

# Przykładowa lista kontrolna dla ScrumMasterów.

Autor: Michael James ([mj4scrum@gmail.com](mailto:mj4scrum@gmail.com))

Tłumaczenie na język polski: Bogdan Brześciński ([Bogdan.Brzescinski@procognita.pl](mailto:Bogdan.Brzescinski@procognita.pl))

Korekta: Tomasz de Jastrzębiec Wykowski ([Tomasz.Wykowski@procognita.pl](mailto:Tomasz.Wykowski@procognita.pl))

20 czerwca 2015

<http://scrummasterchecklist.org/>

## Moderator na pełny etat?

Przeciętny ScrumMaster może pracować z dwoma lub trzema zespołami jednocześnie. Jeżeli odpowiada ci ograniczenie swojej roli do organizowania spotkań, egzekwowania ram czasowych i reagowania na trudności zgłaszane przez pracowników, możesz osiągnąć to pracując na pół etatu. Zespół prawdopodobnie wciąż przekroczy oczekiwania Twojej organizacji sprzed wprowadzenia Scruma i raczej nie wydarzy się żadna katastrofa.

Jeżeli jednak marzy ci się zespół, który osiąga cele, wcześniej uważane za niemożliwe w transformującej się organizacji, rozważ zostanie *świątecznym* ScrumMasterem.

Świąteczny ScrumMaster może współpracować z *jednym* Zespołem w danej chwili.

Zalecamy jednego ScrumMastera dedykowanego do każdego Zespołu, składającego się z około siedmiu osób. Zwłaszcza na początku.

Jeśli nie dowiedziałeś się jeszcze jak dużo jest do zrobienia, popatrz na Twojego Product Ownera, Twój zespół, jego praktyki inżynieryjne, oraz całą organizację na zewnątrz zespołu. Chociaż nie ma jednej recepty dla wszystkich, nakreśliłem typowe tematy, których ScrumMasterzy często nie dostrzegają. Proszę, oznacz każdy kwadrat za pomocą  $\checkmark$ ,  $\Delta$ ,  $?$ , lub N/A, zgodnie z opisem zawartym na ostatniej stronie.

## Część I – Jak sobie radzi mój Product Owner?

ScrumMasterzy zwiększają efektywność Product Ownerów pomagając im znaleźć sposoby na utrzymanie Rejestru Produktowego i Planu Wydań. (Zauważ, że tylko Product Owner może ustalać kolejność Elementów Rejestru Produktowego)

- Czy Rejestr Produktowy jest spriorytetyzowany zgodnie z jego/jej ostatnim zamysłem?
- Czy wymagania i oczekiwania od wszystkich interesariuszy zostały ujęte w Rejestrze Produktowym? Pamiętaj: Rejestr jest żywym dokumentem i ciągle ewoluuje.
- Czy Rejestr Produktowy jest akceptowalnych rozmiarów? Aby osiągnąć zarządzalną liczbę elementów, utrzymuj niewielkie, bardziej szczegółowe elementy na górze Rejestru a duże idee na dole. Nadmierna analiza elementów znajdujących się daleko od góry Rejestru jest nieefektywna. Twoje wymagania będą się zmieniały na bieżąco w trakcie dyskusji pomiędzy tworzącymi produkt a interesariuszami i klientami.

