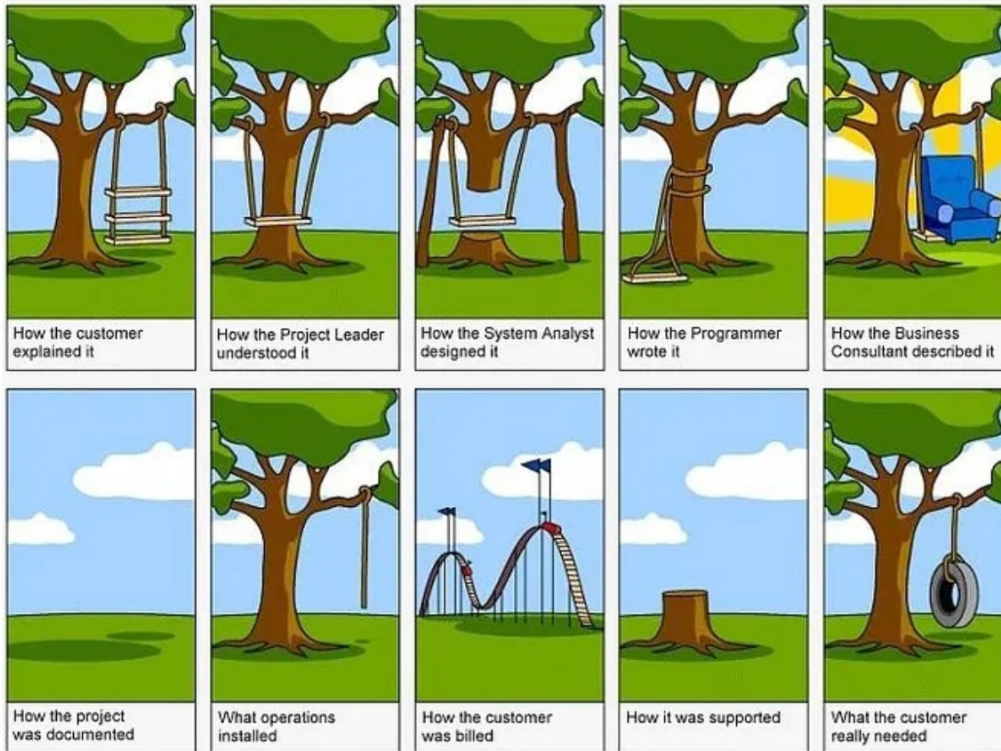


Analiza wymagań



Procesy inżynierii wymagań

DILBERT by Scott Adams



Analiza wymagań („co?”)

- **identyfikacja i specyfikacja wymagań względem systemu**
- **konstrukcja modelu logicznego oprogramowania**

- *zwykle prowadzone „top – down”*
- *iteracyjne uszczegóławianie*
- *potrzeba weryfikacji i walidacji*

Działania analizy wymagań prowadzone są w całym cyklu życia oprogramowania

Identyfikacja i specyfikacja wymagań w podejściu obiektowym – spodziewane produkty

Zasadniczy wynik:


- ✓ dokument Specyfikacji Wymagań (SWS)
*ustrukturyzowany zbiór informacji,
obejmujący wymagania wobec systemu*
- ✓ słownik pojęć (gdy przydatny)
- ✓ wsparcie notacjami modelującymi
 - dla nieobektowych
 - dla obiektowych (przypadki użycia)

Gdy przyrostowo:

- ✓ zgrubna specyfikacja wymagań (SWS)
 - ✓ słownik pojęć
 - ✓ wsparcie notacjami modelującymi
- ✓ szczegółowa specyfikacja wymagań
kolejnego przyrostu
 - ✓ wsparcie notacjami modelującymi

Inżynieria wymagań – oznacza, że działania są metodyczne, podlegają planowaniu oraz bazują na technikach systematycznych i powtarzalnych

Czynności w procesie identyfikacji i specyfikacji wymagań

- ⇒ identyfikowanie, pozyskiwanie wymagań
 - ⇒ dokumentowanie
 - ⇒ analizowanie
 - analiza rozszerzona
 - ⇒ specyfikacja
 - wsparcie notacjami modelującymi
 - ⇒ weryfikacja i walidacja
 - ⇒ pielęgnacja
-  *iteracje!*
- ⇒ konstrukcja modelu logicznego
 - ⇒ weryfikacja i walidacja
 - ⇒ pielęgnacja

Kto ich dokonuje?



Klient – analityk – ??

