

*Ewa Marchwicka*

Politechnika Wroclawska, Wydział Zarządzania  
ORCID: 0000-0001-6097-784X

*Tymon Marchwicki*

Politechnika Wroclawska, Wydział Zarządzania  
ORCID: 0009-0007-9105-8249

## Adaptacyjne planowanie sprintu na podstawie informacji o ryzyku

---

### Streszczenie

Zarządzanie ryzykiem w (standardowym) modelu *scrum* realizowane jest tylko w sposób niejawny. Z przeglądu literatury przeprowadzonego przez autorów wynika, że brakuje podejść ilościowych do jawnego zarządzania ryzykiem w środowisku *scrum*. Prezentowane rozważania mają służyć ustaleniu, czy możliwe jest zarządzanie ryzykiem zgodnie z metodyką *scrum* w sposób ilościowy, na podstawie jawnej informacji o ryzyku podawanej w sposób lingwistyczny, a także przeanalizowanie zalet takiego podejścia. W artykule opracowana została także koncepcja umożliwiająca kwantyfikację określonych rodzajów ryzyka w sprincie. Prezentowane podejście ułatwia adaptację planu sprintu, w którym na wyjściu zwracany jest zbiór możliwych akcji naprawczych. Akcje te są skwantyfikowane pod kątem kryterium wartości biznesowej dla klienta, które określa się mianem wartości sprintu (ang. *sprint value*). Proponowane podejście zostało zaprezentowane wraz z przykładem ilustracyjnym. Z przeprowadzonych badań, wypełniających wspomnianą lukę badawczą, wynika, że w metodyce *scrum* można zastosować ilościowe zarządzanie ryzykiem, na podstawie jawnej informacji o ryzyku podawanej na wejściu w formie wyrażen lingwistycznych.

**Słowa kluczowe:** metodyka *scrum*, niepewność, projekty informatyczne, systemy rekomendacyjne, zarządzanie ryzykiem

**Kody klasyfikacji JEL:** M100, M150

---

## 1. Wprowadzenie

*Scrum* to jedna z najpopularniejszych metodyk zwinnych. Jawne zarządzanie ryzykiem nie jest wbudowane w *scrum*, ale ryzykiem można zarządzać w *scrum* na kilka (niejawnych) sposobów [Dingsøyr, 2021]. Autorzy tej metodyki [Schwaber, Sutherland, 2020] zalecają stosowanie krótszych sprintów w celu zmniejszenia wpływu różnych form ryzyka na sprint. Krótkie cykle rozwojowe pozwalają redukować ryzyko zewnętrzne i operacyjne [Georgiev, 2014]. Ponadto częste informacje zwrotne od interesariuszy, specyficzne dla *scrum* i *agile*, służą zmniejszeniu niepewności związanej z wymaganiami [Dönmez, 2018]. Wszystkie sprinty są wyposażone w działające oprogramowanie [Beck i in., 2023], dzięki czemu interesariusze mogą ocenić produkt na koniec każdego z nich i dostosować swoje preferencje. Zespół może ograniczyć ryzyko nieukończenia sprintu na czas poprzez prześledzenie historii zadań zrealizowanych w poprzednich sprintach i oszacowanie na ich podstawie zakresu możliwego do wypracowania w przyszłych sprintach, a także poprzez udoskonalanie elementów rejestru produktu (ang. *product backlog*), przyczyniające się do lepszego zrozumienia wymagań [Schwaber, Sutherland, 2020; Dönmez, 2018; Dingsøyr, 2021]. Podejście *agile/scrum* nie polega na eliminowaniu ryzyka i niepewności, ale na „kontrolowaniu i dostosowywaniu się do nieoczekiwanych zdarzeń” [Dönmez, 2018, s. 95]. Metodyka *scrum* pozwala szybko reagować na pojawiające się przejawy ryzyka poprzez organizowanie codziennych spotkań, podczas których członkowie zespołu zgłaszają potencjalne problemy i zagrożenia. Na podstawie tych informacji członkowie zespołu mogą zaktualizować plan sprintu [Schwaber, Sutherland, 2020], wspólnie decydując, jakie zmiany będą najlepsze w odpowiedzi na ryzyko, przy czym wszelkie modyfikacje nie są w tym przypadku kwantyfikowane. Dodatkową praktyką związaną z ryzykiem, często stosowaną w metodyce *scrum* (choć niebędącą jej częścią), jest uwzględnianie ryzyka w estymacjach poprzez wykorzystanie wybranych technik estymacji [Prater, 2019; Ghazali, 2018]. Podsumowując, standardowe podejście do ryzyka w *scrum* jest proste i praktyczne. Pozwala złagodzić negatywne skutki wystąpienia ryzyka. Niestety takie podejście wydaje się niewystarczające, aby zapewnić sukces projektu. Z raportów Standish Group International [2015] wynika, że także projekty *scrum* często kończą się niepowodzeniem. Kiedy w sprincie zmaterializuje się wiele przypadków ryzyka, jego początkowy plan może szybko stać się nieaktualny, a zespół programistów może mieć problemy z terminowym ukończeniem sprintu. Dodatkowo kontekst *agile* może nawet nasilić negatywne skutki niektórych rodzajów ryzyka, takich jak rotacja personelu, ponieważ „dokumentacja jest ograniczona, a komunikacja mniej formalna” [Dingsøyr, 2021, s. 77].

Celem tego artykułu jest przeanalizowanie, czy możliwe jest ilościowe zarządzanie ryzykiem w *scrum*, na podstawie jawnej informacji o ryzyku podawanej w sposób lingwistyczny, i zaproponowanie bazującego na tej koncepcji podejścia. Zostanie ono następnie zilustrowane przykładem, a także przeanalizowane pod kątem zalet. Podejście to powinno pozwolić zoptymalizować plan sprintu z uwzględnieniem jawnych informacji o ryzyku. Dane wejściowe mają charakter lingwistyczny, co odpowiada sposobowi, w jaki ludzie oceniają zwykle ryzyko oraz podejściu stosowanemu przez PMI [2021]. W opisywanym modelu akcje naprawcze są odpowiedzią na te przypadki ryzyka, które już wystąpiły lub są przewidywane w sprintcie. Możliwe akcje naprawcze są kwantyfikowane, a akcje korygujące plan sprintu rekomendowane zgodnie z kryteriami dotyczącymi jakości sprintu. Ze względu na to, że ryzyko ma charakter niedeterministyczny, rekomendacje wyjściowe zawierają informacje o prawdopodobieństwie. W ramach przyszłych badań planowane jest przeanalizowanie liczby opóźnień w sprintach w wyniku zastosowania zaprezentowanego podejścia.

## 2. Przegląd literatury

### 2.1. Definicja ryzyka i jego czynniki

W literaturze można znaleźć wiele definicji ryzyka. Autorzy postrzegają je najczęściej jako niepewne zdarzenie [m.in. PMI, 2021; Dönmez, 2018; Amaral, 2019; Tsiara, 2016; Menezes, 2018; Islam, 2009; Rahman, 2021; Yim, 2015]. Niektórzy definiują ryzyko jako prawdopodobieństwo [m.in. Nicolas, 2018; Morrison, 2018; Nogueira, 2014; Jabangwe, 2010; Neves, 2016; Higuera, 2016; Gemmer, 1997; Trzeciak, 2021] lub ewentualność (ang. *possibility*) [np. Boehm, 1991], a rzadziej także jako: przyszły problem [Suda, 2010], zmienną projektu [Boban, 2014], negatywny skutek [Kosztyn, 2020] czy niepewność [Han, 2015; Kuo, 2019]. Prawdopodobieństwo jest również brane pod uwagę wyłącznie jako jeden z atrybutów ryzyka [np. Dönmez, 2018; Boban, 2004], często bywa też wymieniane wraz z jego konsekwencjami [m.in. Morrison, 2018; Jabangwe, 2010; Islam, 2009] i określane jako dotkliwość (ang. *severity*) [Amaral, 2019; Higuera, 2016], istotność/ważność (ang. *importance*) [Neves, 2016] lub oddziaływanie (ang. *impact*) [Gemmer, 1997]. W wielu artykułach ryzyko definiowane jest jako zagrożenie, gdy uwaga badaczy skupia się na jego negatywnych skutkach [m.in. Nicolas, 2018; Amaral, 2019; Nogueira, 2014; Tsiara, 2016], w innych przypadku podkreśla się natomiast też szanse, jakie może dawać jego wystąpienie [m.in. PMI, 2021; de Menezes, 2019; Dingsøyr, 2021; Trzeciak, 2021]. Niektórzy autorzy rozróżniają ryzyko i niepewność [np. Dönmez, 2018; Rahman, 2021; Yim, 2015; Taipalus, 2020], wskazując, że prawdopodobieństwo wystąpienia ryzyka można oszacować, lecz nie jest to możliwe w przypadku niepewności. Czasami wspomina się, że ryzyko jest podzbiorem niepewności, które występują w formie zarówno zagrożeń, jak i szans, podczas gdy ryzyko wiąże się wyłącznie z zagrożeniami [Taipalus, 2020]. Inni autorzy używają terminów „ryzyko” i „niepewność” zamiennie [m.in. Han, 2015; Kuo, 2019; Hazir, 2020].

