

ResearchCruiseApp

Dobór i adaptacja metodyki

Wersja	1.0.0
Data utworzenia obecnej wersji	2025-06-05
Data utworzenia dokumentu	2025-06-05
Stanisław Nieradko	s193044
Krzysztof Nasuta	s193328
Paweł Pstrągowski	s193473
Bartłomiej Krawisz	s193319
Filip Dawidowski	s193433

1. O projekcie i produkcie

- **Projekt:** System zarządzania rejsami statku RV Oceanograf biura Armatora Uniwersytetu Gdańskiego.
- **Produkt:** Aplikacja webowa do zarządzania rejsami statku biura Armatora Uniwersytetu Gdańskiego.

Produkt skierowany jest do Biura Armatora oraz pracowników uczelni, którzy chcą zorganizować badawczy rejs statkiem. Aplikacja umożliwia im wypełnienie wymaganych formularzy oraz zarządzanie terminami rejsów.

2. Ocena według modelu uproszczonego - 5 kryteriów

Na podstawie pięciu kryteriów modelu uproszczonego Boehma i Tunera została wstępnie określono preferowany typ metodyki (zwinna vs zdyscyplinowana).

2.1 Rozmiar

Projekt dotyczy stworzenia aplikacji webowej dla specyficznej jednostki (RV Oceanograf) i określonej grupy użytkowników (Biuro Armatora, pracownicy UG). Zakres funkcjonalny, obejmujący zarządzanie formularzami i terminami rejsów oznacza projekt małej wielkości.

- Ocena: Mały.
- Uzasadnienie: Projekty o mniejszej skali często korzystają na elastyczności i szybkości iteracji charakterystycznych dla metodyk zwinnych. Złożoność nie wydaje się na tyle duża, aby wymuszać ciężkie, sformalizowane podejście.
- Wskazanie: Preferencja dla metodyk zwinnych.

2.2 Krytyczność

System będzie zarządzał rejsami badawczymi, co wiąże się z koordynacją zasobów, bezpieczeństwem danych i potencjalnymi konsekwencjami finansowymi lub operacyjnymi w przypadku błędów (np. błędne rezerwacje, utrata danych z formularzy).

- Ocena: Średnia (kluczowe fundusze).
- Uzasadnienie: Choć nie jest to system krytyczny dla życia, jego awaria lub błędy mogą prowadzić do zakłóceń w badaniach naukowych, strat finansowych i problemów organizacyjnych dla Uniwersytetu. Wymaga to zapewnienia odpowiedniej jakości i niezawodności.
- Wskazanie: Skłania się ku metodykom zdyscyplinowanym w zakresie zapewnienia jakości, testowania i zarządzania ryzykiem, ale nie wyklucza podejścia zwinnego z odpowiednimi adaptacjami.

2.3 Dynamika

Wymagania dotyczące podstawowych funkcjonalności (formularze, harmonogramy) mogą być na początku stosunkowo stabilne. Szczegóły implementacyjne, interfejs użytkownika, specyficzne pola w formularzach czy reguły walidacji powinny być dostosowywane wraz z napływem informacji zwrotnej od użytkowników oraz klientów, a także dostosowane do ich procesów.

- Ocena: Średnia do wysokiej.
- Uzasadnienie: Aplikacje webowe zwykle podlegają zmianom wynikającym z feedbacku użytkowników. Możliwość adaptacji do zmieniających się potrzeb lub odkrywania nowych usprawnień w trakcie projektu jest istotna.

- Wskazanie: Preferencja dla metodyk zwinnych, dobrze radzących sobie ze zmieniającymi się wymaganiami.

2.4 Osoby

Projekt będzie realizowany dla Biura Armatora Uniwersytetu Gdańskiego. Zespół deweloperski posiada odpowiednie kompetencje do tworzenia aplikacji webowych. Kluczowa jest ich dostępność i zaangażowanie Biura Armatora w procesie definiowania wymagań funkcjonalnych i użytkowych. Dodatkowo zastosowaniu metodyk zwinnych sprzyja niewielki zespół deweloperski 5 deweloperów i 1 opiekun projektu.

