

Inżynieria oprogramowania

Zarządzanie projektem

Arkadiusz Chrobot

Katedra Systemów Informatycznych, Politechnika Świętokrzyska

Kielce, 6 listopada 2024

Notatki

1 / 29

Plan

Notatki

Motto

Wprowadzenie

Scrum

Planowanie

Zarządzenie ryzykiem

2 / 29

Motto

Notatki

„W jaki sposób projekt może się spóźnić o cały rok?
...dzień po dniu.”

Fred Brooks Jr. „*Legendarny osobomiesięc: opowieści o inżynierii oprogramowania*”

3 / 29

Wprowadzenie

Notatki

Zarządzenie projektem informatycznym to trudne zadanie, z wielu powodów. Jednym z nich jest charakter produktu końcowego. Oprogramowanie nie ma fizycznej, namacalnej postaci. Jest ono czystą informacją. Taka natura oprogramowania znacznie utrudnia śledzenie postępu prac nad nim. To oznacza, że konieczne są specjalne metody zarządzania projektami informatycznymi.

W ramach tego wykładu zostanie opisana popularna metoda zarządzania projektem, o nazwie *Scrum* oraz przedstawione będą sposoby zarządzania ryzykiem i planowania stosowane zarówno w tradycyjnych, jak i zwinnych metodach.

4 / 29

Scrum została opracowana we wczesnych latach 1990. przez Kena Schwabera i Jeffa Sutherlanda. Inspiracją dla jej powstania był System Produkcji Toyoty, który z kolei powstał na bazie prac dra Williama Edwardsa Deminga. Scrum jest *planem ramowym* (ang. *framework*) dla zarządzania projektami i rozwiązywania złożonych problemów. Jej użycie nie ogranicza się wyłącznie do inżynierii oprogramowania. Szczegółowy opis tej metody znajduje się w *Przewodniku po Scrumie*, który dostępny jest za darmo zarówno po [angielsku](#), jak i w kilku innych wersjach [językowych](#), w tym po polsku. Proszę zwrócić uwagę, że Scrum **nie jest** kompletnym przepisem na zarządzaniem projektem, a tylko zbiorem wskazówek pozwalających zorganizować pracę nad produktem. Musi ona zostać dostosowana do potrzeb zespołu wytwórczego.

Zasady Scrum

Scrum bazuje na trzech zasadach:

- Przejrzystość** Proces tworzenia produktu musi być widoczny zarówno dla ludzi, którzy wykonują tę pracę, jaki i dla tych, którzy czekają na jej wynik. Ważne decyzje są podejmowane zgodnie z postrzeganym stanem trzech formalnych artefaktów, które zostaną opisane później. Przejrzystość (ang. *transparency*) pozwala zespołowi na kontrolę (ang. *inspection*) jego pracy.
- Kontrola** Artefakty Scrum i postęp, zmierzający do celu projektu, muszą często i dokładnie być badane (ang. *inspect*), aby wykryć wszystkie problemy. Kontrole pozwalają dostosować (ang. *adapt*) istniejące procesy do napotkanych problemów.
- Adaptacja** Jeśli jakiś aspekt procesu lub produktu nie ma wymaganej i założonej jakości, to cały proces musi zostać usprawniony najszybciej jak się da, aby zapobiec dalszym stratom.

Wartości Scrum

- Zakłada się, że osoby praktykujące z sukcesem Scrum są:
 - zaangażowane** (ang. *commitment*) by osiągnąć cele projektu i wzajemnie się w tym dążeniu wspierać,
 - skupione** (ang. *focus*) aby jak najbardziej zbliżyć się do celów projektu,
 - otwarte** (ang. *openness*) na dyskusję o ich pracy i napotkanych trudnościach,
 - pelne szacunku** (ang. *respect*) dla umiejętności i niezależności każdej z nich,
 - odważne** (ang. *courage*) w radzeniu sobie ze złożonymi problemami.