W tabeli 1 zestawiono ze sobą popularne w literaturze definicje ryzyka, aby pokazać, do jak wielu znaczeń odnosi się to pojęcie.

W niniejszej publikacji wykorzystano definicję ryzyka cytowaną za PMBOK, ale nieznacznie zmodyfikowaną. Ryzyko rozumiane jest tu bowiem jako „niepewne zdarzenie lub warunek, który, jeśli wystąpi, to będzie miał negatywny wpływ na jeden lub więcej celów projektu” [PMI, 2021]. Przyjęta na potrzeby tej pracy definicja została uproszczona tak, aby objąć nią jedynie negatywne skutki tego zjawiska, pierwotna definicja PMBOK odnosi się natomiast także do skutków pozytywnych.

Tabela 1. Wybrane definicje ryzyka

Definicja (autor)	Główne pojęcie	Zagrożenie/Szansa	Atrybuty
PMI [2021]	zdarzenie	obydwa	–
Dönmez [2018]	zdarzenie	–	prawdopodobieństwo
Nicolas [2018]	prawdopodobieństwo	zagrożenie	–
Morrison [2018]	prawdopodobieństwo	–	prawdopodobieństwo, konsekwencje
Amaral [2019]	zdarzenie	zagrożenie	prawdopodobieństwo, dotkliwość
Nogueira [2014]	prawdopodobieństwo	zagrożenie	–
Tsiara [2016]	zdarzenie	zagrożenie	–
Menezes [2018]	zdarzenie	obydwa	–
Suda [2010]	przyszły problem	zagrożenie	–
Boban [2004]	zmienna projektu	zagrożenie	prawdopodobieństwo
Jabangwe [2010]	prawdopodobieństwo	zagrożenie	prawdopodobieństwo, konsekwencje
Dingsøyr [2021]	zdarzenie	obydwa	–
Neves [2016]	prawdopodobieństwo	zagrożenie	prawdopodobieństwo, istotność/ważność
Higuera [1996]	prawdopodobieństwo	zagrożenie	prawdopodobieństwo, dotkliwość
Islam [2011]	zdarzenie	zagrożenie	prawdopodobieństwo, konsekwencje
Han [2015]	niepewność	zagrożenie	prawdopodobieństwo, oddziaływanie
Gemmer [1997]	prawdopodobieństwo	zagrożenie	prawdopodobieństwo, oddziaływanie
Boehm [1991]	ewentualność	zagrożenie	ekspozycja (prawdopodobieństwo, strata)
Kuo [2019]	niepewność	zagrożenie	ewentualność/przypadkowość (prawdopodobieństwo, strata)
Kosztyn [2020]	efekt	zagrożenie	prawdopodobieństwo
Trzeciak [2021]	prawdopodobieństwo	obydwa	–
Yim [2015]	zdarzenie	zagrożenie	–
Rahman [2021]	zdarzenie	zagrożenie	–
Ibadov [2018]	zdarzenie	zagrożenie	–

Źródło: opracowanie własne.

W literaturze obok ryzyka pojawia się często pojęcie pokrewne, jakim jest czynnik ryzyka (ang. *risk factor*). Niektórzy autorzy nazywają go zamiennie również wskaźnikiem ryzyka [m.in. Hsieh, 2018; Yim, 2015]. Czynnik ryzyka definiowany jest jako niepewny warunek,

który wpływa negatywnie na koszt projektu, czas jego trwania i jakość [Hijazi, 2014] lub zagraża ukończeniu projektu [Neves, 2016]. Czasami określa się go także jako cechę [Islam, 2009] lub czynnik wewnętrzny/zewnętrzny [Menezes, 2018], który przyczynia się do wystąpienia ryzyka, albo jako ryzyko zidentyfikowane [Hartono, 2018]. Ogólnie rzecz biorąc, ryzyko jest spowodowane sumą składających się na nie czynników [Menezes, 2018]. Często łatwiej jest oszacować prawdopodobieństwo wystąpienia ryzyka (np. przekroczenia zaplanowanego czasu), wymieniając wszystkie jego czynniki.

## 2.2. Katalogi czynników ryzyka

Istnieje wiele standardowych czynników ryzyka związanych z projektowaniem oprogramowania. Ich usystematyzowane listy nazywa się zwykle katalogami czynników ryzyka. Wykaz wybranych pozycji literatury wraz z katalogami czynników ryzyka związanych z oprogramowaniem oraz informacją o liczbie pozycji w katalogu i jego typie zamieszczono w tabeli 2.

Tabela 2. Wybrane katalogi czynników ryzyka związanych z oprogramowaniem

Katalog ryzyka (autor)	Liczba pozycji	Typ katalogu
Abioye [2020]	77	SDLC ( <i>software development lifecycle</i> )
Alfozan [2020]	59	SD ( <i>software development</i> )
Beecham [2021]	63	GSD ( <i>global software development</i> )
Georgiev [2014]	35	SD ( <i>software development</i> )
Hijazi [2014]	99	SDLC ( <i>software development lifecycle</i> )
Nicolas [2018]	30	GSD ( <i>global software development</i> )
Dönmez [2018]	12	ASD ( <i>agile software development</i> )
Hsieh [2018]	27	NSD ( <i>new software development</i> )
Menezes [2018]	34	<i>product &amp; development &amp; program</i>
Sundararajan [2019]	31	GSD ( <i>global software development</i> )
Al-Shehab [2021]	37	SD ( <i>software development</i> )
Iftikhar [2021]	26	GSD ( <i>global software development</i> )
Asif [2020]	26	SD ( <i>software development</i> )
Alves [2021]	20	SD ( <i>software development</i> )
Suda [2010]	10	SD ( <i>software development</i> )
Boehm [1991]	10	SD ( <i>software development</i> )
Takagi [2005]	22	SD ( <i>software development</i> )
Han [2015]	22	SD ( <i>software development</i> )
Klein [2000]	18	SD ( <i>software development</i> )
Yim [2015]	36	<i>project (electrical, mechanical, software)</i>
Wallace [2004]	53	SD ( <i>software development</i> )
Marchwicka [2020]	72	GSD ( <i>global software development</i> )

Źródło: opracowanie własne.

Mając do dyspozycji usystematyzowane katalogi czynników ryzyka, można opracować zautomatyzowane narzędzia i metody ilościowe zarządzania ryzykiem. Są one często nazywane systemami rekomendacyjnymi.