- Ocena: Korzystna dla zwinności.
- Uzasadnienie: Metodyki zwinne opierają się na samoorganizujących się zespołach i bliskiej współpracy z klientem. Jeśli te warunki są spełnione, efektywność projektu wzrasta.
- Wskazanie: Preferencja dla metodyk zwinnych, przy założeniu aktywnego udziału Biura Armatora.

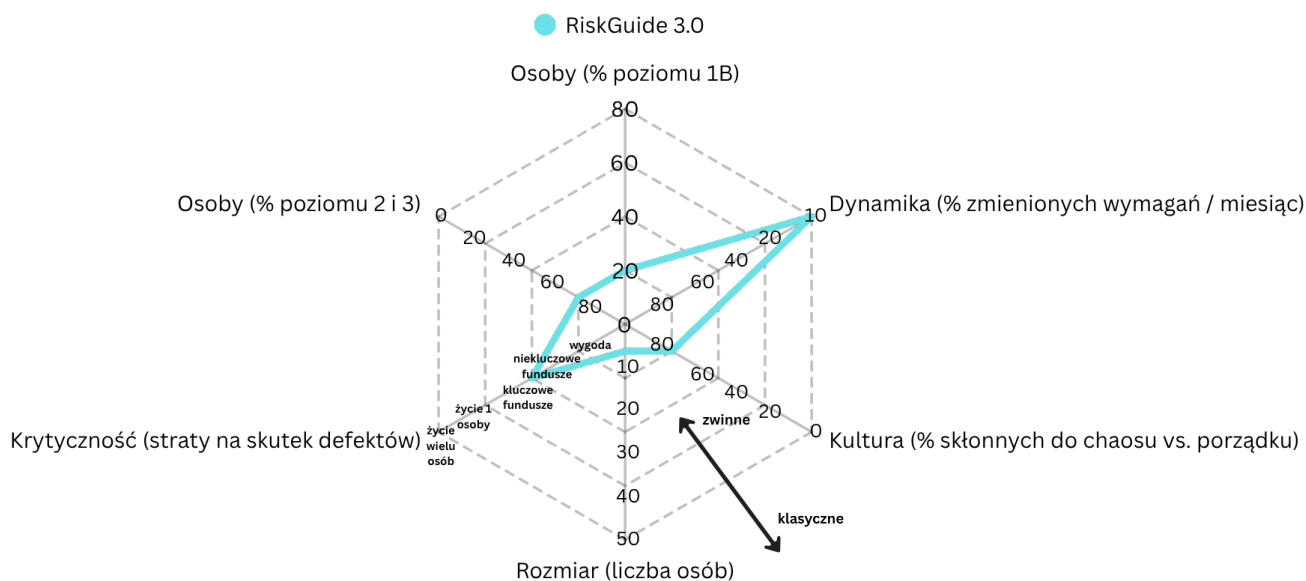
2.5 Kultura

Projekt realizowany jest dla środowiska akademickiego (Uniwersytet Gdański). Oznacza to sformalizowane procedury decyzyjne lub wymogi dokumentacyjne. Projekt ma na celu usprawnienie pracy, co sprzyja otwartości innowacyjne rozwiązania w zakresie zestawu technologii, narzędzi. Wszyscy deweloperzy poza jednym są skłonni do chaosu.

- Ocena: Mieszana.
- Uzasadnienie: Kultura uniwersytecka może być otwarta na innowację i jednocześnie posiadać cechy organizacji hierarchicznej. Biuro Armatora ma ustalone procedury, które system będzie musiał wspierać.
- Wskazanie: Metodyki zwinne mogą być efektywne, ale z koniecznością adaptacji do formalnych wymogów uczelni (np. w zakresie akceptacji etapów, dokumentacji).

2.6 Podsumowanie

Podsumowanie oceny uproszczonej: Projekt dobrze pasuje do metodyk zwinnych, szczególnie pod względem rozmiaru, dynamiki i ilości osób. Krytyczność i kultura organizacyjna wskazują na potrzebę włączenia pewnych elementów dyscypliny, szczególnie w obszarze zapewnienia jakości i dostosowania do procedur uczelnianych.