Zespół Scrum

- Zespół Scrum powinien składać się z nie więcej niż 10 osób. Ich role powinny być następujące:
 - Scrum master** jest odpowiedzialny za wprowadzenie i dbanie o przestrzeganie zasad Scrum w zespole, a także za usuwanie wszelkich przeszkód, które mogłyby zaszkodzić produktywności.
 - Product owner** jest odpowiedzialny za maksymalizację wartości produktu wynikającej z pracy zespołu Scrum, zarządzanie artefaktem *Product backlog*, wskazanie *celu produktu* (ang. *Product goal*) i reprezentowanie potrzeb interesariuszy (ang. *stakeholders*).
 - Wytwórcy** (ang. *developers*) to samoorganizujący się, samoorganizujący się profesjonaliści, o różnych umiejętnościach, niezbędnych do wytworzenia wartości w każdym *sprint*.

Praca w Scrum jest wykonywana w sposób iteracyjny i przyrostowy. Jedna iteracja nazywa się *sprintem*. Maksymalny czas trwania sprintu to miesiąc, ale zwykle trwa on krócej, na ogół dwa tygodnie lub nawet mniej. W trakcie trwania sprintu nie mogą zostać wprowadzone żadne zmiany, które mogłyby zagrozić jego celowi (ang. *Sprint Goal*) lub obniżyć jakość produktu. W porozumieniu z Product ownerem zawartość artefaktu *Product backlog* może być uszczegółowiona, a zakres prac renegocjowany i doprecyzowany, jeśli wytwórcy uzyskają nowe informacje na temat budowanego produktu. Sprint może zostać przerwany przez Product ownera, jeśli jego cel przestanie być ważny.

Każdy sprint składa się z czterech rodzajów *wydarzeń*: *planowania sprintu* (ang. *Sprint Planning*), *codziennego Scrumu* (ang. *Daily Scrum*), *przeglądu sprintu* (ang. *Sprint Review*) i *retrospekcji sprintu* (ang. *Sprint Retrospective*).

Wydarzenia Sprintu

W trakcie *planowania sprintu* zespół Scrum określa *cel sprintu* (ang. *Sprint Goal*) wskazując jaką wartość dla interesariuszy przyniesie ten sprint. Podejmowana jest też decyzja, które zadania z *Product backlog* mogą być zrealizowane w planowanym sprincie. Następnie wytwórcy definiują co musi być zrobione, aby wybrane zadania dały wartościowy przyrost. *Codzienny Scrum* jest spotkaniem pozwalającym wytwórcom zbadać postęp i dostosować *Sprint backlog*, jeśli jest to konieczne. Zgodnie z nazwą takie spotkania odbywają się codziennie i nie powinny zajmować więcej niż 15 minut. *Przegląd Sprintu* odbywa się na jego zakończenie. Celem tego wydarzenia jest zbadanie wyników sprintu i określenie dalszego kierunku prac. Rezultaty są prezentowane najważniejszym interesariuszom i omawiany jest postęp w kierunku osiągnięcia *celu projektu*. Dla miesięcznego sprintu to spotkanie nie powinno trwać dłużej niż 4 godziny. W przypadku krótszych sprintów powinno ono także trwać krócej. *Retrospekcja sprintu* również odbywa się na jego zakończenie. Jej celem jest znalezienie sposobu na zwiększenie jakości produktu i efektywności pracy. W trakcie tego wydarzenia badany jest ostatni sprint, określane są wszystkie znaczące problemy i wdrażane są środki zapobiegawcze — mogą one być nawet dodane do *Sprint backlog* dla następnego sprintu.

Artefakty Scrum

Istnieją trzy artefakty Scrum:

Product backlog to uporządkowana lista zadań (ang. *item*) koniecznych do udoskonalenia produktu. Zadanie określa pracę do wykonania. Zadania w liście mogą być dodawane, usuwane lub modyfikowane. Zmiana polega na podziale dużych zdań na małe, bardziej precyzyjne. Zadania są przenoszone z *Product backlog* do *Sprint backlog* jeśli mogą być zrealizowane w trakcie pojedynczego sprintu. Decyzje o tym podejmują wytwórcy, określając wcześniej rozmiar poszczególnych zadań. *Cel produktu* (ang. *product goal*) definiuje zamierzony stan końcowego produktu i przyświeca pracy całego zespołu.

