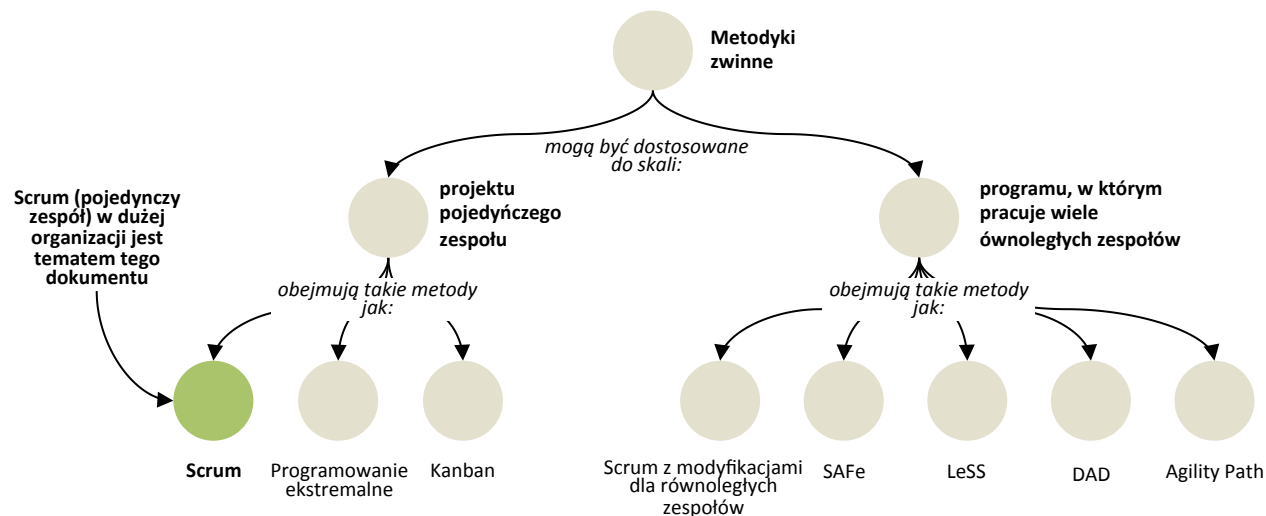


Scrum w dużych organizacjach mapa pojęć

zakres dokumentu

Piszemy o stosowaniu Scrum w środowisku dużej firmy.

- Scrum jest jedną z metodyk zwinnych, które służą do organizacji projektów IT. Tematem tego dokumentu jest Scrum, ponieważ spośród wszystkich metodyk zwinnych zyskał największą popularność. Mimo to, jeśli korzystasz z innej metodyki zwinnej, duża część tego dokumentu nadal będzie mieć zastosowanie
- Scrum nie jest metodyką zarządzania programem. Jeśli zamierzasz zarządzać programem składającym się z wielu projektów zwinnych połączonych współzależnościami, powinieneś dostosować metodykę tak, aby była skalowalna (publicznie dostępne są opisy takich przedsięwzięć, np. Spotify) lub użyć jednej z gotowych metod, np. SAFe – jest to jednak poza zakresem tego dokumentu
- Część praktyk, które opisujemy, wynika z warunków narzucanych przez specyfikę dużych organizacji i nie są częścią Scrum (np. potrzeba szacowania i śledzenia budżetu)



Czy metodyki zwinne są skalowalne?

- Zarządzanie równoległymi zespołami wymaga wprowadzenia zmian do oryginalnej wersji Scrum
- Największe trudności to między innymi istnienie wielu Właścicieli Produktu, współzależności pomiędzy zespołami oraz konieczność monitorowania postępu programu jako całość
- Najprostszym przypadkiem skalowania ponad jeden zespół byłby jeden Właściciel Produktu zarządzający rejestrem rozwijanym przez kilka zespołów, podczas gdy Scrum Master odpowiadałby za zarządzanie zależnościami
- Jednym z podejść do dalszego skalowania jest stopniowe modyfikowanie podejścia, które sprawdziło się w mniejszej skali; innym podejściem natomiast będzie użycie gotowej metodyki dopasowanej do większej skali (patrz mapa pojęć powyżej)
- Skalowanie Scrum wymaga wprowadzenia praktyk koordynowania równoległych prac nad rozwojem rozwiązania (np. Scrum of Scrums) oraz wyznaczenie dodatkowych ról zapewniających jednoosobowe właścicielstwo pewnych obszarów (np. właściciel systemu), aby łatwiej przeprowadzać zmiany w odpowiednich aplikacjach