- Czy którekolwiek z wymagań (zwłaszcza te najbliższe góry Rejestru Produktowego) mogą być lepiej wyrażone jako niezależne, negocjowalne, wartościowe, dające się oszacować, małe i testowalne Historie Użytkownika<sup>1</sup>?
- Czy wytłumaczyłeś Product Ownerowi co to jest *Dług Techniczny* i jak go uniknąć? Ważnym elementem tej układanki może być dołączenie automatycznych testów i refaktoryzacji do Definicji Ukończenia dla każdego elementu Rejestru Produktowego.
- Czy Rejestr Produktowy jest „*Promiennikiem Informacji*” pozwalającą wszystkim interesariuszom na natychmiastowy dostęp do informacji?
- Jeżeli używasz zautomatyzowanego narzędzia do zarządzania Rejestrem Produktowym, to czy wszyscy wiedzą jak łatwo z niego korzystać? Narzędzia do automatycznego zarządzania niosą ze sobą ryzyko stworzenia „*Zamrażalnika Informacji*”<sup>2</sup>, jeżeli ScrumMaster nie zapewni odpowiedniego promieniowania / naświetlania.
- Czy potrafisz naświetlić informacje pokazując wydruki?
- Czy potrafisz naświetlić informacje tworząc duże, widoczne wykresy?
- Czy pomogłeś Product Ownerowi przyporządkować elementy w Rejestrze Produktowym do odpowiednich Wydań lub grup priorytetowych?
- Czy wszyscy znają faktyczny stan Planu Wydania? Możesz spróbować pokazać Wykresy Spalania<sup>3</sup>, po tym jak elementy zostaną uznane za „Ukończone” podczas Przeglądu Sprintu. Wykresy pokazujące jednocześnie tempo ukończenia Elementów Rejestru Produktowego przez zespół jak i dodawania nowych elementów pozwalają na wcześniejsze wykrycie zmiany zakresu i harmonogramu.
- Czy Twój Product Owner zaktualizował Plan Wydania po ostatnim Przeglądzie Sprintu? Ci z Product Ownerów, którzy dostarczają odpowiednio przetestowany produkt na czas, poświęcają czas na ponowne planowanie Wydania po każdym Sprincie. Często wymaga to zmiany priorytetów i przesunięcia części pracy do następnych wydań.

## Część II – Jak sobie radzi mój zespół?

Podczas gdy jesteś zachęcany, aby dawać dobry przykład współpracując z członkami zespołu nad ich zadaniami, istnieje ryzyko, że skupisz się za bardzo na tematach technicznych. Dlatego weź pod uwagę Twoje podstawowe obowiązki w stosunku do zespołu:

- Czy Twój zespół jest w stanie *przeptywu*? Poniżej kilka objawów tego stanu<sup>4</sup>:
  - Jasno określone cele (oczekiwania i reguły są znane, a cele osiągalne i dostosowane do umiejętności danego członka zespołu).
  - Koncentracja i skupienie – wysoki poziom koncentracji na ograniczonym polu uwagi.
  - Utrata poczucia samoświadomości; połączenie działań oraz gotowości.
  - Konkretna i natychmiastowa informacja zwrotna (sukcesy i porażki są widoczne w trakcie działania, co pozwala odpowiednio na nie reagować).
  - Równowaga pomiędzy poziomem umiejętności a wymaganiami (zadanie nie jest zbyt łatwe ani za trudne).
  - Poczucie panowania nad sytuacją lub aktywnością.
  - Działanie daje wewnętrzną satysfakcję, więc nie sprawia wysiłku.

<sup>1</sup> <http://xp123.com/articles/invest-in-good-stories-and-smart-tasks/>

<sup>2</sup> <http://c2.com/cgi/wiki?InformationRefrigerator>

<sup>3</sup> Mike Cohn, *Agile Estimation and Planning*. (2005).

<sup>4</sup> Mihaly Csikszentmihalyi, *Flow: The Psychology of Optimal Experience* (1990), wyd. polskie: *Przeptyw* (2008)