### 2.3. Systemy rekomendacyjne uwzględniające informacje o ryzyku

Mimo że w literaturze dotyczącej zarządzania projektami widnieje wiele artykułów na temat jawnego zarządzania ryzykiem oprogramowania, istnieje luka badawcza w przypadku gotowych do zastosowania narzędzi i metod ilościowych wspomagających decyzje podejmowane przez kierowników projektu z uwzględnieniem informacji o ryzyku. Podobne wnioski prezentują Asif [2020] i Aslam [2017], którzy dodatkowo stwierdzają, że dostępne modele są zbyt złożone, aby można je było zastosować w praktyce. Asif [2020] przedstawił model oparty na regułach, który generuje łagodzące działania naprawcze w odniesieniu do wybranych czynników ryzyka. Model został przetestowany na 26 czynnikach ryzyka i 57 działaniach łagodzących ze skutecznością na poziomie 51–55%. Aslam [2017] zaproponował podobny system wspomagania decyzji, który opiera się na pytaniach, opcjach i regułach celem wyboru odpowiedniej strategii zarządzania ryzykiem w kontekście rozwoju oprogramowania rozproszonego. System jest testowany na wielu studiach przypadków. Vizzini [2017] zaprezentował technikę zarządzania ryzykiem projektowym, opartą na analizie profilu ryzyka i działaniach korygujących. Jej algorytm realizowany jest w stałych punktach kontrolnych, zwanych WPS (status postępu prac). Pewnym ograniczeniem tego modelu jest jego deterministyczny charakter, który nie odzwierciedla rzeczywistej natury ryzyka. Niedeterministyczny system wspomagania decyzji wprowadził Fang [2012], przedstawiając model, który opiera się na symulacji. Rozwiązanie to wspiera kierownika projektu w podejmowaniu działań w odpowiedzi na ryzyko i uwzględnia interakcje między jego poszczególnymi przejawami. Inny system rekomendacji wspomagających podejmowanie decyzji opisał Filippetto [2021]. Polega on na wykorzystaniu historycznych danych projektowych w celu uzyskania informacji o zmianach kontekstu projektu (np. wielkości zespołu czy wymagań), znajdowaniu podobieństw między bieżącym kontekstem a kontekstem historycznym oraz przewidywaniu określonych przejawów ryzyka, które mogą wystąpić w analogicznej sytuacji. Autor nie podał jednak żadnych szczegółów dotyczących implementacji tej metody, przedstawił jedynie sam model.

Wszystkie opisane wyżej modele i systemy wspomagania decyzji mają ogólne przeznaczenie i nie uwzględniają specyficznych cech metodyki *scrum*. Choć ostatnia z wymienionych publikacji odnosi się do zarządzania projektami zwinnymi, przedstawiony w niej model nie jest przeznaczony wprost dla podejścia *agile* – został on zaprezentowany jedynie na przykładzie użycia w projekcie zwinnym.

## 2.4. Scrum uwzględniający informacje o ryzyku

Niektórzy autorzy twierdzą, że opisane we wprowadzeniu niejawnie zarządzanie ryzykiem, wbudowane w podejście *agile*, jest niewystarczające [np. Dingsøyr, 2021; Kuchta, 2013]. Z tego powodu w literaturze można odnaleźć pozycje włączające jawne zarządzanie ryzykiem w obszar zainteresowań związany z metodyką *scrum*. Chaouch [2019] zaproponował podejście do zarządzania ryzykiem oparte na PMBOK, które stanowi rozszerzenie *scrum*. Mousaei [2018] i Esteki [2020] opisali autorskie propozycje, integrujące elementy zarządzania ryzykiem z metodyki PRINCE2 i metodyki *scrum*. Bardzo prosty schemat zarządzania ryzykiem oparty na dodatkowych spotkaniach *scrum* przedstawił także Hammad [2018]. Z kolei Tavares, da Silva i de Souza [2017a, 2017b] przeanalizowali praktyki zarządzania ryzykiem w projektach *scrum* na podstawie ankiet przeprowadzonych w dziesięciu różnych projektach.

## 3. Materiały i metody

W tej sekcji opisano koncepcję autorskiego podejścia do planowania sprintu, uwzględniającego informacje o ryzyku. Przegląd literatury, wywiady z kierownikami projektów i obserwacje uczestniczące (autorzy byli członkami kilku zespołów *scrum*) zostały wykorzystane jako metody badawcze w celu rozszerzenia standardowego planowania *scrum* o jawne zarządzanie ryzykiem za pomocą danych wejściowych wyrażonych w sposób lingwistyczny i danych wyjściowych w postaci rekomendacji (tj. na wyjściu otrzymywany jest zbiór możliwych akcji, spośród których można wybrać dowolny wariant). Na podstawie przeglądu literatury zidentyfikowano lukę badawczą, a następnie przeanalizowano wywiady i obserwacje uczestniczące, aby wybrać dane wejściowe i wyjściowe oraz kryterium optymalizacyjne, które są potrzebne do jawnego zarządzania ryzykiem w *scrum*.

Zgodnie z przyjętą wcześniej definicją ryzyko to „niepewne zdarzenie lub warunek, który jeśli wystąpi, to będzie miał negatywny wpływ na jeden lub więcej celów projektu” [PMI, 2021]. W kontekście sprintów zamiast „cele projektu” używane jest określenie „cele sprintu”. Na potrzeby niniejszej publikacji uwzględniono następujące cele sprintu: czas, koszt, zakres i jakość. Jest to lista standardowa, oparta na żelaznym trójkącie zarządzania projektami [PMI, 2021]. Podstawą ilościowego, adaptacyjnego podejścia do zarządzania ryzykiem w metodyce *scrum*, zaproponowanego w artykule, jest kryterium optymalizacyjne, stanowiące w tym przypadku swoistą nowość. Kryterium nazwane zostało *sprint value* (SV) i zdefiniowane jako prognozowana wartość zadań dostarczanych klientowi na koniec sprintu, pomniejszona o prognozowany koszt adaptacji do takiego stanu (równanie 1).

$$SV = P - A = \omega_p Capacity - (\omega_a^1 A_1 + \omega_a^2 A_2 + \omega_a^3 A_3 + \omega_a^4 A_4 + \omega_a^5 A_5 + \omega_a^6 A_6) \quad (1)$$

gdzie:

$SV$  – *sprint value*,

$P$  – prognozowana wartość zadań ukończonych w sprincie,

$A$  – prognozowany koszt adaptacji do takiego stanu,

*Capacity* – maksymalna wartość zadań w sprincie (uwzględniająca prognozowane ryzyko),

$A_1$  – koszt adaptacji polegającej na nieukończeniu zadań w trakcie sprintu,

$A_2$  – koszt adaptacji polegającej na spadku jakości zadań,

$A_3$  – koszt adaptacji polegającej na zmienionym zakresie zadań,

$A_4$  – koszt adaptacji polegającej na zredukowanym zakresie zadań,

$A_5$  – koszt adaptacji polegającej na rozszerzonym zakresie zadań,

$A_6$  – koszt adaptacji polegającej na poniesieniu dodatkowych kosztów zadań,

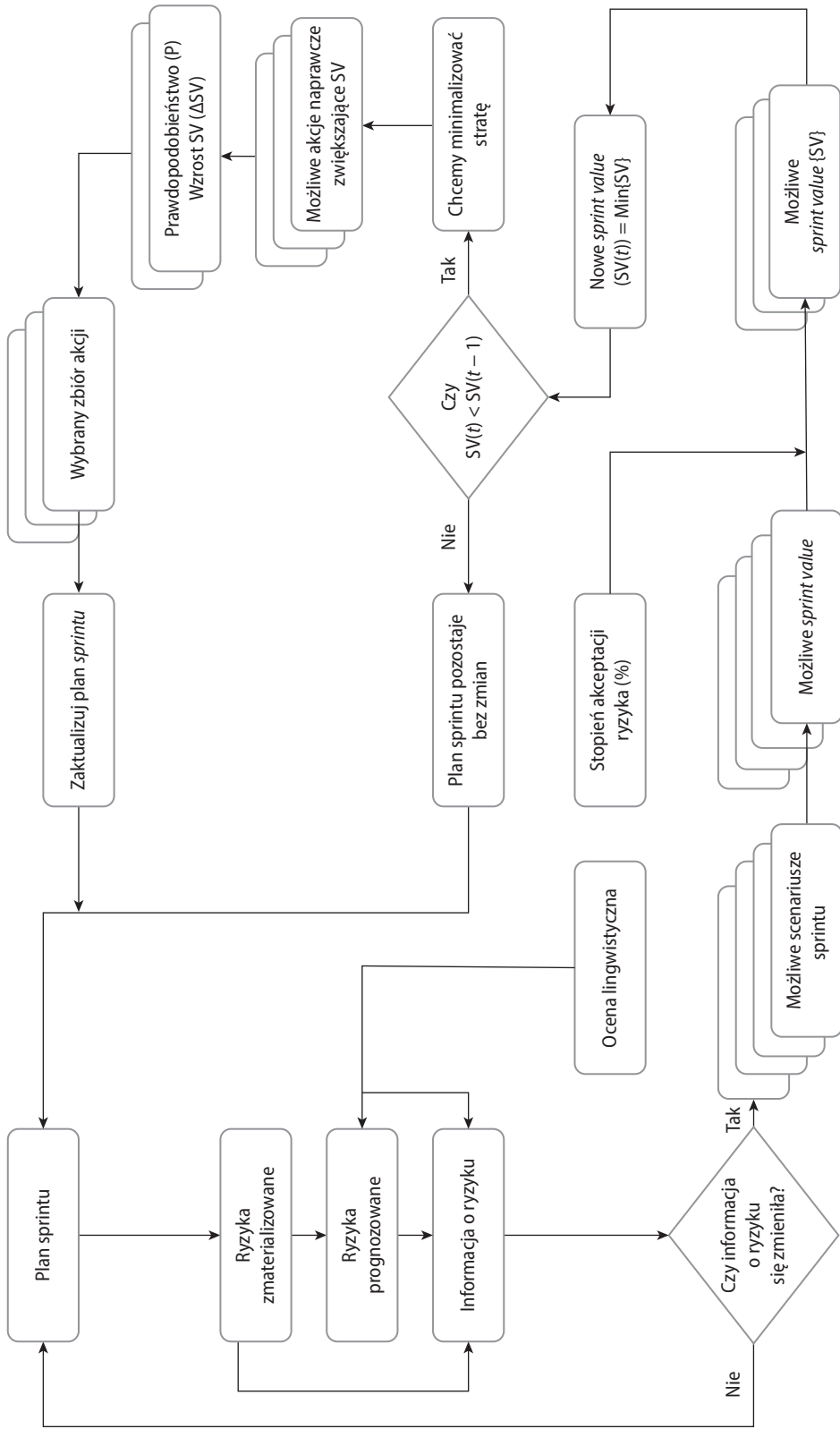
$\omega_p$  – waga przypisana prognozowanej wartości,