⇒ identyfikowanie, pozyskiwanie wymagań

Projektowanie SI

Typowe źródła wymagań

- ♦ Klient: zamawiający, użytkownicy, zarządzający, administratorzy, inni udziałowcy;
- ♦ Pośredni udziałowcy
- ♦ Literatura przedmiotu (dziedziny)
- ♦ Przepisy i kodeksy,
 - z dziedziny aplikacji, informatyczne i inne
- ♦ Plany strategiczne i biznesowe zamawiającego
- ♦ Dokumentacja istniejącego systemu
- ♦ Aktualny system – działanie, usterki, ...
- ♦ Analogiczne rozwiązania
- ♦ Wiedza, w tym - informatyczna



Projektowanie SI

Identyfikacja wymagań

- Poznanie literatury przedmiotu
- Wywiady (interviews) i dyskusje
- Kwestionariusze
- Analiza dokumentów
- Analiza istniejącego systemu
- Próbkowanie
- Obserwacje
- Prezentacje u dostawców i wizytowanie podobnych instalacji
- Symulacja punktów widzenia
- Wykorzystanie kilku spośród wskazanych metod

Problemy przy pozyskiwaniu wymagań

- Wiedza związana z wymaganiami często nie jest dostępna w postaci gotowej
- Wiedza ta może być rozproszona pomiędzy wiele źródeł
- Wydobyć jej od ludzi wymaga intensywnej komunikacji i nie jest łatwe
- Ludzie najczęściej prezentują subiektywny punkt widzenia
- Wiedza pochodząca z różnych źródeł może być wzajemnie sprzeczna
- Jak zapewnić kompletność?
- Jak wiernie oddać przekazane treści?



⇒ **dokumentowanie wymagań**

➤ Dokumentowanie wymagań

Są dwa podstawowe cele działań dokumentacyjnych:

- **udokumentowanie zgromadzonych wymagań i źródeł ich pochodzenia,**
- **nadanie wymaganiom reprezentacji przydatnej w dalszych etapach prac,** nie tylko na dalszych etapach inżynierii wymagań, ale także np. podczas działań konstrukcyjnych, walidacji kolejno powstających produktów czy tworzenia instrukcji eksploatacyjnych.

Stąd ogromna rola spójnego, czytelnego także dla udziałowców zapisywania wymagań oraz nadania im takiej reprezentacji, by zarządzanie udostępnianiem, zmianami i śledzeniem było możliwe efektywne.

Dokumentowanie wymagań

**Zrozumiałość i przejrzystość wypowiedzi uzyskuje się
wspierając wypowiedzi tekstowe za pomocą**

- **zestawień tabelarycznych**
- **ustrukturalizowanej postaci dokumentów**
- **notacji graficznych** – zwykle są tu stosowane
 - hierarchie funkcji i tablice powiązań,
 - schematy organizacji,
 - diagramy przepływu (ang. flowcharts)
 - tablice i diagramy decyzyjne
 - coraz częściej – scenariusze i przypadki użycia



⇒ analizowanie wymagań

Projektowanie SI

⇒ analizowanie

- **Wstępna analiza polega na przedstawieniu udokumentowanych wymagań udziałowcowi w celu ich sprawdzenia**

Czy to jest poprawny zapis tego co powiedziałeś? Czy to odzwierciedla to co miałeś na myśli? Czy moja interpretacja jest właściwa?

Daje to możliwość dokonania korekty tak, aby usunąć wątpliwości czy wymagania zostały jasno zrozumiane

- **Połączenie wymagań różnych udziałowców i ich analiza**

wykorzystanie standardowego wzorca dokumentu specyfikacji wymagań systemowych (SWS)

analiza pod kątem nadmiarowości, sprzeczności i konfliktów, względnej ważności wymagań, kompletności (brakujących wymagań)

analiza wpływu (analiza rozszerzona, ang. impact analysis)

wyjaśnienie niejednoznaczności, wypełnienie luk, potwierdzenie przyjętych założeń

- **Zbiorcza analiza: porównanie wymagań z zakresem projektu**

Czy zebrano wymagania od wszystkich udziałowców? Czy cały, wcześniej określony zakres projektu jest 'pokryty' wymaganiami? Czy wyłączone wymagania leżące poza zakresem?

Czy uwzględniono w wymaganiach biznesowy punkt widzenia (cele systemu)?

Czy dla wymagań zdefiniowano mierzalne kryteria akceptowalności?

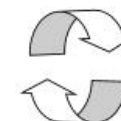
folia: 10

folia: 7

Projektowanie SI

⇒ analiza rozszerzona

- Analiza wzajemnego oddziaływania projektu i otoczenia
- Analiza potrzeb i wpływu udziałowców
- Analiza ryzyka (zagrożeń dla projektu)



Projekt Informatyczny

folia: 10

⇒ specyfikacja, sformułowanie *wymagania*

Wymaganie to (*Std IEEE 1233*)

- (A) warunek lub cecha (zdolność, ang. *capability*) **nieodzowna dla rozwiązania problemu lub osiągnięcia zamierzonego celu**;
- (B) warunek lub cecha, które muszą być spełnione bądź posiadane przez system, **aby być w zgodzie z kontraktem, standardem**, specyfikacją lub innym formalnie narzuconym dokumentem;
- (C) udokumentowana reprezentacja warunku lub zdolności wg (A) lub (B)

➤ według Standardu IEEE 1233 opis wymagania powinien zawierać:

- unikalny identyfikator wymagania,
- treść
- priorytet dla klienta
- niezbędność spełnienia wymagania (ang. *criticality*) dla sukcesu projektu (dla spełnienia innych wymagań),
- wykonalność wymagania – we wspólnej ocenie klienta i analityka
- ryzyko, w odniesieniu do prac projektu i do niespełnienia wymagania w systemie
- źródło(-a) wymagania,
- kategorię (typ) wymagania