- Czy członkowie Zespołu lubią się wzajemnie, spędzają czas razem i wspólnie świętują swoje sukcesy?
- Czy członkowie Zespołu wzajemnie pilnują utrzymywania wysokich standardów? Czy stawiają przed sobą wyzwania prowadzące do rozwijania swoich umiejętności?
- Czy są problemy lub możliwości, których omawiania Zespół unika, ponieważ powodują zbyt duży dyskomfort?<sup>5</sup>
- Czy próbowałeś różnych sposobów oraz miejsc dla przeprowadzanych Retrospektyw Sprintu?<sup>6</sup>
- Czy Zespół skupia się na Celu Sprintu? Być może powinieneś w połowie Sprintu pomóc Zespołowi przeglądnąć kryteria akceptacyjne dla Elementów Rejestru Produktu wybranych do obecnego Sprintu.
- Czy Tablica Zadań Sprintu odzwierciedla to, co zespół faktycznie robi? Wystrzegaj się „czarnej materii” ukrytych zadań i takich, które zajmują więcej niż jeden dzień pracy. Zadania nie związane ze zobowiązaniami zespołu na obecny Sprint stanowią przeszkodę dla tych zobowiązań.
- Czy Twój Zespół składa się z 3-9 osób posiadających wszystkie umiejętności, niezbędne do zbudowania Przyrostu Produktu, który potencjalnie można wdrożyć po każdym Sprincie?
- Czy Tablica Zadań Twojego Zespołu jest na bieżąco aktualizowana?
- Czy artefakty samorządzącego się Zespołu (Tablica Zadań, Wykres Spalania Sprintu, Lista Przeszkód, itd.) są widoczne dla członków zespołu i wygodne w użyciu?
- Czy te artefakty są odpowiednio chronione przed osobami z zewnątrz? Nadmierna kontrola codziennej aktywności przez ludzi z zewnątrz może utrudniać wewnętrzną przejrzystość i samorządzenie.
- Czy członkowie Zespołu sami zgłaszają się do wybranych zadań?
- Czy potrzeba spłaty Długu Technicznego została ujęta w Definicji Ukończenia, stopniowo czyniąc kod coraz bardziej przyjaznym w pracy?
- Czy członkowie Zespołu zostawiają swoje tytuły i stanowiska za drzwiami, wspólnie odpowiadając za wszystkie aspekty ustalonej pracy (testowanie, prowadzenie dokumentacji, itd.)?

### Część III – Jak wyglądają nasze praktyki inżynierskie?

- Czy wasze środowisko deweloperskie posiada guzik “naciśnij, żeby przetestować”, pozwalający każdemu (z tego lub innego zespołu) na wygodne sprawdzenie, że nie spowodował błędu regresji (nie uszkodził wcześniej działającej funkcjonalności)? Zwykle jest to osiągnięte poprzez wykorzystanie środowisk xUnit (JUnit, NUnit, itp.).
- Czy utrzymujecie odpowiednią równowagę pomiędzy zautomatyzowanymi testami systemowymi end-to-end (testy funkcjonalne) oraz zautomatyzowanymi testami jednostkowymi?
- Czy zespół pisze zarówno testy funkcjonalne jak i testy jednostkowe w tym samym języku w którym się tworzą produkt? Fakt, że tylko część zespołu potrafi obsługiwać język skryptowy lub narzędzia do nagrywania testów nie poprawia współpracy.
- Czy Twój zespół odkrył użyteczną szarą strefę pomiędzy testami systemowymi a testami jednostkowymi?<sup>7</sup>

<sup>5</sup> Kerry Patterson, *Crucial Conversations: Tools for Talking When Stakes are High* (2002). Weź również pod uwagę zaproszenie profesjonalnego moderatora, który może uczynić trudne rozmowy bardziej komfortowymi.

<sup>6</sup> Derby/Larson *Agile Retrospectives: Making Good Teams Great* (2006).

<sup>7</sup> <http://blogs.collab.net/agile/2007/03/07/junit-is-not-just-for-unit-testing-anymore/>

- Czy system *Ciągłej Integracji*<sup>8</sup> automatycznie alarmuje, gdy ktoś spowoduje błąd regresji? Czy ta pętla informacji zwrotnej może zostać skrócona do godzin lub minut? („Codzienne kompilacje są dla mięczaków” – Kent Beck).
- Czy *wszystkie* testy są częścią Ciągłej Integracji?
- Czy członkowie Zespołu odnaleźli wartość ciągłego projektowania i refaktoryzacji<sup>9</sup>, jako alternatywę do podejścia „zaprojektuj wszystko na początku”? Refaktoryzacja posiada ścisłą definicję: zmienianie wewnętrznej struktury bez zmieniania zewnętrznych zachowań. Refaktoryzacja powinna mieć miejsce kilka razy na godzinę, za każdym razem, gdy kod jest powielony, występuje złożona logika warunkowa (rozpoznawalna przez dużą ilość wcięć lub długość funkcji), kiepsko nazwane identyfikatory, nadmierne sprzężenia pomiędzy obiektami, itp. Bezpieczną refaktoryzację umożliwi tylko dobre pokrycie kodu testami automatycznymi. Zaniedbanie refaktoryzacji utrudnia późniejsze wprowadzanie zmian do produktu, zwłaszcza, że trudno znaleźć dobrych deweloperów chętnych do pracy na złym kodzie.
- Czy twoja Definicja Ukończenia dla każdego Elementu Rejestru Produktowego zawiera pokrycie kodu automatycznymi testami i refaktoryzację? Korzystanie z Test Driven Development (TDD) zwiększa prawdopodobieństwo osiągnięcia tego punktu.
- Czy członkowie zespołu programują w parach przez większość czasu? Używanie tej metody może znacząco ułatwić utrzymywanie kodu oraz zmniejszyć ilość błędów. Ponieważ stawia to ludziom nowe wyzwania czasem może wyglądać, że zabiera więcej czasu (jeżeli mierzymy ilość linii kodu, a nie gotową funkcjonalność). Daj dobry przykład programując w parze z kolejnymi członkami zespołu. Niektórzy z nich zaczną korzystać z tej techniki na stałe.