SAFe

- SAFe jest rozbudowaną metodą, która z jednej strony może wydawać się odpowiednia dla niektórych organizacji, z drugiej jednak niektórzy praktycy metodyk zwinnych uważają, że odchodzi ona za daleko od zasad zwinnych (np. w kwestiach samoorganizacji) na rzecz utrzymania kontroli
- SAFe przewiduje trzy poziomy:
 - Zespół, zorganizowany podobnie jak w Scrum
 - Program, łączący fragmenty produktów poszczególnych zespołów we wspólną wersję zwaną Trainem
 - Portfolio, odpowiedzialne za wizję, wymagania, architekturę oraz budżet

LeSS

- LeSS (Large-Scale Scrum) jest metodą uważaną za lżejszą w porównaniu do SAFe
- Autorzy metody wprowadzają tylko niezbędne zmiany do Scrum, tak, aby nie zmniejszać „zwinności” podejścia
- Są dwie odmiany tej metody – dla mniejszych programów (do 10 zespołów Scrum) oraz dla większych inicjatyw
- Marketing i formalizacja metody (certyfikacja, szkolenia) na moment pisania tego materiału nie są jeszcze rozbudowane

Scrum w dużych organizacjach lista kontrolna

Na podstawie naszych doświadczeń w pracy z dużymi organizacjami IT przygotowaliśmy listę punktów kontrolnych, które według nas są krytyczne dla udanego wdrożenia Scrum.

Poniższa lista nie ma na celu zaadresowania wszystkich aspektów w złożonej strukturze organizacji, ale jest pomocna jako szybki test, jak uniknąć najczęściej popełnianych błędów.

dłaczego trudniej jest..

..wdrożyć Scrum w dużej organizacji niż w małej, niezależnej firmie?

- menedżerowie przywykli do pracy w strukturze silosowej i **obwiniania innych jednostek w przypadku niepowodzeń** (IT vs. biznes), zamiast dążyć do wspólnych celów
- pracownicy niechętnie wychodzą **poza swoje role**, podczas gdy członkowie zespołów Scrumowych powinni być wielozadaniowi
- poruszanie się w złożonej strukturze organizacyjnej i równoległych projektach **jest częściej bardziej skomplikowane niż sam produkt** i wymaga efektywnej komunikacji bardziej niż technicznych umiejętności
- programiści muszą korzystać z zastanych rozwiązań i pracować zgodnie ze **standardami wypracowanymi przez organizację**, co spowalnia tempo przyswajania wiedzy i produkcji rozwiązania

1. skład zespołu

stwórz właściwy mix członków zespołu –nie tylko programiści.

- a. jest jasno określony Właściciel Produktu
- b. Właściciel Produktu jest odpowiednio umocowany i może podejmować wszystkie decyzje dotyczące zakresu i budżetu
- c. Właściciel Produktu jest w 100% zaangażowany w projekt lub wyznaczył swojego pełnomocnika (np. analityka biznesowego)
- d. Scrum Master jest w pełni zaangażowany w projekt
- e. przynajmniej jeden członek zespołu ma duże doświadczenie w projektach zwinnych, aby dzielić się wiedzą i służyć wsparciem

2. gotowość zespołu

zapoznaj wszystkich członków zespołu z ich rolami w projekcie.