$\omega_a^1 \dots \omega_a^6$  – wagi przypisane prognozowanym adaptacjom.

*Sprint value* łączy pojęcie *sprint capacity* z prognozowanym ryzykiem oraz adaptacją planu (czyli zwinnością). *Sprint capacity* to maksymalna wartość zadań, które mogą być ukończone w sprincie, wyrażana w jednostce *story point*, proporcjonalna do dostępności zespołu. W tym kontekście *sprint capacity* obejmuje również prognozowane przejawy ryzyka i jest niedeterministyczne. Prognozowany koszt adaptacji określa, ile będą kosztować wszystkie adaptacje na zakończenie sprintu. Może składać się on z kilku komponentów (w zależności od rodzaju potrzebnej adaptacji). Wszystkie elementy wzoru są wyrażane w jednostce *story point*. Dzięki temu, że w kryterium użyto wag, jego cechą jest elastyczność. W zależności od potrzeb projektowych zespół może decydować, które adaptacje są preferowane (przypisane im wagi będą niższe, co przełoży się na niższy koszt adaptacji), a których należy unikać (uzyskają wyższe wagi). Przed rozpoczęciem sprintu kryterium *sprint value* jest równe *sprint capacity*. Ze względu na to, że w zaproponowanej koncepcji ryzyko traktowane jest jako zagrożenie, *sprint value* wyznaczone na początku sprintu będzie tym mniejsze, im większy udział ryzyka przewidziano. Koszty adaptacji pojawią się, gdy pierwotny plan sprintu będzie musiał ulec zmianie (np. skutek wystąpienia ryzyka). Ideą stosowania akcji naprawczych jest takie dostosowanie czasu, zakresu lub jakości, aby nowy plan uzyskał jak najwyższą wartość *sprint value*. Zwykle bardzo ważnym kryterium jest czas. Oznacza to, że największą wagę powinien mieć komponent  $A_1$ . Gdy wszystkie planowane zadania zostaną zakończone zgodnie z planem na koniec sprintu, da to największą satysfakcję po stronie klienta. Przykładowo, gdy zagrożone jest terminowe zakończenie wszystkich zadań sprintu, gdyż w trakcie jego realizacji wystąpiło nieprzewidziane ryzyko, nadal istnieje możliwość ograniczenia zakresu innych zadań lub poniesienia dodatkowych kosztów za nadgodziny, aby zyskać więcej czasu. Istnieje prawdopodobieństwo, że dana akcja naprawcza zwiększy wartość *sprint value*, symulując np. możliwe przebiegi sprintu (rysunek 1).



Rysunek 1. Symulacje możliwych przebiegów sprintu na podstawie informacji o ryzyku wraz z ich oceną (*sprint value*)



Źródło: opracowanie własne.

Na rysunku 1 przedstawiona jest sytuacja, w której mamy zaplanowany sprint zgodnie z informacją o ryzyku (prognozami podanymi w formie lingwistycznej). Jeżeli informacja o ryzyku uległa zmianie w stosunku do ostatniego planu (np. zmieniły się prognozy lub zmaterializowały te przejawy ryzyka, które nie były prognozowane), to analizowane są możliwe scenariusze przebiegu sprintu. Każdy z nich oceniany jest na podstawie *sprint value*. Poszczególne scenariusze mogą doprowadzić do uzyskania różnych wartości tego kryterium. Nie interesują nas jednak wszystkie możliwe warianty. Pewien procent scenariuszy o najmniejszej wartości *sprint value* odrzucamy (jest to tzw. stopień akceptacji ryzyka). Nowe *sprint value* to minimalne *sprint value* spośród wszystkich nieodrzuconych scenariuszy. Wymaga ono sprawdzenia, czy jest mniejsze od poprzedniego *sprint value*. Jeśli zachodzi taka zależność, to prognozowana wartość, którą jesteśmy w stanie dostarczyć klientowi (uwzględniając koszt adaptacji), jest mniejsza od aktualnie planowanej. W tym przypadku należy minimalizować stratę. Wyjściem rekomendowanym jest lista możliwych akcji naprawczych wraz z ich wartościami *sprint value* i prawdopodobieństwem uzyskania tych wartości. Użytkownik może wybierać między poszczególnymi wariantami. Analizowane są różne akcje naprawcze (adaptacje) zwiększające *sprint value* (z uwzględnieniem prawdopodobieństwa takiej poprawy), tak by wybrać spośród nich najbardziej optymalny wariant. Zastosowanie wybranych akcji naprawczych zmienia plan sprintu, co obrazuje przykład przedstawiony w tabeli 3.

Tabela 3. Przykładowe akcje naprawcze sprintu

Lp.	Akcja	Opis
1	redukcja zakresu	zadanie $i$ zostaje zredukowane o $q$ funkcji
2	dekompozycja	zadanie $i$ zostaje zdekomponowane na $n$ mniejszych
3	usunięcie	zadanie $i$ zostaje usunięte
4	nadgodziny	planuje się przeznaczyć $h$ dodatkowych nadgodzin, aby wydłużyć czas sprintu
5	dodanie zadania	nowe zadanie $i$ dodawane jest do sprintu
6	redukcja jakości	zadanie $i$ zostaje zredukowane o $q$ przypadków testowych, które nie zostaną sprawdzone

Źródło: opracowanie własne.

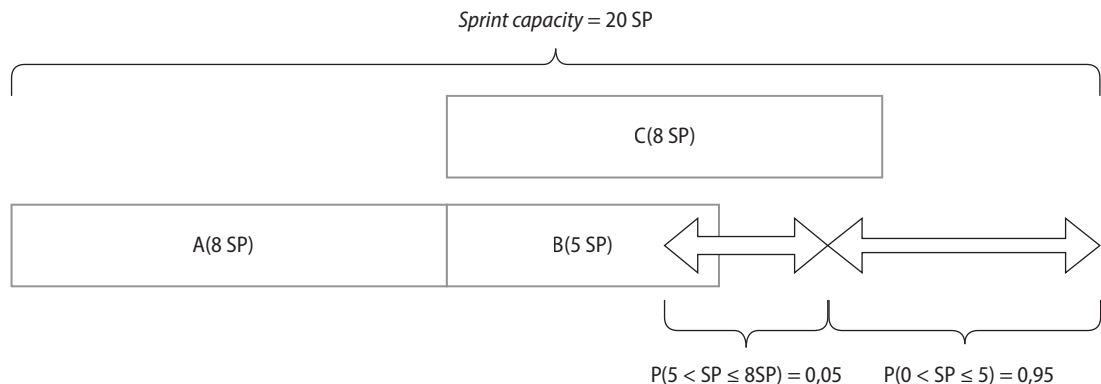
## 4. Rezultaty

W tym rozdziale przedstawiono prosty przykład ilustrujący możliwość zastosowania opisanego podejścia w przykładowym planie sprintu.