3. Ocena według zaadaptowanego modelu pełnego - 7 kryteriów

3.1 Zastosowanie

Główne cele (Primary Goals):

- Ocena projektu: Główne cele (stworzenie systemu do zarządzania rejsami, formularzami i terminami) są jasno określone. Jednakże, sposób ich realizacji i szczegółowa funkcjonalność nie jest znana i powinna zostać doprecyzowana wraz z tworzeniem produktem.
- Uzasadnienie: Projekt ma na celu dostarczenie konkretnej wartości użytkownikom (usprawnienie organizacji rejsów). Możliwość wczesnego dostarczania działających części systemu i zbierania feedbacku jest konieczna.
- Dopasowanie: Metodyki zwinne, umożliwiające iteracyjne odkrywanie i dostarczanie wartości.

Środowisko (Environment):

- Ocena projektu: Aplikacja webowa będzie działać w środowisku internetowym, serwer w chmurze, aplikacja webowa na komputerze i telefonach, dostępna dla Biura Armatora. Środowisko użytkowników (Biuro Armatora, naukowcy) mają specyficzne potrzeby i niedoprecyzowane oczekiwania.
- Uzasadnienie: Zrozumienie i adaptacja do środowiska pracy użytkowników jest kluczowa. Elastyczność w reagowaniu na ich potrzeby i integracja z istniejącymi procesami uczelnianymi są istotne.
- Dopasowanie: Metodyki zwinne, wspierające adaptację do środowiska i wymagań użytkowników.

3.2 Zarządzanie

Komunikacja (Communication):

- Ocena projektu: Efektywna komunikacja z Biurem Armatora (główny interesariusz) oraz przedstawicielami pracowników uczelni (użytkownicy końcowi) będzie kluczowa dla sukcesu aplikacji. Ze względu na dużo obowiązków ze strony klienta, komunikacja z nim oraz spotkania występują rzadko. Omawiane są na nich zagadnienia z różnych dziedzin dotyczących produktu.
- Uzasadnienie: Regularna informacja zwrotna, demonstracje postępów i wspólne rozwiązywanie problemów są niezbędne do stworzenia użytecznego narzędzia.
- Dopasowanie: W stronę metodyk zwinnych, które kładą nacisk na częstą i bezpośrednią komunikację z klientem/użytkownikami (w tym przypadku nie tak częsta komunikacja).

3.3 Techniczne

Wymagania (Requirements):

- Ocena projektu: Ogólny zakres jest znany (formularze, terminy). Szczegółowe wymagania dotyczące logiki biznesowej, pól formularzy, przepływów pracy, wyglądu UI, funkcjonalność aplikacji bez dostępu do internetu, uprawnień użytkowników wymagają doprecyzowania i mogą/będą ewoluować.
- Uzasadnienie: Trudno jest z góry zdefiniować wszystkie szczegóły. Iteracyjne podejście do zbierania i implementacji wymagań pozwoli na lepsze dopasowanie produktu do rzeczywistych potrzeb.
- Dopasowanie: Metodyki zwinne, akceptujące i zarządzające zmieniającymi się wymaganiami.

Wytwarzanie (Development/Construction):

- Ocena projektu: Tworzenie aplikacji webowej. Nowoczesne technologie webowe (.NET9 i React) dobrze współgrają z iteracyjnymi i przyrostowymi podejściami.
- Uzasadnienie: Możliwość szybkiego prototypowania, budowania komponentów i częstego wdrażania wersji testowych lub produkcyjnych (jeśli to możliwe) jest zaletą.
- Dopasowanie: Metodyki zwinne, wspierające iteracyjne wytwarzanie i częste dostarczanie działającego oprogramowania.

3.4 Osoby

Klient (Customer/Client):

- Ocena projektu: Klientami są pracownicy Biura Armatora. Ich zaangażowanie w proces jest kluczowe (testowanie, feedback, spotkania z deweloperami).
- Uzasadnienie: Aktywny udział klienta w definiowaniu priorytetów i akceptacji wykonanych prac jest fundamentem zwinności.
- Dopasowanie: Metodyki zwinne, wymagające bliskiej współpracy z klientem.