- a. wszyscy członkowie zespołu, łącznie z Właścicielem Produktu, odbyli podstawowe szkolenie ze Scrum
- b. Właściciel Produktu jest świadomy, że początkowe dema będą odbiegały od finalnego rozwiązania
- c. Właściciel Produktu został przeszkolony ze stałego kontrolowania budżetu i harmonogramu projektu
- d. zarówno biznesowi jak i techniczni członkowie zespołu opisują projekt jako „nasz” a nie „ich”
- e. członkowie zespołu są zachęceni do uczestniczenia we wszystkich typach zadań (analiza, kodowanie, testy)
- f. Scrum Master nie jest kierownikiem projektu, lecz członkiem zespołu, odpowiadającym za usuwanie przeszkód w projekcie

3. przygotowanie projektu

w złożonej organizacji niektóre elementy lepiej przygotować z wyprzedzeniem.

- a. wszyscy członkowie zespołu pracują w jednej lokalizacji, a jeśli nie, to zapewnione są rozwiązania do teleobecności
- b. zespół ma swój własny pokój projektowy
- c. rejestr produktu jest gotowy przed pierwszą iteracją
- d. istnieje wysokopoziomowa koncepcja/projekt/architektury rozwiązania, które ma zostać stworzone
- e. środowiska, łącznie ze środowiskiem demo, powstały przed rozpoczęciem projektu
- f. dostępne jest dedykowane narzędzie do zarządzania rejestrem produktu dostępne dla wszystkich członków zespołu, wraz z Właścicielem Produktu

4. realizacja projektu

upewnij się, że zespół przestrzega kluczowych praktyk.

- a. Właściciel Produktu ma stałą kontrolę nad rejestrem produktu
- b. Każdy sprint kończy się prezentacją demo
- c. Właściciel Produktu uczestniczy w planowaniu sprintu i prezentacjach demo
- d. w trakcie planowania sprintu, wszystkie historyjki zostają wycenione
- e. oszacowania są porównywane z rzeczywistą pracochłonnością: wszyscy członkowie zespołu zapisują przepracowany czas
- f. Właściciel Produktu lub jego pełnomocnik bierze udział w retrospekcji, aby na bieżąco rozwiązywać problemy

Scrum w dużych organizacjach rozwiązywanie problemów

Poniżej znajdują się typowe problemy, które mogą się pojawić w trakcie wprowadzania Scrum, wraz z przykładowymi rozwiązaniami.

Rozwiązywanie problemów

Ogólne zasady rozwiązywania problemów przy wprowadzaniu Scrum:

- wprowadzenie Scrum polega bardziej na zmianie kultury organizacji niż na przedstawieniu nowej metodyki, dlatego też rozwiązując problem w pierwszej kolejności należy szukać rozwiązania miękkiego, zamiast ulegać pokusie formalizacji i komplikowania procesów
- należy pamiętać, że decyzja o wdrożeniu Scrum była podjęta z myślą o konkretnych celach, dlatego nie obawiaj się dostosowywać jego praktyk do potrzeb organizacji, jeśli jest to korzystne z punktu widzenia celu
- zainwestuj w intensywne szkolenia swojego zespołu na początkowym etapie wprowadzania Scrum – postawa i zaangażowanie zespołu jest najważniejszym czynnikiem sukcesu

1. Właściciel Produktu nie spełnia swojej roli – decyzje biznesowe są podejmowane za późno lub są zmieniane, a stan rozwoju aplikacji nie jest regularnie sprawdzany

- jeśli Właściciel Produktu nie może poświęcić projektowi 100% swojego czasu (lub gdy 100% nie jest wystarczające), wyznacz jego pełnomocnika, który będzie w stanie przejąć 80% zaangażowania wymaganego przez programistów, oszczędzając tym samym czas Właściciela Produktu na ważne decyzje i dema
- zadbaj o to, aby w zespole znajdowali się członkowie z doświadczeniem w analizie biznesowej, tak, aby nie obciążać Właściciela Produktu zadaniami wymagającymi rozległej analizy