Planowanie rozpoczyna się od pustego rejestru sprintu, pustego zbioru ryzyk. Pojemność sprintu (ang. *sprint capacity*) jest stała i wynosi 20 *story points* (20 SP). Zadanie A szacowane na 8 SP (bez uwzględnienia ryzyk) zostaje zaplanowane w sprincie jako pierwsze. Wymaga ono złożonej integracji z zewnętrzną biblioteką, która została wydana po raz pierwszy rok temu. Ponieważ biblioteka jest nowa, należy spodziewać się pewnych problemów w korzystaniu z niej. Planując zadanie A, zespół decyduje się na wybór czynnika ryzyka zwane-

go „niedojrzałą technologią”. Jak wynika z historii, czynnikowi temu towarzyszyło często wystąpienie ryzyka określanego jako „opóźnienia w integracji”. Prawdopodobieństwo ryzyka ocenia się jako „duże”, a oddziaływanie jako „średnie”. W wyniku symulacji generuje się możliwe scenariusze przebiegu sprintu, tj. takie, w których zawarta jest informacja o tym, jak dane ryzyko może wydłużyć czas realizacji zadania, pomniejszając realną pojemność sprintu. Poziom akceptacji ryzyka wynosi 5% (tj. 5% skrajnych opóźnień jest odrzucane). Zgodnie z symulacją i na podstawie dostępnych danych historycznych „opóźnienia w integracji” o średnim oddziaływaniu mogą powodować wydłużenie czasu realizacji zadania od 0 do 5 SP w 95% przypadków (te wartości są brane pod uwagę) i od 5 do 8 SP w 5% przypadków (te wartości są odrzucane). Oznacza to, że realna maksymalna wartość zadań zaplanowanych w sprincie może wynosić od  $20 - 5 = 15$  SP do  $20 - 0 = 20$  SP w 95% przypadków (taka wartość jest uwzględniana) oraz od  $20 - 8 = 12$  SP do  $20 - 5 = 15$  SP w 5% przypadków (taka wartość jest pomijana). Można zaobserwować, jak wystąpienie ryzyka w postaci „opóźnień w integracji” może odpowiednio obniżyć maksymalną wartość zadań w sprincie (pojemność sprintu). W tej sytuacji zadanie C, szacowane na 8 SP, nie może być zaplanowane w sprincie, ponieważ w więcej niż 5% przypadków jego pojemność byłaby niewystarczająca (od 12 SP do 15 SP), aby pomieścić te dwie pozycje oszacowane na  $8 \text{ SP} + 8 \text{ SP} = 16 \text{ SP}$ . Zamiast tego zadanie B oszacowane na 5 SP może zostać zaplanowane w sprincie, ponieważ w 95% przypadków pojemność (od 15 SP do 20 SP) wystarczyłaby, aby pomieścić te dwa zadania oszacowane na  $8 \text{ SP} + 5 \text{ SP} = 13 \text{ SP}$  (rysunek 2).

Rysunek 2. Na zakończenie planowania sprint zawiera zadania A oraz B

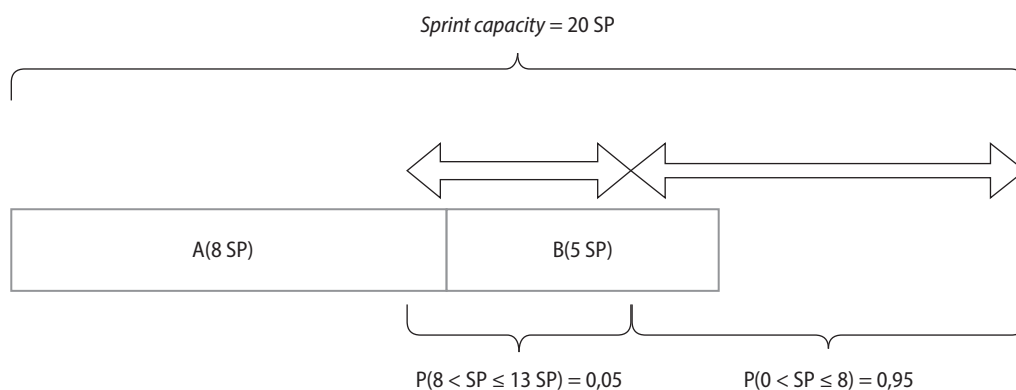


Źródło: opracowanie własne.

Rozpoczyna się sprint i po kilku dniach okazuje się, że nowa biblioteka (będąca częścią zadania A) jest trudna do automatycznego przetestowania i stąd pojawia się nowy czynnik ryzyka o nazwie „automatyzacja bardzo złożonego zadania”. Ten dodatkowy czynnik ryzyka sprawia, że oddziaływanie ryzyka „opóźnienia w integracji” szacowane jest jako duże, co powoduje dalsze wydłużenie czasu realizacji zadania A od 0 do 8 SP w 95% przypadków (poprzednio maksymalna wartość opóźnienia wynosiła 5 SP), a nawet o 8 SP do 13 SP w 5%

przypadków. W tej sytuacji maksymalna wartość zadań w sprincie zostaje zmniejszona do wartości zawierającej się między 12 SP a 20 SP w 95% przypadków lub wartości wynoszącej między 7 SP a 12 SP w 5% przypadków. Istnieje przy tym ryzyko, że sprint nie zostanie ukończony, ponieważ w więcej niż 5% przypadków (poziom akceptacji ryzyka) pojemność sprintu wynosząca od 7 do 12 SP (ograniczona do tej wartości ze względu na zwiększone ryzyko) nie będzie wystarczająca do pomieszczenia zadań A i B, które są szacowane na 13 SP (rysunek 3). W tej sytuacji należy zastosować akcje naprawcze.

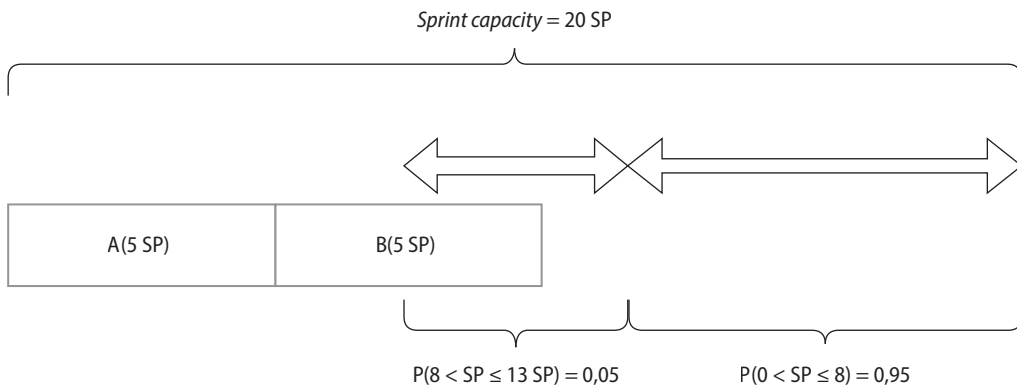
Rysunek 3. Wzrost szacowanego ryzyka w sprincie po kilku dniach



Źródło: opracowanie własne.