#### Część IV – Jak radzi sobie organizacja?

- Czy Zespoły komunikują się ze sobą w wystarczający sposób? „Scrum-of-Scrums” jest tylko jednym ze sposobów na osiągnięcie tego stanu. Nie koniecznie najlepszym.
- Czy Zespoły mogą niezależnie od siebie dostarczać działającą funkcjonalność, modyfikując różne elementy architektury jeżeli jest taka potrzeba?<sup>10</sup>
- Czy Wasi ScrumMasterowie spotykają się razem i pracują nad listą utrudnień w organizacji?
- Jeżeli jest taka możliwość, czy utrudnienia organizacyjne są wywieszane na ścianie gabinetu dyrektora d/s tworzenia oprogramowania? Czy ich koszt może być przedstawiony w gotówce, utraconym czasie lub jakości, albo utraconych szansach na pozyskanie klienta (ale ucz się na błędach Ken’a Schwabera – „Martwy ScrumMaster to bezużyteczny ScrumMaster”)<sup>11</sup>
- Czy Twoja organizacja oferuje ścieżkę kariery zgodną ze zbiorowymi celami Twojego Zespołu? Odpowiedz „nie” jeżeli promuje<sup>12</sup> programowanie kosztem automatyzacji testów lub przygotowania dokumentacji dla użytkownika.
- Czy Twoja organizacja została uznana przez prasę branżową lub inne niezależne źródło jako jedno z najlepszych miejsc pracy lub lidera w swojej branży?
- Czy tworzysz organizację *uczącą się*?

<sup>8</sup> <http://www.martinfowler.com/articles/continuousIntegration.html>

<sup>9</sup> Martin Fowler, *Refactoring: Improving the Design of Existing Code* (1999).

<sup>10</sup> <http://FeatureTeamPrimer.org/>

<sup>11</sup> Ken Schwaber, *Agile Project Management with Scrum* (2004)

<sup>12</sup> Alfie Kohn, *Punished By Rewards: The Trouble with Gold Stars, Incentive Plans, A's, Praise, and Other Bribes* (1999)

## **Wnioski**

Jeżeli możesz zaznaczyć wszystkie punkty z tej listy i wciąż masz czas w ciągu dnia, chciałbym Cię poznać.

Nie ma gotowej recepty na wzbudzenie ludzkiej pomysłowości. Ten dokument zawiera listę punktów, które mogą, lecz nie koniecznie muszą, pomóc w Twojej sytuacji.

Kiedy zdasz sobie sprawę co możesz zrobić, aby coś zmienić, możesz się zorientować, że boisz się zmiany. Będzie to znak że jesteś na dobrej drodze.