Kultura (Culture):

- Ocena projektu: Zespół projektowy jest otwarty na zwinne praktyki, a kluczowi interesariusze (Biuro Armatora) są gotowi na iteracyjną współpracę i regularny feedback.
- Uzasadnienie: Kultura współpracy, rodzaj problemu biznesowego, zaufania i otwartości na zmiany sprzyja sukcesowi projektów zwinnych.
- Dopasowanie: Metodyki zwinne, przy założeniu wspierającej kultury lub możliwości jej kształtowania w ramach projektu.

3.5 Podsumowanie oceny pełnej

Wszystkie siedem kryteriów w modelu pełnym również wskazuje na silne dopasowanie projektu do metodyk zwinnych. Szczególnie istotne jest to w kontekście wymagań, komunikacji i wytwarzania.

4. Model dostarczania produktu końcowego projektu

Biorąc pod uwagę charakter projektu (aplikacja webowa) oraz wnioski z poprzednich ocen, sugerowane są następujące modele dostarczania:

- **Przyrostowy (Incremental Delivery):** Jest to najbardziej odpowiedni model. Aplikacja może być dostarczana w kolejnych, funkcjonalnych częściach (przyrostach).

Na przykład:

- Moduł zarządzania użytkownikami i podstawowe formularze.
- Moduł zarządzania terminami rejsów (kalendarz, rezerwacje).
- Zaawansowane funkcje formularzy (np. walidacja, statusy).
- Raportowanie i funkcje administracyjne.
- Refaktoryzacja UX/UI.

Każdy przyrost dostarcza wartość użytkownikom i pozwala na zbieranie informacji zwrotnych.

- **Ciągły (Continuous Delivery):** W późniejszych fazach projektu, lub w fazie utrzymania, można dążyć do modelu ciągłego dostarczania, gdzie mniejsze zmiany i poprawki są wdrażane często i automatycznie. Na początkowym etapie rozwoju, model przyrostowy jest bardziej realistyczny.
- **Jednorazowy (Single Delivery):** Model nieodpowiedni ze względu na ryzyko niedopasowania produktu do potrzeb użytkowników i brak możliwości wczesnej weryfikacji.

Sugerowana metodyka wspierająca model przyrostowy: Metodyki zwinne takie jak Scrum lub Kanban naturalnie wspierają model przyrostowy. Scrum, z jego sprintami kończącymi się dostarczeniem potencjalnie wdrażalnego przyrostu produktu, wydaje się dobrym wyborem.

4.1. Brak zbędnych spotkań i ról.

- Scrum wymaga regularnych spotkań (planowanie sprintu, daily, review, retrospektywa) oraz wyznaczenie ról (Scrum Master, Product Owner), co tworzy dodatkowe obowiązki i zajmuje czas, który można przeznaczyć na pisanie funkcjonalności.
- Kanban nie narzuca ról - zespół sam decyduje, jak często i w jaki sposób się komunikuje (zdalnie przez komunikator discord), co pozwala ograniczyć poświęcony czas na rzeczy niepotrzebne z perspektywy produktu.

4.2. Elastyczność i szybkie reagowanie na zmiany.

- W Scrumie nie dodaje się nowych zadań w trakcie sprintu co jest ograniczeniem.
- W Kanbanie można dodawać nowe zadania w zależności od potrzeb.

4.3. Brak potrzeby planowania sprintów

- Scrum wymaga estymacji zadań i rozpisania ich na sprint(y) co jest czasochłonne (procentowo) w stosunku do rozmiaru pracy (mały projekt).
- W Kanbanie nie planuje się sprintów, zadania można podzielić i wykonywać po kolei.

4.4. Lepsze dopasowanie do trybu pracy zespołu

- Kanban sprawdza się w zespole gdzie każdy członek zna swoje zadania, a formalne role są zbędne (poza wybranym przedstawicielem do komunikacji z klientem dla wspólnego stanowiska zespołu).
- Pozwala na większą samodzielność i elastyczność w pracy.