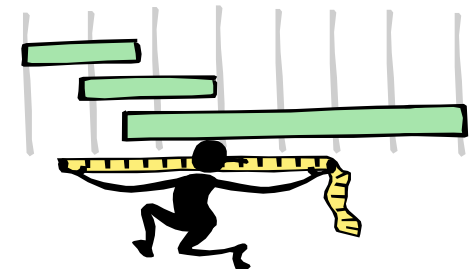
Mierzalność wymagań

Spełnienie wymagań pozwala ocenić osiągnięcie celów / ukończenie projektu – odwołanie do wymagań przy ocenie systemu (ocenie jego pewnej reprezentacji) ma sens tylko wtedy, gdy **wymagania są mierzalne**

- np. dotyczą wielkości liczbowych, jak pojemność bazy danych, liczba abonentów itp,
- są określone konkretnie, np. protokoły, formaty struktury danych
- są zdefiniowane poprzez inne, mierzalne cechy

Wymagania stanowią kryteria dla walidacji i akceptacji systemu

- *testy systemowe i walidacyjne*
- *testy akceptacyjne*



Przykład specyfikacji wymagania

Wymagania dzielone są na standardowe kategorie:

- A. Wymagania ogólne
- B. Wymagania dotyczące systemu (*funkcjonalne, niefunkcjonalne*)
- C. Ograniczenia środowiska pracy systemu
- D. Wymagania projektowo - wdrożeniowe

T. Łutowicz: System monitorowania problemów procesu dydaktycznego

Identyfikator:	F-E-3	Priorytet:	1	Status:	Zaakceptowany
Tytuł:	Dodawanie problemów				
Opis:	<p>Dodawanie nowego problemu.</p> <p>Automatycznie problemowi nadany powinien być unikalny numer ID, data jego dodania oraz status problemu ustawiony na 'Nowe'.</p> <p>Użytkownik określa następujące wymagane wartości atrybutów:</p> <ul style="list-style-type: none"> - Tytuł problemu - Treść problemu - Adresat – pojedynczy użytkownik lub grupa użytkowników (możliwe będzie dodanie filtrowania adresatów po grupie) - Kategoria problemu <p>Opcjonalnie użytkownik może również:</p> <ul style="list-style-type: none"> - ustawić pole priorytetu problemu (<i>Bardzo Ważne / Ważne / Normalne</i>) - dodać załącznik/załączniki do problemu - dodać osobę, która poza adresatem może być zainteresowana danym problemem 				
Źródło:	Imię, nazwisko, stanowisko lub inne wskazanie				
Powiązane wymagania:	F-E-4 Ustawianie adresata F-E-6 Dodawanie załączników F-O-1 Zmiana aktualnego stanu problemu F-E-11 Dodawanie osoby zainteresowanej do problemu N-1 Interfejs użytkownika				

⇒ specyfikacja zbioru wymagań

Dokument SWS

Typowy zakres:

- ✓ O dokumencie
- ✓ Cele biznesowe, kontekst, udziałowcy i użytkownicy, zakres informacji, dane
- ✓ Specyfikacja poszczególnych wymagań, *ustrukturalizowana, najczęściej według ich kategorii i ukazująca zależności pomiędzy wymaganiami*
- ✓ Elementy powiązane, jak *słownik pojęć, plany testów akceptacyjnych,...*

➔ Szablon SWS !

Numer zlecenia i nazwa i akronim projektu: (nazwa projektu)		Zleceniodawca: (nazwa klienta zewnętrznego)	Zleceniobiorca: (P/J/WSTK)
Zespół projektowy: (członkowie zespołu projektowego)		Kierownik projektu: (kierownik zespołu)	Opiekun projektu: (uczelniany opiekun projektu)
Nazwa dokumentu: Specyfikacja Wymagań Systemowych		Odpowiedzialny za dokument: (nazwisko, imię)	

Historia dokumentu				
Wersja	Opis modyfikacji	Rozdział / strona	Autor modyfikacji	Data
{wersja}	{opis, np. wstępna wersja}	{np. całość}	{nazwisko, imię}	{data zmiany}
{wersja}	{np. poprawka wstępnego opisu}	{np. punkty 3.3 i 4}	{nazwisko, imię}	{data zmiany}

Opis etapu: Celem etapu jest udokumentowanie procesu i podjętych decyzji związanych ze zbieraniem, przetwarzaniem i analizowaniem wymagań.

Oczekiwane produkty: Specyfikacja wymagań systemowych w postaci dokumentu o nazwie GRX(XYZ)-SWS-WER1.doc o strukturze według poniższego szablonu.