2. Programiści nie mogą dotrzeć do wiedzy i dokumentacji (np. o interfejsach) dotyczących innych systemów wykorzystywanych w organizacji i tracą czas na zbędną komunikację i oczekiwanie

- sprawdź, czy Scrum Master spełnia swoje zadania: jego rolą jest rozwiązywanie problemów programistów, aby ci mogli wydajnie pracować; być może Scrum Master nie jest w 100% oddelegowany do projektu, jest zajęty innymi zadaniami lub nie ma wystarczającego doświadczenia, aby szybko zaadresować problemy
- upewnij się, że co najmniej jeden członek zespołu ma bezpośredni dostęp do kluczowych ekspertów spoza zespołu (pracuje w tej samej lokalizacji) i może dotrzeć do nich szybko, uzyskując informacje w nieformalny sposób
- zweryfikuj, czy członkowie zespołu mają odpowiednie doświadczenie w analizie biznesowej, tak, aby mogli pokierować organizacją i zdobyć dostęp do wiedzy szybciej niż programiści

3. Wyprodukowane oprogramowanie jest niskiej jakości, co przejawia się w częstym występowaniu błędów z wcześniejszych sprintów

- zweryfikuj, czy istnieje wysokopoziomowy projekt rozwiązania oraz zasady programowania i czy są one znane członkom zespołu
- sprawdź, czy refaktoring kodu jest uwzględniony w wycenach
- sprawdź, czy testy jednostkowe są uwzględnione w wycenach
- w przypadku gdy zespół nie pracuje w jednej lokalizacji, zastanów się nad oddelegowaniem doświadczonego członka zespołu do miejsca, gdzie występują problemy
- zorganizuj zespół w taki sposób, aby testy mogły być wykonywane w sposób ciągły i równoległy do kodowania, a nie tylko tuż przed końcem sprintu lub projektu

4. Dema ze sprintów nie odbywają się, gdyż „nie ma wystarczająco dużo nowych rzeczy do zaprezentowania” lub pojawiają się trudności organizacyjne (nie działa środowisko, Właściciel Produktu nie jest dostępny)

- upewnij się, że zespół podzielił wymagania biznesowe na niewielkie historyjki tak, aby ich realizacja zajmowała maksymalnie jeden sprint
- sprawdź, czy są dostępne stabilne środowiska, na których można zaprezentować ukończone historyjki
- zweryfikuj, czy Właściciel Produktu ściśle współpracuje z programistami i czy bierze udział w prezentacjach rozwiązania

Wszystkie 12 problemów dostępne są w wersji pełnej. Skontaktuj się z nami, aby ją otrzymać.

Scrum w dużych organizacjach [skontaktuj się z nami](#)

wersja pełna

jeśli wprowadzasz Scrum w swojej organizacji, skontaktuj się z nami, aby otrzymać pełną wersję:

contact@goldenberry.eu

Wersja pełna obejmuje między innymi:

- pozostałe opisy problemów i ich rozwiązań
- opis Sprintu 0
- porównanie elementów Scrum w małych i dużych organizacjach
- pożądaný profil Scrum Mastera
- pożądaný profil Właściciela Produktu
- przykłady narzędzi wykorzystywanych w Scrum

o goldenberry

Goldenberry jest firmą doradcą w obszarze zarządzania i IT.

Wspieramy organizacje w ich najbardziej wymagających inicjatywach, w tym w transformacjach IT.

Potrzebujesz wsparcia w projekcie Scrum? Skontaktuj się z nami, aby porozmawiać o tym jak możemy pomóc.

autorzy

Dina Dąbrowska associate

dina.dabrowska@goldenberry.eu

Bartłomiej Owczarek partner

bartlomiej.owczarek@goldenberry.eu

wersja dokumentu 1.1, październik 2014.

Metropolitan
Pl. Piłsudskiego 3
00-078 Warsaw

tel. +48 22 449 00 50

Berkeley Square House, 2nd floor
Berkeley Square
W1J 6BD London

tel. +44 20 7887 6067

www.goldenberry.eu