Sprint backlog zawiera *cel sprintu* (ang. *sprint goal*), plan dostarczenia *przyrostu* (ang. *increment*) i zadania wybrane z *Product backlog*. *Sprint backlog* jest tworzony i zarządzany przez wytwórców, ale w porozumieniu z *Product ownerem*. Cel sprintu określa co ma zostać osiągnięte jako wynik sprintu.

Artefakty Scrum — kontynuacja

Przyrost (ang. *Increment*) jest konkretnym wynikiem, który przybliży projekt do *celu projektu*. Musi on dać się zintegrować z innymi przyrostami, być użytecznym i zostać starannie przetestowany. W trakcie jednego sprintu może powstać i zostać dostarczonych interesariuszom wiele przyrostów. Przyrost musi spełniać *definicję ukończenia* (ang. *Definition of Done* — *DoD*), która formalnie określa miary jakości dla produktu. Jeśli nie spełnia on definicji ukończenia, to jest zwracany do *Product backlog*. Definicję ukończenia tworzy zespół Scrum.

Zalety Scrum:

- + pozwala zespołowi szybko reagować na zmiany w wymaganiach,
- + czyni postęp lepiej widoczny niż w tradycyjnych metodach zarządzania,
- + wprowadza do procesu punkt wiedzy interesariuszy.

Wady Scrum:

- jest dosyć abstrakcyjna i może być trudna do wprowadzenia,
- ma „obciążenia kulturowe”.

Planowanie

W Scrum i innych metodach zwinnych w zasadzie nie ma planowania długoterminowego, ale postęp prac jest śledzony i dla każdej iteracji (sprintu) szacowana jest ilość pracy, która może być wykonana.

W tradycyjnych metodach zarządzania projektem *kierownik projektu* (ang. *project manager*) jest odpowiedzialny za przygotowanie harmonogramu i jego aktualizację wraz z postępem prac.

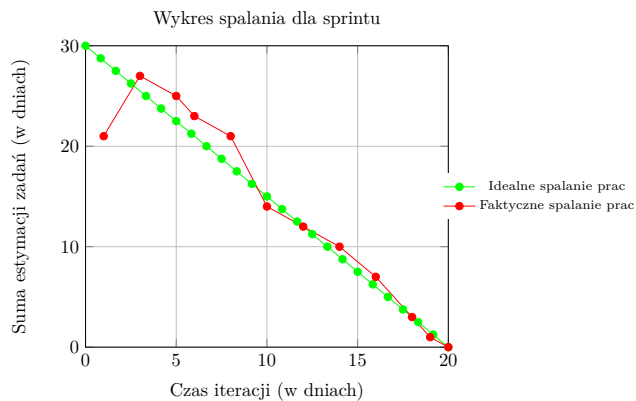
Planning poker

Planning poker jest ciekawą metodą szacowania ilości pracy, która musi być wykonana dla każdego zdania, które jest przenoszone z *Product backlog* do *Sprint backlog*. Ta metoda ma formę gry karcianej, w której każda z kart ma przypisaną liczbę ze *zmodyfikowanego ciągu Fibonacciego*: 0, 1/2, 1, 2, 3, 5, 8, 13, 20, 40, 100. Te liczby nazywają się *Story points* i opisują wysiłek potrzebny do ukończenia zadania. Talia może również zawierać karty z wizerunkiem kubka kawy (gracz musi zrobić sobie przerwę), symbolem ∞ symbol (zadanie jest zbyt duże, aby go oszacować) i symbolem ? (gracz nie wie, jak oszacować zadanie). Każdy uczestnik gry ma swoją talię kart. *Moderator* wybiera zadanie z *Product backlog*, przedstawia je, a pozostali gracze dyskutują nad jego kosztem. Po dyskusji każdy gracz wybiera jedną kartę, nie pokazując jej pozostałym graczom. Na znak dany przez *moderatora* wszyscy pokazują swoje karty. Jeśli są one jednakowe, to takie jest przyjmowane oszacowanie zadania. Jeśli są różnice, to gracze ze skrajnymi wartościami kart (najniższa i/lub najwyższa estymacja) uzasadniają swoje oceny i proces głosowania jest powtarzany.