Założmy, że przyjęto następujące wagi:  $\omega_g = 6$ ,  $\omega_1^1 = 6$ ,  $\omega_1^2 = 5$ ,  $\omega_1^3 = 4$ ,  $\omega_1^4 = 3$ ,  $\omega_1^5 = 2$ ,  $\omega_1^6 = 1$ , a także, że pojemność sprintu jest przekroczona w 10% przypadków, w pozostałych 90% przypadków mieści się zaś w normie. Wartość kryterium *sprint value* jest w tych 10% przypadków niska, ponieważ składowa A1 (nieukończenie zadań w sprincie) osiąga najwyższą wartość w odniesieniu do kryterium *sprint value*. Koszt adaptacji wynosi tutaj 5 SP (jest to szacunkowa wartość zadania B, które może zostać nieukończone w sprincie). W takiej sytuacji kryterium ma wartość  $SV = 6 \cdot (8 SP + 5 SP) - 6 \cdot 5 SP = 48 SP$ . Nawet jeśli w 90% przypadków wartość kryterium będzie znacznie większa, to te 10% przypadków (czyli więcej niż poziom akceptacji ryzyka) powoduje, że należy udoskonalić plan sprintu za pomocą akcji naprawczych.

Założmy, że rekomendowane są dwie akcje naprawcze: A1 – zmniejszenie zakresu zadania A, skutkujące zmianą szacunku A z 8 SP do 5 SP (spadek o 3 SP), oraz A2 – obniżenie jakości zadania B, skutkujące zmianą szacunku z 5 SP do 2 SP (spadek o 3 SP). Ponieważ składnik jakościowy jest bardzo ważny w kontekście *sprint value*, akcja A1 zapewnia większy wzrost wartości tego kryterium. Ilustrują to następujące równania:  $SV(A1) = 6 \cdot (5 SP + 5 SP) - 3 \cdot 3 SP = 60 SP - 9 SP = 51 SP$  oraz  $SV(A2) = 6 \cdot (8 SP + 2 SP) - 5 \cdot 3 SP = 60 SP - 15 SP = 45 SP$ . Nawet jeśli prawdopodobieństwo wzrostu *sprint value* będzie nieco mniejsze niż w przypadku zastosowania akcji A2, to wybrana zostanie ostatecznie akcja A1 spośród dwóch rekomendowanych działań naprawczych (rysunek 4).

Rysunek 4. Wzrost wartości kryterium *sprint value* po zastosowaniu akcji naprawczych

Źródło: opracowanie własne.

Na podstawie opisanej koncepcji oraz przedstawionego przykładu można stwierdzić, że w *scrum* istnieje możliwość zastosowania ilościowego zarządzania ryzykiem, opartego na jawnej informacji o ryzyku podawanej na wejściu w formie wyrażen lingwistycznych. W dalszej części rozważań przeanalizowano właściwości takiego podejścia oraz jego zalety.

## 5. Dyskusja

Podejście zaprezentowane w tym artykule pozwala w sposób jawny zarządzać różnymi przejawami ryzyka w sprintach. Jest ono dość proste w użyciu: jawne informacje o ryzyku w postaci czynników i prognoz ryzyka związanych ze sprintem stanowią dane wejściowe do planowania. Wynikiem są akcje korygujące plan sprintu, które pozwalają zwiększyć wartość zaproponowanego w tej pracy kryterium, zwanego *sprint value*. Akcje rekomendowane w ramach tej metody uwzględniają również prawdopodobieństwo takiej poprawy.

Dane wejściowe do planowania są podawane w postaci rozmytej, ponieważ jest to łatwy sposób wyrażenia niepewnego charakteru ryzyka, co jest również szeroko rozpowszechnione w literaturze [np. PMI, 2021; Gemmer, 1997; Morrison, 2018; Amaral, 2019; Tiwana, 2004]. Podanie precyzyjnych szacunków ryzyka w formie wartości (np. opóźnień) byłoby dla zespołu trudne do osiągnięcia. Z uwagi na iteracyjny charakter sprintów bardziej pożądane jest tu użycie wyrażen lingwistycznych, które są kwantyfikowane *post factum* na koniec każdego sprintu, a następnie wykorzystywane do kolejnych symulacji. Nowatorski element stanowią w tym przypadku zaproponowane kryterium optymalizacyjne oraz akcje korygujące. Akcje i kryteria są rozpatrywane na poziomie sprintu.

Opisany wyżej proces planowania obejmuje jawny sposób zarządzania ryzykiem w przeciwieństwie do standardowej metodyki *scrum*, gdzie akcje są przeprowadzane w sposób ukryty. Zmiana ta wydaje się kluczowa dla lepszego radzenia sobie z niepewnością, a jej głównym celem jest poprawa planowania sprintu. Takie podejście może być pomocne w przypadku

niedoświadczonych zespołów *agile* [Standish Group International, 2015] lub zespołów, które pracują przy wyższym niż zwykle poziomie złożoności i niepewności sprintu (np. zespoły badawczo-rozwojowe pracujące w środowisku *scrum*).

Opisane rozwiązanie pozwala podjąć wymierną decyzję o zmianach w planie sprintu na podstawie informacji o ryzyku. Ponieważ kwantyfikacja ta jest obciążona niepewnością, wyraża się ją za pomocą prawdopodobieństwa. Parametrem może być w tym przypadku minimalny poziom prawdopodobieństwa, dzięki czemu rekomendowane będą tylko te akcje naprawcze, które poprawiają wartość *sprint value* z prawdopodobieństwem większym niż zadany parametr. Lista akcji zaproponowanych w tym artykule może być dowolnie rozszerzana.

Wiele z sugerowanych akcji naprawczych opiera się na zmianach zakresu, które są powszechnie stosowane w zarządzaniu projektami w odpowiedzi na opóźnienia czasowe [np. Shirazi, 2017; Nibyza, 2015], także w kontekście podejścia *agile* [np. Cao, 2010]. Kwantyfikowalne i oparte na rekomendacjach podejście do sugerowania optymalnych zmian w planie sprintu, według najlepszej wiedzy autorów, nie zostało jednak spopularyzowane. Według zaproponowanej w tej pracy koncepcji ryzykiem zarządza się poprzez rekomendowanie zmian w planie, które opiera się na kryterium *sprint value*. Kryterium to może być dostosowywane przez zespół poprzez przypisywanie mu różnych wag. W przypadku projektów *scrum* składnik czasowy (z kryterium *sprint value*) powinien być bardzo ważny, ponieważ zespoły *scrum* chciałyby dostarczać sprinty na czas, aby zapewnić sobie satysfakcję po stronie klienta. Równie istotny wydaje się także element jakościowy, gdyż poprzez jego minimalizację zmniejsza się ryzyko przyszłych poprawek. Dla tych dwóch wspomnianych kryteriów należy rozważyć najwyższe wartości wag w przypadku *sprint value*.

Uwzględnienie opisanego podejścia wymaga zmiany koncepcji planowania sprintu. W standardowym modelu planowania sprintu ryzyko jest zwykle uwzględniane w szacunkach [np. Ghazali, 2018]. Przedstawione w tej pracy badania sugerują zmianę standardowego podejścia poprzez oddzielne rozpatrywanie zadań w sprincie na potrzeby zarządzania ryzykiem. Powinno to umożliwić obliczenie prędkości sprintu (ang. *sprint velocity*) z większą dokładnością, ponieważ prędkość sprintu jest tutaj cechą zespołu, istniejącą niezależnie od poziomu ryzyka ukrytego w sprincie. Przy rozpatrywaniu ryzyka i szacunków oddzielnie, w sposób opisany w niniejszym artykule, wartości prędkości sprintu obliczone dla kolejnych akcji nie będą miały w sobie niepewności związanej z poszczególnymi przejawami ryzyka występującymi w sprincie.

## 6. Podsumowanie

Planowanie stanowi ważną część zarządzania projektami. Ulepszenia w tym zakresie są wymieniane wśród najczęściej cytowanych wyzwań dotyczących zwinnego tworzenia oprogramowania [Chloros, 2022]. Ponieważ nie jest oczywiste, czy niejawne zarządzanie ryzykiem wbudowane w podejście *agile* jest właściwe, niektórzy autorzy sugerują, że ryzykiem należy

zarządzać w sposób jawny [Kuchta, 2013]. W artykule zastosowano drugie z wymienionych podejść. Należy przy tym zwrócić uwagę na fakt, że niejawny sposób traktowania ryzyka (standardowy dla *agile*) jest prostszy niż jawny. Możliwość kontrolowania ryzyka jest jednak niewątpliwą zaletą podejścia jawnego. Przegląd literatury pokazuje, że w kontekście planowania procesu związanego z tworzeniem oprogramowania ryzyko może być uwzględniane na różne sposoby. Najprostsze rozwiązania opierają się na szacunkach. Ryzyko nie jest jednak w tym przypadku zarządzane, lecz ukryte w zadaniach. W literaturze istnieje też kilka gotowych i prostych w użyciu metod planowania, które umożliwiają jawne zarządzanie ryzykiem, choć nie zawsze da się je zastosować w *scrum*.