## **Formularz Przeszkód w Organizacji**

**Widoczny Problem:**

**Prawdziwa Przyczyna (użyj „5 x dlaczego?”):**

**Wpływ na Firmę:**

**Wpływ Emocjonalny:**

**Jasne Żądanie:**

---

## **Formularz Przeszkód w Organizacji**

**Widoczny Problem:**

**Prawdziwa Przyczyna (użyj „5 x dlaczego?”):**

**Wpływ na Firmę:**

**Wpływ Emocjonalny:**

**Jasne Żądanie:**

## **Formularz Przeszkód w Organizacji**

**Widoczny Problem:**

**Prawdziwa Przyczyna (użyj „5 x dlaczego?”):**

**Wpływ na Firmę:**

**Wpływ Emocjonalny:**

**Jasne Żądanie:**

---

## **Formularz Przeszkód w Organizacji**

**Widoczny Problem:**

**Prawdziwa Przyczyna (użyj „5 x dlaczego?”):**

**Wpływ na Firmę:**

**Wpływ Emocjonalny:**

**Jasne Żądanie:**

## **Formularz Przeszkód w Organizacji**

**Widoczny Problem:**

**Prawdziwa Przyczyna (użyj „5 x dlaczego?”):**

**Wpływ na Firmę:**

**Wpływ Emocjonalny:**

**Jasne Żądanie:**

---

## **Formularz Przeszkód w Organizacji**

**Widoczny Problem:**

**Prawdziwa Przyczyna (użyj „5 x dlaczego?”):**

**Wpływ na Firmę:**

**Wpływ Emocjonalny:**

**Jasne Żądanie:**



## INSTRUKCJA

Jeżeli otrzymałeś ten dokument jako element szkolenia a Twój pracodawca stosował Scruma lub metodologie pokrewne, proszę odnieść się do tego, co tam widziałeś. Zaznacz każdą kratkę jednym z poniższych symboli:

- √ (idzie nam całkiem nieźle)
- Δ (może być ulepszone i wiem jak zacząć)
- ? (może być ulepszone ale nie wiem jak zacząć)
- N/A (nie dotyczy/nie przyniesie żadnych korzyści)

Jeżeli Twój obecny lub poprzedni pracodawca nie stosował Scruma lub metodologii pokrewnych, zaznacz każdą kratkę jednym z poniższych symboli:

- √ (idzie nam całkiem nieźle/łatwe do osiągnięcia)
- Δ (osiągnięcie będzie wyzwaniem, ale wiem jak zacząć)
- ? (osiągnięcie będzie wyzwaniem i nie wiem jak zacząć)
- N/A (nie dotyczy/nie przyniesie żadnych korzyści)

Gdy wypełnisz całą listę, określ 2-6 przeszkód w Twojej organizacji na załączonych formularzach, niezależnie od tego, czy znajdują się na liście kontrolnej. Wybieraj takie przeszkody, które mają chociażby 1% szans na usunięcie.

# Słownik Polsko-Angielski sformułowań Agile

Ponieważ nie ma wciąż ogólnie przyjętych polskich tłumaczeń angielskich zwrotów Agile, poniżej zamieszczamy nasze wersje, które stosowaliśmy w powyższym dokumencie. Pełny słownik dostępny na: <http://procognita.pl/zasoby/artykuly/czytaj/artykul/angielsko-polski-subiektywny-slowniczek-zwrotow-agile-i-lean-57/>

Ciągła Integracja – Continuous Integration (CI)

Definicja Ukończenia – Definition of Done (DoD)

Element Rejestru Produktu – Product Backlog Item (PBI)

Plan Wydania – Release Plan

Potencjalnie Wdrażalny – Potentially Shippable

Promiennik Informacji – Information Radiator

Przegląd Sprintu – Sprint Review

Przeszkody – Impediments

Przyrost Produktu – Product Increment

Rejestr Produktowy – Product Backlog (PB)

Retrospektywa Sprintu – Sprint Retrospective

Samoorganizacja – Self-organization

Tablica Zadań – Task Board

Testy Akceptacyjne – Acceptance Tests

Testy Automatyczne – Automated Tests

Testy Jednostkowe – Unit Tests

Wykres Spalania Sprintu – Sprint Burndown

Wykresy Spalania Wydania – Release Burndown

Zamrażalnik Informacji – Information Refrigerators

Zobowiązanie – Commitment

W trakcie tłumaczenia zastosowaliśmy męskie wersje czasowników jedynie dla ułatwienia czytania. Oczywiście powyższa lista kontrolna może być używana przez przedstawicieli dowolnej płci.

Jeżeli masz jakiegokolwiek sugestie na temat zawartości listy, albo tłumaczenia, będę wdzięczny, jeżeli do mnie napiszesz na [Bogdan.Brzescinski@procognita.pl](mailto:Bogdan.Brzescinski@procognita.pl)