1 Wprowadzenie - o dokumencie

1.1 Cel dokumentu

{Określenie celu, jakiemu ma służyć ten dokument; tutaj: zdefiniowanie wymagań na podstawie analizy otoczenia projektu / przedsięwzięcia oraz analizy potrzeb klienta}

1.2 Zakres dokumentu

{Określenie, co zawiera dokument; tutaj: analiza otoczenia – klienci, kontekst biznesowy systemu, określenie udziałowców, zdefiniowanie wymagań}

1.3 Dokumenty powiązane

{Pełna lista opracowań poza zakresem dokumentu, lecz powiązanych merytorycznie z tym dokumentem; ujęte wersje poszczególnych dokumentów; także poprzednie wersje dokumentu SWS}

1.4 Odbiorcy

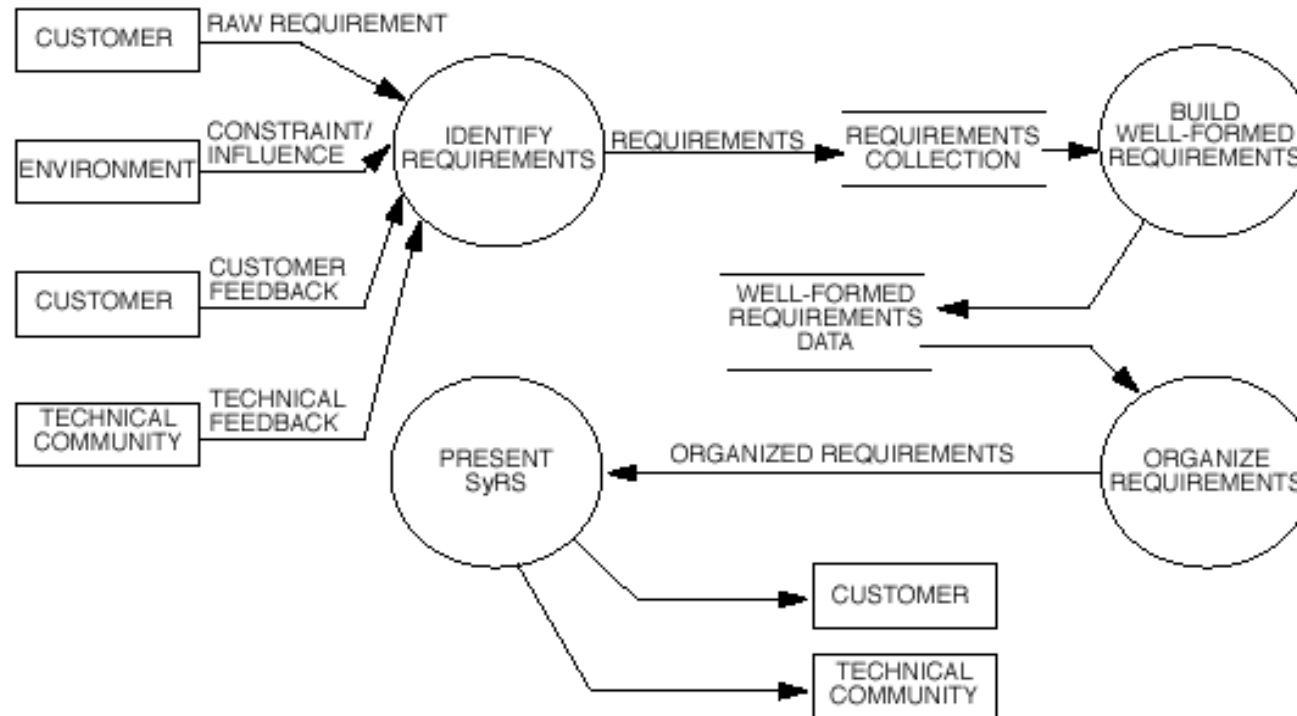
{Określenie adresatów dokumentu np. zleceniodawca, zleceniobiorca, członkowie zespołu projektowego oraz (jeśli znane) wymienione z nazwiska osoby, do których dokument ma dotrzeć; aspekty prawne i licencyjne (kto posiada prawa autorskie do dokumentu / jaki akt prawny je reguluje?)}

1.5 Słownik pojęć

{Wyjaśnienie używanych w dokumencie oznaczeń, pojęć i skrótów np. akronimy, nazwy skrótowe, itp; definicje są konieczne, aby wszyscy czytelnicy mogli zrozumieć treść dokumentu; ważne jest umieszczenie na tej liście pojęć o charakterze technicznym oraz pojęć dziedziny merytorycznej zastosowania aplikacji. Nie jest konieczne wyjaśnianie słów używanych powszechnie}

2 Projekt w kontekście

Proces tworzenia dokumentu SWS (*ang. SyRS*) według Standardu IEEE 1233



Identyfikacja i specyfikacja wymagań w podejściu *agile* (SCRUM)

Zasadniczy wynik:

- ✓ rejestr produktowy (rejestr zaległości produktu)
- ✓ opis zasad (reguł) biznesowych
- ✓ zbiór scenariuszy użytkowych (historyjek)
- ✓ słownik pojęć
- ? wsparcie notacjami modelującymi (przypadki użycia)



- ✓ historyjki związane z kolejnymi produktami (przyrostami)
- ✓ PJATK: przypadki użycia

Analiza agilowa (SCRUM)