Zaproponowane w niniejszym artykule podejście pozwala zarządzać ryzykiem poprzez rekomendowanie akcji naprawczych w planie sprintu, które odzwierciedlają aktualne informacje o ryzyku. Na podstawie dokonanego przeglądu literatury można stwierdzić, że podobne metody nie zostały jeszcze przebadane ani zaproponowane, a w szczególności brakuje zaś narzędzi odpowiednich do planowania sprintu na podstawie informacji o ryzyku. Jedną z dostępnych metod, uwzględniającą jawne zarządzanie ryzykiem i opierającą się na rekomendacjach [Filippetto, 2021], jest zbyt ogólna i nie uwzględnia cech specyficznych dla metodyki *scrum* (jest stosowana tylko w kontekście zwinnym). Z kolei przedstawione w artykule podejście wspiera kierowników projektów w podejmowaniu wymiernych decyzji dotyczących ryzyka związanego ze sprintem. Zostało ono zilustrowane na konkretnym przykładzie, a także przeanalizowane pod kątem zalet. Za jego pomocą pokazano, jak zoptymalizować plan sprintu z uwzględnieniem jawnych informacji o ryzyku. Podejście to ma tę zaletę, że jest proste. Rekomendowane akcje naprawcze przeprowadzane są na podstawie kryterium optymalizacji planu sprintu, które można dostosować do potrzeb projektu poprzez ustawienie wag komponentów optymalizacyjnych.

## Bibliografia

### Wydawnictwa zwarte

1. Chloros, D., Gerogiannis V.C., Kakarontzas G. (2022). Use of Software and Project Management Metrics in Agile Software Development Methodologies. A Systematic Mapping Study. W: *ESSE '22: Proceedings of the 2022 European Symposium on Software Engineering* (s. 25–32). New York: Association for Computing Machinery.
2. Dingsøyr, T., Petit, Y. (2021). 4 Managing Layers of Risk: Uncertainty in Large Development Programs Combining Agile Software Develop. W: *Project Risk Management. Volume II: Managing Software Development Risk* (s. 75–96), K.J. Engemann, R.V. O'Connor (Eds.). Boston–Berlin: De Gruyter Oldenbourg.
3. Hammad, M., Inayat, I. (2018). Integrating Risk Management in Scrum Framework. W: *International Conference on Frontiers of Information Technology* (s. 158–163). Islamabad.
4. Higuera, R.P., Haimes, Y.Y. (1996). *Software Risk Management*, Technical Report CMU/SEI-96-TR-012. Pittsburgh: Software Engineering Institute.

5. Islam, S. (2009). Software Development Risk Management Model: A Goal Driven Approach. W: *Proceedings of the ESEC/FSE Doctoral Symposium '09* (s. 5–8). New York: Association for Computing Machinery.
6. Jabangwe, R., Nurdiani, I. (2010). *Global Software Development Challenges and Mitigation Strategies. A Systematic Review and Survey Results*, master thesis in School of Computing in Ronneby.
7. Menezes, J.V. de (2019). *Measuring Risks in Software Development Projects*, PhD thesis presented to the Center of Informatics of Federal University of Pernambuco.
8. Nibyiza, F., Shukla, J., Ndabaga, E. (2015). *Analysis of Project Scope Change Management as a Tool for Project Success*, Case Study of Akazi Kanoze Projects (2011–2014).
9. Nogueira, M., Machado, R.J. (2014). Importance of Risk Process in Management Software Projects in Small Companies. W: *Advances in Production Management Systems. Innovative and Knowledge-Based Production Management in a Global-Local World. APMS 2014. IFIP Advances in Information and Communication Technology* (vol. 439, s. 358–365), B. Grabot, B. Vallespir, S. Gomes, A. Bouras, D. Kiritsis (Eds). Heidelberg: Springer.
10. PMI (2021). *A Guide to the Project Management Body of Knowledge and the Standard for Project Management*.
11. Prater, J.B., Kirytopoulos, K., Ma, T. (2019). An Investigation of Estimation Techniques for Information Technology Projects. W: *2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)* (s. 779–783). Macao.
12. Tsiara, E. Siakas, K. (2016). Investigating the Application of Risk Management in Greek IT Companies. W: *Systems Quality: Trends and Practices, BCS Quality Specialist Group's Annual International 24th Software Quality Management (SQM) Conference* (s. 225–235), K. Phalp, V. Katos, S. Meacham, M. Toss, G. Staples (Eds.). Bournemouth: British Computer Society.

### Artykuły naukowe

1. Abioye, T.E., Arogundade, O.T., Misra, S., Akinwale, A.T., Adeniran, O.J. (2020). Toward Ontology-Based Risk Management Framework for Software Projects: An Empirical Study, *Journal of Software: Evolution and Process*, 32, s. 1–24.
2. Alfozan, T., Al-Shehab, A., Gaderrab, H. (2020). Risk Factors Taxonomy in Software Development Projects: Study from Kuwait, *Journal of Theoretical and Applied Information Technology*, 98(09), s. 1513–1522.
3. Al-Shehab, A.J., Al-Fozan, T., Gadelrab, H.F. (2021). Most Severe Risk Factors in Software Development Projects in Kuwait, *Indonesian Journal of Electrical Engineering and Computer Science*, 21, s. 591–600.
4. Alves, L.M., Souza, G., Ribeiro, P., Machado, R.J. (2021). Longevity of Risks in Software Development Projects: A Comparative Analysis with an Academic Environment, *Procedia Computer Science*, 181, s. 827–834.
5. Amaral, A., Elias, G. (2019). A Risk-Driven Multi-Objective Evolutionary Approach for Selecting Software Requirements, *Evolutionary Intelligence*, 12, s. 421–444.
6. Asif, M., Ahmed, J. (2020). A Novel Case Base Reasoning and Frequent Pattern Based Decision Support System for Mitigating Software Risk Factors, *IEEE Access*, 8, s. 102278–102291.



7. Aslam, A., Ahmad, N., Saba, T., Almazayad, A.S., Rehman, A., Anjum, A., Khan, A. (2017). Decision Support System for Risk Assessment and Management Strategies in Distributed Software Development, *IEEE Access*, 5, s. 20349–20373.
8. Beecham, S., Clear, T., Lal, R., Noll, J. (2021). Do Scaling Agile Frameworks Address Global Software Development Risks? An Empirical Study, *The Journal of Systems and Software*, 171, s. 1–30.
9. Boban, M., Pozgaj, Z., Sertić, H. (2004). Strategies for Successful Software Development Risk Management, *Management*, 8(2), s. 77–91.
10. Boehm, B.W. (1991). Software Risk Management: Principles and Practices, *IEEE Software*, 8(1), s. 32–41.
11. Cao, L., Ramesh, B., Abdel-Hamid, T. (2010). Modeling Dynamics in Agile Software Development, *ACM Transactions on Management Information Systems*, 1(5), s. 5–25.
12. Chaouch, S., Mejri, A., Ghannouchi, S.A. (2019). A Framework for Risk Management in Scrum Development Process, *Procedia Computer Science*, 164(1), s. 187–192.
13. Dönmez, D., Grote, G. (2018). Two Sides of the Same Coin – How Agile Software Development Teams Approach Uncertainty as Threats and Opportunities, *Information and Software Technology*, 98, s. 94–111.
14. Esteki, M.H., Gandomani, T.J., Farsani, H.K. (2020). A Risk Management Framework for Distributed Scrum Using PRINCE2 Methodology, *Bulletin of Electrical Engineering and Informatics*, 9(3), s. 1299–1310.
15. Fang, C., Marle, F. (2012). A Simulation-Based Risk Network Model for Decision Support in Project Risk Management, *Decision Support Systems*, 52(3), s. 635–644.
16. Filippetto, A.S., Lima, R., Barbosa, J.L.V. (2021). A Risk Prediction Model for Software Project Management Based on Similarity Analysis of Context Histories, *Information and Software Technology*, 131(1), s. 1–18.
17. Gemmer, A. (1997). Risk Management: Moving Beyond Process, *Computer*, 30(5), s. 33–43.
18. Georgiev, V., Stefanova, K. (2014). Software Development Methodologies for Reducing Project Risks, *Economic Alternatives*, 2, s. 104–113.
19. Ghazali, S.N., Salim, S.S., Inayat, I., Hamid, S.H. (2018). A Risk Poker Based Testing Model For Scrum, *Computer Systems Science and Engineering*, 33(3), s. 169–185.
20. Han, W. (2015). Discriminating Risky Software Project Using Neural Networks, *Computer Standards and Interfaces*, 40, s. 15–22.
21. Hartono, B. (2018). From Project Risk to Complexity Analysis: A Systematic Classification, *International Journal of Managing Projects in Business*, 11, s. 734–760.
22. Hazir, O., Ulusoy, G. (2020). A Classification and Review of Approaches and Methods for Modeling Uncertainty in Projects, *International Journal of Production Economics*, 223, s. 1–50.
23. Hijazi, H., Al-grainy, S., Muaidi, H., Khmour, T. (2014). Risk Factors in Software Development Phases, *European Scientific Journal*, 10(3), s. 213–232.
24. Hsieh, M., Hsu, Y., Lin, C. (2018). Risk Assessment in New Software Development Projects at the Front End: A Fuzzy Logic Approach, *Journal of Ambient Intelligence and Humanized Computing*, 9, s. 295–305.
25. Ibadov, N. (2019). Construction Project Planning Under Fuzzy Time Constraint, *International Journal of Environmental Science and Technology*, 16(9), s. 4999–5006.