Wykres spalania (ang. *burn down chart*)

Do śledzenia postępu w iteracji (sprintcie) lub całym projekcie można użyć *wykresu spalania* (ang. *burn down chart*) i *wykresu wypalania* (ang. *burn up chart*). Przykład wykresu spalania przedstawiono na Rysunku 1. Zielony wykres to *prosta idealnego spalania* ukazująca ilość pracy, która powinna zostać wykonana każdego dnia sprintu, aby ukończyć wszystkie zadania. Czerwony wykres obrazuje faktycznie wykonaną pracę. Z jego kształtu wynika, że przez większość czasu zespół miał problem z utrzymaniem tempa prac zgodnego z oszacowaniami, choć w niektórych okresach wyprzedzał estymacje. Ostatecznie jednak, wszystkie zaplanowane zadania zostały ukończone. Wykres wypalania jest zwierciadlanym odbiciem, wzdłuż prawej osi, wykresu spalania i obrazuje pracę, która już została ukończona.

Wykres spalania (ang. *burn down chart*)



Rysunek: Przykład wykresu spalania

17 / 29

Notatki

Planowanie w tradycyjnym zarządzaniu

W tradycyjnych modelach zarządzania (np. kaskadowym), kierownik zespołu dzieli całą pracę do wykonania na zadania (ang. *tasks*) i etapy (ang. *milestones*). Każdy etap jest zakończeniem jednego lub kilku zadań i wiąże się z uzyskaniem określonego wyniku. Czas niezbędny do ukończenia zadania musi zostać oszacowany. Nie powinien on być krótszy niż jeden tydzień roboczy, ani dłuższy niż osiem takich tygodni. W pierwszym przypadku w projekcie pojawiło by się zbyt wiele zadań, a w drugim postęp prac byłby trudny do monitorowania. Początkowy harmonogram jest tylko oszacowaniem i musi być aktualizowany wraz z postępem prac.

18 / 29

Notatki

Zadania i etapy

Zadanie	Czas (dni)	Zależność
T1	8	
T2	15	
T3	15	T1 (M1)
T4	10	
T5	10	T2, T4 (M2)
T6	5	T1, T2 (M3)
T7	20	T1 (M1)
T8	25	T4 (M5)
T9	15	T3, T6 (M4)
T10	15	T5, T7 (M7)
T11	7	T9 (M6)
T12	10	T11 (M8)

19 / 29

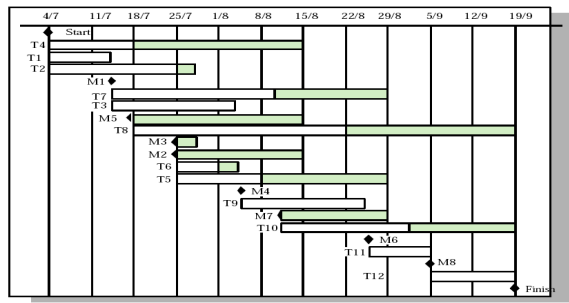
Notatki

Wykres Gantta

Wykres Gantta pozwala przedstawić harmonogram projektu w sposób graficzny. Na następnym slajdzie przedstawiono taki wykres, który odwzorowuje dane w tabeli na poprzednim slajdzie. Romby reprezentują etapy, a paski zadania. Proszę zwrócić uwagę, że niektóre z zadań i etapów mają zielone „ogony”. W ten sposób oznaczono *rezerwy czasowe* (ang. *time spare*), nazywane także *zapasami czasu* (ang. *time buffer*). Zakończenie takiego zadania lub osiągnięcie takiego etapu może być opóźnione o taką ilość czasu, bez konsekwencji dla terminu zakończenia całego projektu. Zadania i etapy bez takiego „ogona” muszą być ukończone lub osiągnięte w wyznaczonym czasie, nie mogą zostać opóźnione.

