie nadaje się do stosowania zasad zwinności na poziomie zespołu/projektu, to tym bardziej nie będzie nadawać się do wprowadzenia tych zasad na poziomie przedsiębiorstwa. Przedstawione poniżej poglądy sprawią, że wszelkie próby rozszerzenia zasad zwinności na całe przedsiębiorstwo spełzną na niczym:
  - „Zastosowanie określonych praktyk zwinności sprawi, że z modelu kaskadowego przejdziemy na zwinność na poziomie całego przedsiębiorstwa”.
  - „Nie możemy stosować zasad zwinności w skali zespołu, ponieważ jesteśmy zbyt dużym podmiotem. Spróbujmy więc zastosować je na poziomie całego przedsiębiorstwa”.
3. To nie tak, że zwinność na poziomie całego przedsiębiorstwa zapewni poszczególnym zespołom zwinność lub jest czymś, co pozwala organizacjom, które nie wprowadziły zasad zwinności, lepiej kontrolować działania bądź uzyskać od swoich zespołów ładnie wyglądającą dokumentację i solidne kluczowe etapy realizacji projektów. Zwinność to sposób myślenia, a przeniesienie jej zasad na poziom całego przedsiębiorstwa zmienia sposób myślenia o firmie.
4. Stosuj podejście przyrostowe. Istnieje kilka różnych rodzajów podejścia do wdrażania zasad zwinności na poziomie całego przedsiębiorstwa — może to być zarówno podejście typu „wszystko za jednym zamachem”, jak i zaplanowanie usprawnień rozłożonych w czasie. Po bliższym przyjrzeniu się sprawie szybko okazuje się jednak, że podejście typu „wszystko za jednym zamachem” nie działa. Podobnie rzecz ma się z tworzeniem oprogramowania Polarion: w 2008 roku zespół zaczął działać według zasad zwinnego tworzenia oprogramowania i w dalszym ciągu wdraża nowe praktyki, sterując tym procesem od ośmiu lat — i stopniowo przenosząc zasady zwinności na poziom całego przedsiębiorstwa. W artykule InfoQ zatytułowanym „Siedem grzechów głównych podejścia Scrum i innych antywzorców zwinnego tworzenia oprogramowania” (<http://www.infoq.com/news/2016/03/agileindia-7sins-scrum>) wyraźnie stwierdzono, że najlepszą radą dla zwolenników zwinnego tworzenia oprogramowania jest koncepcja przyrostowości:
  - Wykorzystuj retrospektywy.
  - Wprowadzaj usprawnienia stopniowo.
  - W danym momencie skup się na jednej lub dwóch rzeczach.
  - W miarę potrzeb korzystaj ze szkoleń.
5. Na koniec jeszcze jedna, nie mniej ważna rada: skorzystaj z odpowiedniego narzędzia. Jeśli używanie niewłaściwego narzędzia stanowi problem w przypadku zwinnego tworzenia oprogramowania, stanie się wręcz koszmarem po przeniesieniu tych zasad na poziom całego przedsiębiorstwa. Procesy zwinności na poziomie przedsiębiorstwa są o wiele bardziej złożone niż te, z którymi mamy do czynienia na poziomie zespołu. To oczywiste: angażują o wiele więcej osób pracujących w danej organizacji i obejmują swoim zasięgiem znacznie więcej działań niż „tylko” tworzenie oprogramowania.
 

Chcąc skorzystać z zasad zwinności przeniesionych na poziom całego przedsiębiorstwa, musimy przyjąć podejście stopniowego wprowadzania zmian. Tajemnicą sukcesu jest częste lub ciągłe usprawnianie procesów.

Przyjmując zasady zwinności na poziomie całego przedsiębiorstwa i korzystając z nich, trzeba dysponować zestawem narzędzi, który:

  - obejmuje wiele dziedzin,
  - wspiera poszczególne praktyki i procesy w sposób przyrostowy,
  - zawiera wiedzę na temat procesu i przeprowadza użytkowników przez cały proces,
  - pomaga zmierzyć jakość procesu,
  - jest otwarty na łatwe usprawnienia procesów.

## Siemens PLM Software

### Centrala

Granite Park One  
5800 Granite Parkway  
Suite 600  
Plano, TX 75024  
USA  
+1 972 987 3000

### Ameryka Pn. i Płd.

Granite Park One  
5800 Granite Parkway  
Suite 600  
Plano, TX 75024  
USA  
+1 314 264 8499

### Europa

Stephenson House  
Sir William Siemens Square  
Frimley, Camberley  
Surrey, GU16 8QD  
+44 (0) 1276 413200

### Azja i Oceania

Suites 4301-4302, 43/F

AIA Kowloon Tower, Landmark  
East

100 How Ming Street

Kwun Tong, Kowloon

Hongkong

+852 2230 3308

## Informacje o firmie Siemens PLM Software

Siemens PLM Software — jednostka biznesowa oddziału Siemens Digital Factory — to czołowy globalny dostawca systemów, oprogramowania i usług z zakresu zarządzania cyklem życia produktu (PLM) i zarządzania działalnością produkcyjną (MOM). Z produktów tych korzysta ponad 140 000 klientów na całym świecie (wykupili oni ponad 15 milionów licencji stanowiących). Siemens PLM Software ma siedzibę w Plano w Teksasie i we współpracy z partnerami dostarcza na rynek oprogramowanie branżowe, które ułatwia przedsiębiorstwom na całym świecie zdobycie zrównoważonej przewagi nad konkurencją w drodze realizacji wartościowych innowacji. Więcej informacji na temat produktów i usług firmy Siemens PLM Software można znaleźć pod adresem [www.siemens.com/plm](http://www.siemens.com/plm).

## [www.siemens.com/plm](http://www.siemens.com/plm)

© 2017 Siemens Product Lifecycle Management Software Inc. Siemens, logo Siemens i SIMATIC IT są zastrzeżonymi znakami towarowymi spółki Siemens AG. Camstar, D-Cubed, Femap, Fibersim, Geolus, I-deas, JT, NX, Omneo, Parasolid, Solid Edge, Syncrofit, Teamcenter i Tecnomatix są znakami towarowymi lub zastrzeżonymi znakami towarowymi spółki Siemens Product Lifecycle Management Software Inc. lub jej podmiotów zależnych w Stanach Zjednoczonych i innych krajach. Excel to zastrzeżony znak towarowy firmy Microsoft Corporation. Wszystkie pozostałe znaki towarowe, zastrzeżone znaki towarowe i znaki usługowe należą do odpowiednich podmiotów.

58112-78500-A8-PL 8/19 LOC