26. Iftikhar, A., Alam, M.M., Ahmed, R., Musa, S., Su'ud, M.M. (2021). Risk Prediction by Using Artificial Neural Network in Global Software Development, *Computational Intelligence and Neuroscience*, s. 1–25.
27. Klein, G., Jiang, J.J. (2000). Software Development Risks to Project Effectiveness, *Journal of Systems and Software*, 52(1), s. 3–10.
28. Kosztyan, Z.T., Jakab, R., Novak, G., Hegedus, C. (2020). Survive IT! Survival Analysis of IT Project Planning Approaches, *Operations Research Perspectives*, 7, s. 1–17.
29. Kuchta, D., Walczak, W. (2013). Risks Characteristic of Agile Project Management Methodologies and Responses to Them, *Operations Research and Decisions*, 23(4), s. 75–95.
30. Kuo, R.J., Nugroho, Y., Zulvia, F.E. (2019). Application of Particle Swarm Optimization Algorithm for Adjusting Project Contingencies and Response Strategies under Budgetary Constraints, *Computers and Industrial Engineering*, 135, s. 254–264.
31. Marchwicka, E. (2020). A Technique for Supporting Decision Process of Global Software Project Monitoring and Rescheduling Based on Risk Analysis, *Journal of Decision Systems*, 29(1), s. 398–412.
32. Menezes, J., Gusmão, C.M., Moura, H.P. (2018). Risk Factors in Software Development Projects: A Systematic Literature Review, *Software Quality Journal*, 27, s. 1149–1174.
33. Morrison, P., Move, D., Pandita, R., Williams, L. (2018). Mapping the Field of Software Life Cycle Security Metrics, *Information Software Technology*, 102, s. 146–159.
34. Mousaei, M., Gandomani, T.J. (2018). A New Project Risk Management Model Based on Scrum Framework and PRINCE2 Methodology, *International Journal of Advanced Computer Science and Applications*, 9(4), s. 442–449.
35. Neves, S.M., de Silva, C.E. (2016). Risk Management Applied to Software Development Projects in Incubated Technology-Based Companies: Literature Review, Classification, and Analysis, *Gestão and Produção*, 23(4), s. 798–814.
36. Nicolas, J., Carrillo De Gea, J.M., Nicolás B., Fernández-Alemán J. L., Toval A. (2018). On the Risks and Safeguards for Requirements Engineering in Global Software Development: Systematic Literature Review and Quantitative Assessment, *IEEE Access*, 6, s. 59628–59656.
37. Rahman, M.H., Chakraborty, R.K., Ryan, M.J. (2021). Managing Uncertainty and Disruptions in Resource Constrained Project Scheduling Problems: A Real-Time Reactive Approach, *IEEE Access*, 9, s. 45562–45586.
38. Shirazi, F., Kazemipoor, H., Tavakkoli-Moghaddam, R. (2017). Fuzzy Decision Analysis for Project Scope Change Management, *Decision Science Letters*, 6, s. 395–406.
39. Suda, K.A., Rani, N.A.B. (2010). The Importance of 'Risk Radar' in Software Risk Management: A Case of a Malaysian Company, *International Journal of Business and Social Sciences*, 1, s. 262–272.
40. Sundararajan, S., Marath, B., Vijayaraghavan, P.K. (2019). Variation of Risk Profile across Software Life Cycle in IS Outsourcing, *Software Quality Journal*, 27, s. 1563–1582.
41. Taipalus, T., Seppänen, V., Pirhonen, M. (2020). Uncertainty in Information System Development: Causes, Effects, and Coping Mechanisms, *Journal of Systems and Software*, 168, s. 1–20.
42. Takagi Y., Mizuno O., Kikuno T. (2005). An Empirical Approach to Characterizing Risky Software Projects Based on Logistic Regression Analysis, *Empirical Software Engineering*, 10(4), s. 495–515.

43. Tavares, B.G., da Silva, C.E., de Souza, A.D. (2017a). Risk Management Analysis in Scrum Software Projects, *International Transactions in Operational Research*, 5, s. 1884–1905.
44. Tavares, B.G., da Silva, C.E., de Souza, A.D. (2017b). Risk Management in Scrum Projects: A Bibliometric Study, *Journal of Communications Software and Systems*, 13(1), s. 1–8.
45. Tiwana, A., Keil, M. (2004). The One-Minute Risk Assessment Tool, *Communication of the ACM*, 47(11), s. 73–77.
46. Trzeciak, M. (2021). Sustainable Risk Management in IT Enterprises, *Risks*, 9(7), s. 135.
47. Vizzini, G., Muriana, C. (2017). Project Risk Management: A Deterministic Quantitative Technique for Assessment and Mitigation, *International Journal of Project Management*, 35, s. 320–340.
48. Wallace, L.G., Keil, M. (2004). Software Project Risks and Their Effect on Outcomes, *Communications of the ACM*, 47(4), s. 68–73.
49. Yim, R.L., Castaneda, J.M., Doolen, T.L., Tumer, I.Y., Malak, R.J. (2015). A Study of the Impact of Project Classification on Project Risk Indicators, *International Journal of Project Management*, 33, s. 863–876.

### Materiały internetowe

1. Beck, K., Beedle, M., Bennekum, A. van, Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, K., Sutherland, J. (2023). *Manifesto for Agile Software Development*, <http://agilemanifesto.org> (dostęp: 10.01.2023).
2. Schwaber, K., Sutherland, J. (2023). *The Scrum Guide*, <http://www.scrum.org> (dostęp: 10.01.2023).
3. Standish Group International (2023). *Chaos Report 2015*, <http://www.standishgroup.com> (dostęp: 10.01.2023).

---

## Adaptational Sprint planning basing on risk information

---

### Abstract

Risk management in (standard) Scrum is handled only implicitly. The authors' literature review shows that there is a lack of quantitative approaches to explicit risk management in Scrum. The aim of this work is to analyze whether it is possible to manage risk in Scrum in a quantitative way, based on explicit risk information provided in a linguistic form as well as analyzing the advantages of this approach. A concept enabling the quantification of risks in a Sprint has been developed. The approach presented in the article makes it easier to adapt a Sprint plan. In this approach, a set of possible corrective actions of the plan are returned as an output. These actions are quantified in terms of the criterion of business value for the Customer, which is called the Sprint Value. The approach is presented

on an illustrative example. The research fills the research gap that was mentioned above. Based on the research conducted, it was found that quantitative risk management can be used in Scrum, based on explicit risk information provided in a form of linguistic expressions.

**Keywords:** Scrum methodology, uncertainty, software projects, recommendation systems, risk management

---