Budowa i integracja SI

↓ **Utworzenie rejestru (backlogu) produktowego**
▪ funkcjonalności / scenariusze / historyjki

↓ **Zasady biznesowe** →

↓ **Planowanie sprintu:** →
każdorazowe określenie historyjek
do dostarczenia w jego ramach

Artefakty SCRUMa

➤ Rejestr zaległości produktowych
(rejestr produktowy, ang. *Product Backlog*)
i historyjki (scenariusze, *user stories*)

Na przykład, dla zamierzonego portalu mogłoby to być:

- Profiles**
 - As a site member, I want to describe myself on my own page in a semi-structured way. That is, I can fill in predefined fields, but also have room for a free-text field or two. (It would be nice to let this free text be HTML or similar.)
 - As a site member, I can fill out an application to become a Practitioner.
-
- News**
 - As a site visitor, I can read current news on the home page.
 - As a site visitor, I can access old news that is no longer on the home page.
-
- Courses and Events**
 - As a site visitor, I can see a list of all upcoming "Certification Courses." I can page through them if there are a lot.
 - As a site visitor, I can see a list of all upcoming "Other Courses" (non-certification courses). I can page through them if necessary.
-
- FAQs**
 - As a site visitor, I can read FAQs.
 - As a site editor, I can maintain an FAQ section.
-
- Resource**
 - As a site member, I can download the latest training material and methodology PDFs.
 - As a visitor, I can download presentations, PDFs, etc. on Scrum that I can use.
- Jobs**

.....

*Priorytety! Także konstrukcyjne
→ Inicjalna lista sprintów!*

Articles
Home Page
Ratings

<http://www.mountaingoatsoftware.com/agile/scrum/product-backlog/example>

Kto to formułuje?



**Klient – Product Owner
– Deweloperzy**

Analiza agilowa (SCRUM)

Zasady biznesowe

6.2 Zasady biznesowe

6.2.1 BR-01

Wszystkie hasła do kont użytkowników muszą być przechowywane w postaci zahasowanej.

6.2.2 BR-02

Wszystkie wpisywane hasła do kont użytkowników muszą być wyświetlane jako ciągi kropek.

6.2.3 BR-03

Wylogowanie z systemu jest możliwe z każdego ekranu aplikacji (poza ekranem logowania).

6.2.4 BR-04

Przy każdym ustawianiu nowego hasła, konieczne jest wpisanie go dwukrotnie.

6.2.5 BR-05

Dana nazwa użytkownika może być przypisana tylko do jednego konta.

6.2.6 BR-06

Dany adres email może być przypisany w danej chwili tylko do jednego konta.

6.2.7 BR-07

Prowadzący może zatrzymywać, uruchamiać ponownie i usuwać wyłącznie te instancje, które zostały przez niego utworzone. Administrator może zatrzymywać, uruchamiać ponownie i usuwać dowolne instancje.

*Ł. Augustynowicz, R. Gruberski,
Ł. Klonowski : Suezmax - Uczelniana
platforma lekkiej wirtualizacji.*

Analiza agilowa (SCRUM): Historyjki (scenariusze)

Identyfikator	US-01
Treść	Jako Administrator Systemu, chcę stworzyć pierwsze konto użytkownika Suezmax w nowo zainstalowanym systemie, aby móc zalogować się do Suezmax.
Powiązane zasady biznesowe	BR-01: hasła do kont użytkowników muszą być przechowywane w postaci zahasowanej
Kryteria akceptacji	1. Można zalogować się na pierwsze konto.

Identyfikator	US-02
Treść	Jako użytkownik Suezmax, chcę zalogować się na swoje konto, aby pracować z systemem.
Powiązane zasady biznesowe	BR-02: wpisywane hasła do kont muszą być wyświetlane jako ciągi kropek
Kryteria akceptacji	1. Po wpisaniu nazwy użytkownika oraz pasującego do niej hasła, użytkownik zostaje zalogowany.

....

Identyfikator	NF-03
Treść	Instancje wirtualne z przekroczonym terminem ważności są automatycznie usuwane w celu zaoszczędzenia zasobów oraz ułatwienia administracji systemu.
Powiązane zasady biznesowe	
Kryteria akceptacji	1. Najpóźniej w ciągu godziny po przekroczeniu daty ważności, instancja zostaje usunięta.

użytkowe

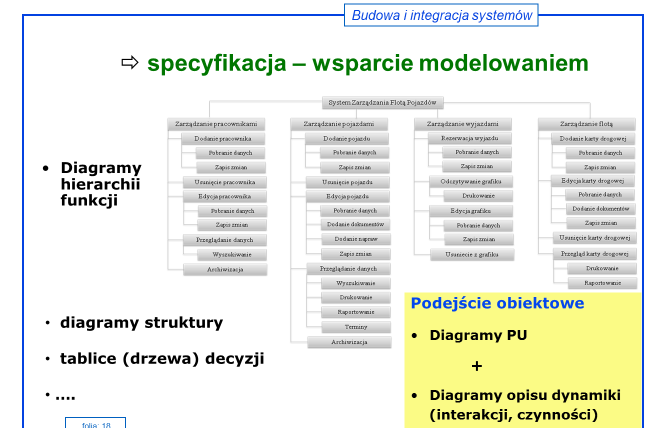
niefunkcjonalne

*Ł. Augustynowicz, R. Gruberski, Ł. Klonowski : Suezmax
- Uczelniana platforma lekkiej wirtualizacji.*