20 / 29

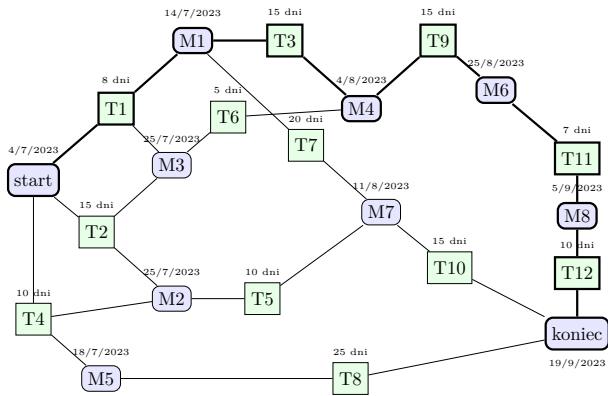
Notatki



Sieci PERT

Sieci PERT zostały opracowane na potrzeby projektu konstrukcji raket Polaris dla marynarki Stanów Zjednoczonych (ang. *The US Navy Polaris Rocket Program*). Oryginalnie skrót PERT oznaczał *Program Evaluation and Review Technique*, ale później jego rozwinięcie zostało zmienione na *Project Evaluation and Review Technique*. Sieć PERT przedstawia harmonogram projektu w postaci grafu skierowanego, w którym wierzchołki reprezentują zadania i etapy. W ten sposób łatwiej jest zespołowi projektowemu dostrzec zależności między zadaniami. To rozwiązania pozwala im również znaleźć *ścieżkę krytyczną* (ang. *critical path*), czyli sekwencję zadań i etapów, które określają całkowity czas realizacji projektu i w związku z tym nie mogą być opóźnione.

Sieci PERT



Zarządzanie ryzykiem

Ryzyko to iloczyn prawdopodobieństwa wystąpienia niesprzyjających okoliczności (zagrożeń) i ich szacowanych konsekwencji. W projektach informatycznych jest wiele źródeł takich zagrożeń, jak frustracja członków zespołu, rotacja pracowników, złe zarządzanie, niewłaściwe narzędzia i niedostateczne oszacowania. Zagrożenia mogą mieć negatywny wpływ na projekt, produkt, a nawet całe przedsiębiorstwo. W metodach zwinnych, takich jak Scrum, informacje zwrotne są dostarczane bardzo często, więc zmiany poziomu ryzyka lub pojawienie się zagrożenia są niemal natychmiast widoczne i zespół może na nie szybko reagować. W tradycyjnych metodach zarządzania, kierownik projektu musi rozpoznać zagrożenia, oszacować ryzyka i przygotować plany przeciwdziałania tym zagrożeniom.

Ocena ryzyka

Notatki

Początkowo kierownik projektu musi zidentyfikować wszystkie przypuszczalne zagrożenia dla projektu, oszacować prawdopodobieństwo ich materializacji oraz wpływ jaki będą one miały na projekt. Prawdopodobieństwo może być wyrażone w sposób opisowy, np. „bardzo małe”, „małe”, „średnie”, „duże”, „bardzo duże”. Natomiast wpływ może być określony jako „pomijalny”, „znośny”, „poważny” lub „katastroficzny”. Wartości tych dwóch zmiennych pozwalają kierownikowi projektu wyznaczyć najbardziej niebezpieczne zagrożenia i przygotować plany przeciwdziałania nim.

25 / 29

Plany przeciwdziałania

Notatki

Kierownik projektu może przygotować trzy rodzaje planów przeciwdziałania zagrożeniom:

Strategie unikania (ang. *avoidance strategies*), których celem jest zmniejszenie prawdopodobieństwa wystąpienia ryzyka;

Strategie minimalizacji (ang. *minimization strategies*), które mają na celu zmniejszenie wpływu jaki może mieć zagrożenie;

Plany awaryjne (ang. *emergency plans*), to plany przygotowywane na okoliczność pojawienia się najpoważniejszych zagrożeń.

26 / 29

Monitorowanie zagrożeń

Notatki

Wraz z postępem projektu niektóre zagrożenia przestają być możliwe, ale prawdopodobieństwo wystąpienia innych wzrasta. Z tego powodu kierownik projektu musi ciągle monitorować źródła potencjalnych zagrożeń i dostosowywać plany przeciwdziałania do bieżącej sytuacji.

27 / 29

Pytania

Notatki

?

28 / 29

Dziękuję Państwu za uwagę!

Notatki

Notatki

Notatki

Notatki