Kanban okazuje się lepszym rozwiązaniem w przypadku tego projektu niż Scrum. Pozwala jeszcze lepsze dostosowywanie priorytetów, nie wymaga czasochłonnego szacowania i planowania sprintów. Dzięki temu zespół może klientowi dostarczyć zauważalnie lepszy i bardziej dopracowany produkt niż w przypadku Scruma.

5. Metodyka i jej adaptacja

Podsumowanie sugestii metodyki: Analiza projektu zarówno według modelu uproszczonego (5 kryteriów), jak i pełnego (7 kryteriów), a także preferowany model dostarczania produktu (przyrostowy), jednoznacznie wskazują na metodyki zwinne (agile) jako najbardziej odpowiednie dla projektu Systemu zarządzania rejsami statku RV Oceanograf.

Wybór konkretnej metodyki: Rekomendowaną bazową metodyką jest Kanban. Zapewnia on ramy dla iteracyjnego i przyrostowego wytwarzania produktu.

Adaptacja metodyki Kanban: Mimo ogólnego dopasowania do Kanban, projekt posiada cechy, które wymagają świadomej adaptacji metodyki.

Krytyczność systemu (średnia):

Niedopasowanie: Standardowy Kanban nie kładzie nacisku na formalne procesy testowania i dokumentację, które są istotne przy systemie zarządzającym rezerwacjami i danymi. Adaptacja:

- Włączenie do "Definition of Done" (DoD) kryteriów dotyczących przeprowadzenia szczegółowych testów funkcjonalnych i akceptacyjnych.
- Tworzenie i aktualizowanie niezbędnej dokumentacji użytkowej i technicznej, dodanie ich do zadań w tablicy Kanban.

Kultura organizacyjna środowiska uniwersyteckiego:

Niedopasowanie: Procedury decyzyjne, wymogi sprawozdawczości lub ograniczona dostępność niektórych interesariuszy mogą kolidować z dynamiką Kanbana (brak codziennych, regularnych spotkań). Adaptacja:

- Wyznaczenie Product Ownera z Biura Armatora, który ma uprawnienia do podejmowania decyzji i jest dostępny dla zespołu.
- Utrzymanie bardziej formalnej dokumentacji projektowej (np. harmonogram milestone'ów) dla potrzeb Biura Armatora jak i deweloperów, równoległe do tablicy Kanban.
- Organizowanie dodatkowych, formalnych spotkań przeglądowych dla szerszego grona interesariuszy po kluczowych kamieniach milowych.

Dostępność użytkowników (pracownicy uczelni):

Niedopasowanie: Regularne zaangażowanie szerokiej grupy pracowników uczelni może być trudne do osiągnięcia ze względu na dużą ilość obowiązków z ich strony. Adaptacja:

- Product Owner (z Biura Armatora) pełni rolę głównego reprezentanta potrzeb użytkowników.

Wymagania dotyczące dokumentacji:

Niedopasowanie: Zwinne podejście preferuje “działające oprogramowanie ponad obszerną dokumentację”. Jednak system dla instytucji publicznej może wymagać odpowiedniej dokumentacji użytkowej i administracyjnej. Adaptacja:

- Włączenie zadań związanych z tworzeniem dokumentacji (np. samouczka, tooltipów) do Product Backlogu i realizowanie ich.
- Wykorzystanie narzędzi wspierających generowanie dokumentacji (np. z kodu - doxygen) tam, gdzie to możliwe (w celach utrzymania projektu, gdy pierwotni deweloperzy przestaną być odpowiedzialni za projekt).
- Regularne przeglądy i akceptacja dokumentacji przez Product Ownera lub wyznaczonych przedstawicieli.

Te adaptacje mają na celu wykorzystanie zalet metodyki Kanan przy jednoczesnym uwzględnieniu specyfiki projektu i kontekstu organizacyjnego Uniwersytetu Gdańskiego, minimalizując ryzyka i zwiększając szanse na sprawne dostarczenie produktu.