⇒ **specyfikacja – wsparcie modelowaniem**

Przypadki użycia

- ↓ **Specyfikacja aktorów**
 - ↓ **Diagram (diagramy) PU**
 - *biznesowe vs. systemu informatycznego*
 - ↓ **Ustrukturalizowany opis (specyfikacja) poszczególnych PU**
 - ↓ **Zachowanie – przebieg wykonania najważniejszych PU (lub ich sekwencji), udział aktorów i wyróżnionych obiektów systemu**
 - diagramy sekwencji, diagramy komunikacji (kolaboracji)
 - diagramy czynności



Przypadki użycia – przykład specyfikacji aktorów w systemie SODL







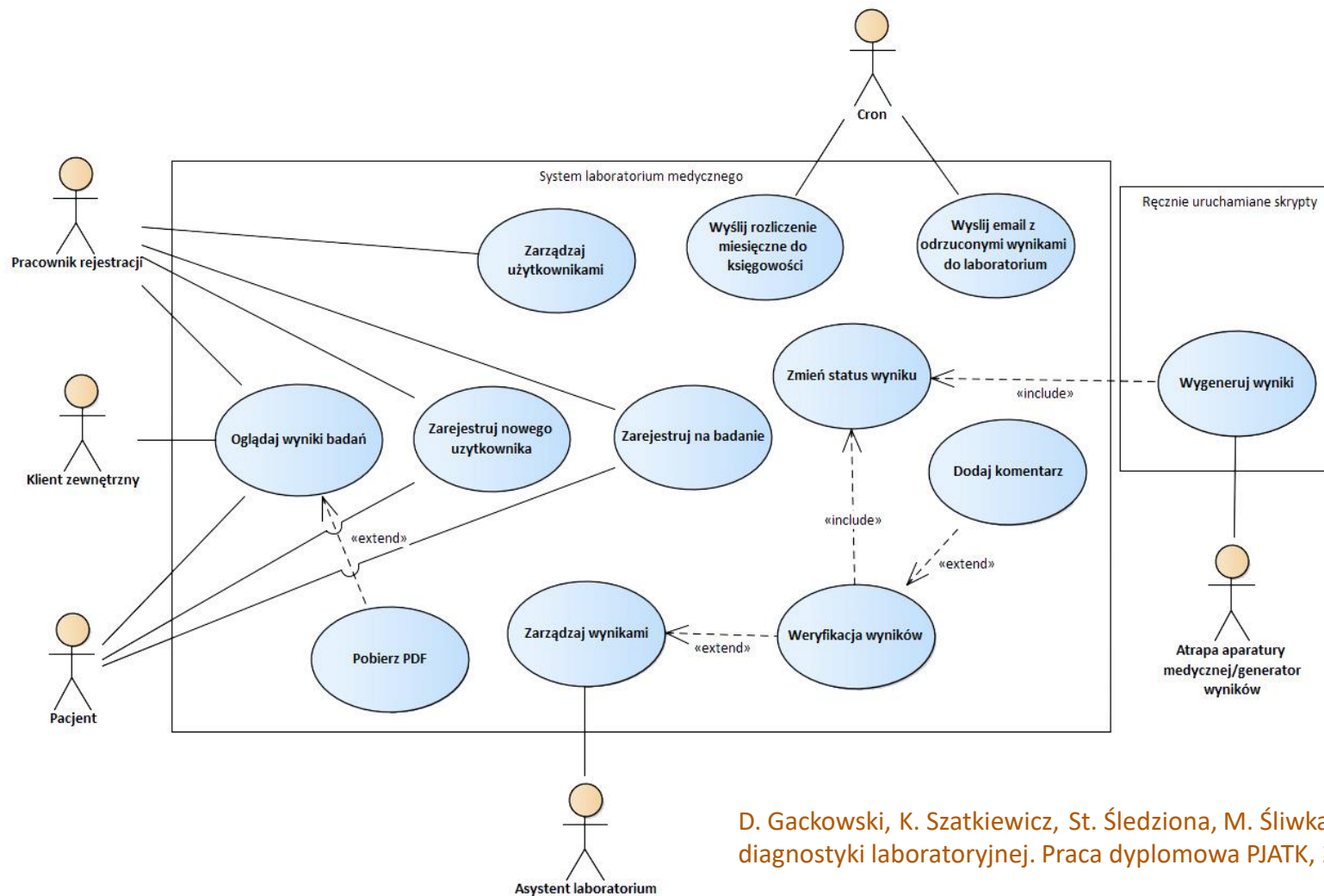
	Pracownik rejestracji – aktywny/główny – osoba upoważniona do rejestrowania pacjentów w ich imieniu, zapisywania ich na badanie i podglądu wyników badań w celu ich wydruku na życzenie pacjenta.
	Pacjent – aktywny/główny – Osoba chcąca wykonać badanie, rejestrująca się za pośrednictwem Internetu lub osobiście w placówce. Po zarejestrowaniu uzyskuje możliwość zapisania się na badanie i podgląd wyników.
	Klient zewnętrzny – aktywny/drugorzędny – Inna placówka medyczna mająca dostęp do podglądu zleconych przez nią badań.
	Asystent laboratorium – aktywny/główny – pracownik placówki medycznej, ma uprawnienia do podglądu wyników badań generowanych przez aparaturę medyczną i weryfikacji ich poprawności.
	Atrapa aparatury medycznej/generator wyników – pasywny/drugorzędny – Kod imitujący wyniki pracy aparatury medycznej. Zapisuje wygenerowane dane do bazy przy pomocy kolejki asynchronicznej.
	Cron – pasywny/drugorzędny – Program do harmonogramowania zadań odpowiedzialny za wysyłanie maili do laboratorium z odrzuconymi wynikami oraz do księgowości z miesięcznym spisem faktur do wystawienia.

Diagram przypadków użycia – przykład SODL



D. Gackowski, K. Szatkiewicz, St. Śledziona, M. Śliwka: System obsługi diagnostyki laboratoryjnej. Praca dyplomowa PJATK, 2017

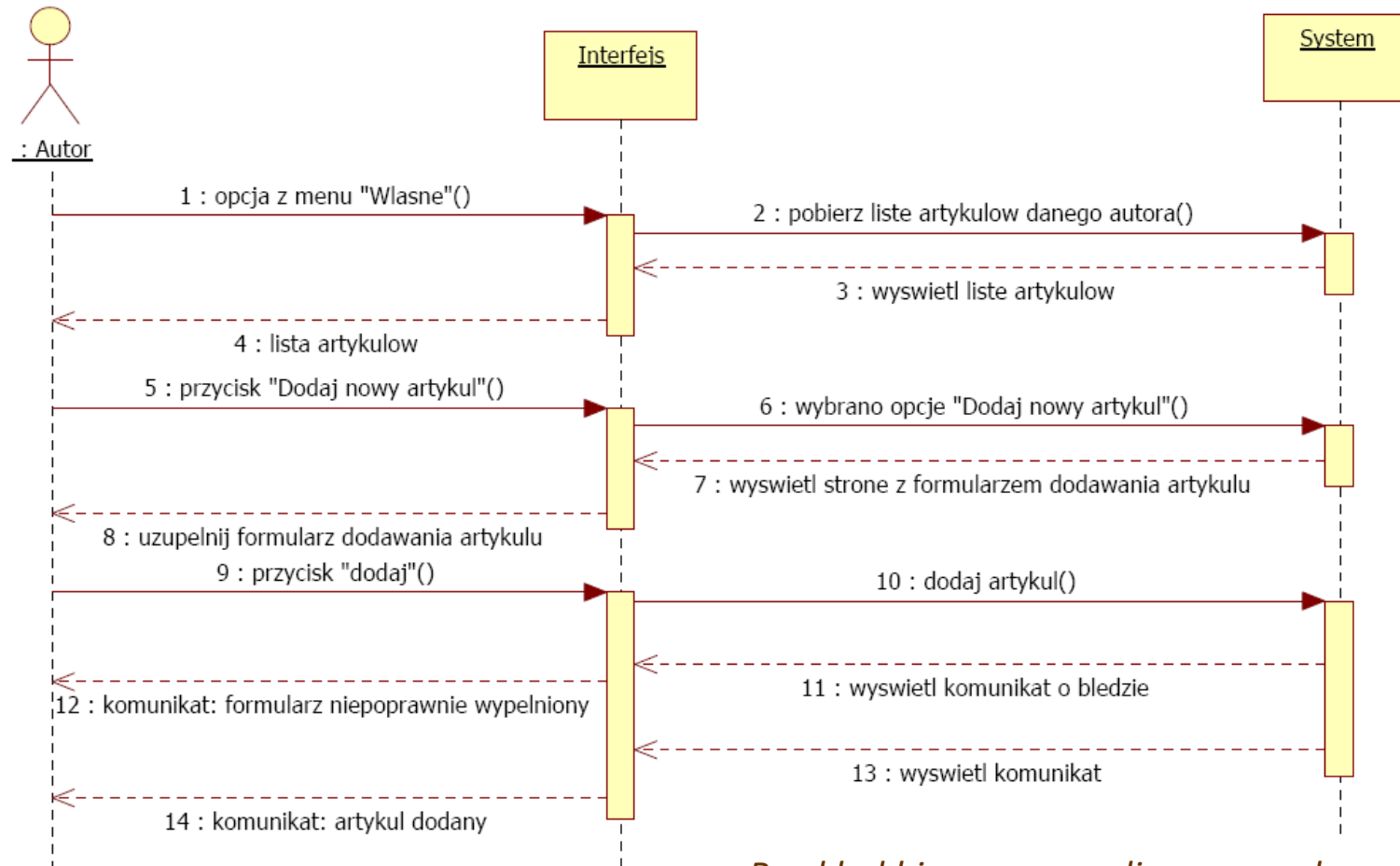
Ustrukturalizowany opis przypadków użycia

M. Kubiś: Projekt i implementacja systemu rozszerzonej analizy wymagań

Autoryzacja

Aktorzy	Kierownik projektu
Streszczenie	Proces logowania do systemu.
Zdarzenie inicjujące	Rozpoczęcie korzystania z systemu
Struktura	Brak
Warunek wstępny	W systemie nie ma zalogowanego użytkownika (wyjątek: W systemie jest zalogowany użytkownik)
Opis	Wpisanie nazwy użytkownika, wpisanie hasła, weryfikacja danych (nazwy użytkownika i hasła)
Wyjątki	W systemie jest zalogowany użytkownik: wyloguj dotychczasowego użytkownika
Warunek końcowy	Użytkownik zalogowany w systemie
Komentarz	Brak

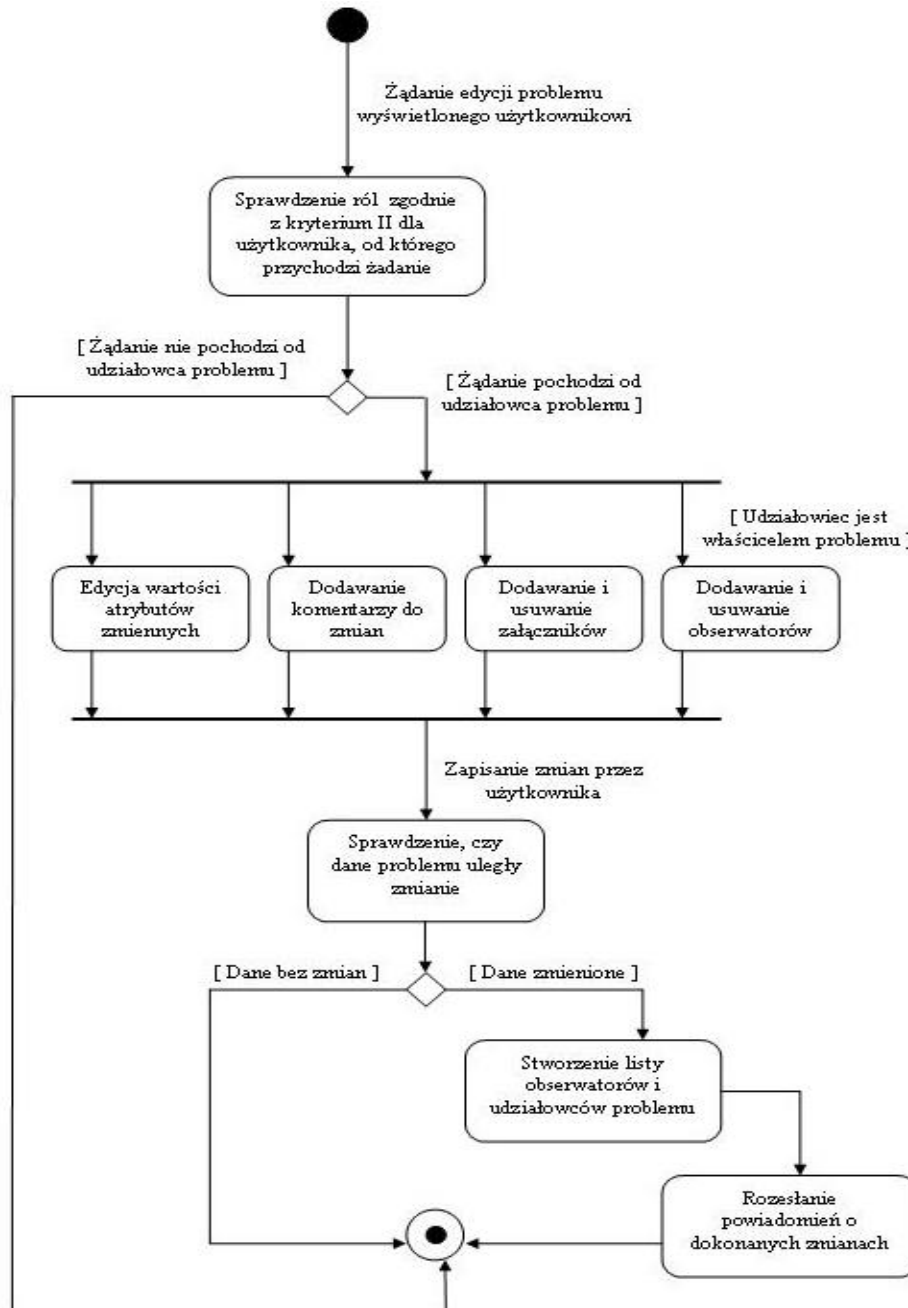
Opis zachowania przy realizacji głównych PU – diagramy interakcji (sekwencji lub komunikacji)



*Przykład biznesowego diagramu sekwencji
PU Dodawanie artykułu [K. Brzeziński]*

Opis zachowania przy realizacji głównych PU – diagramy czynności

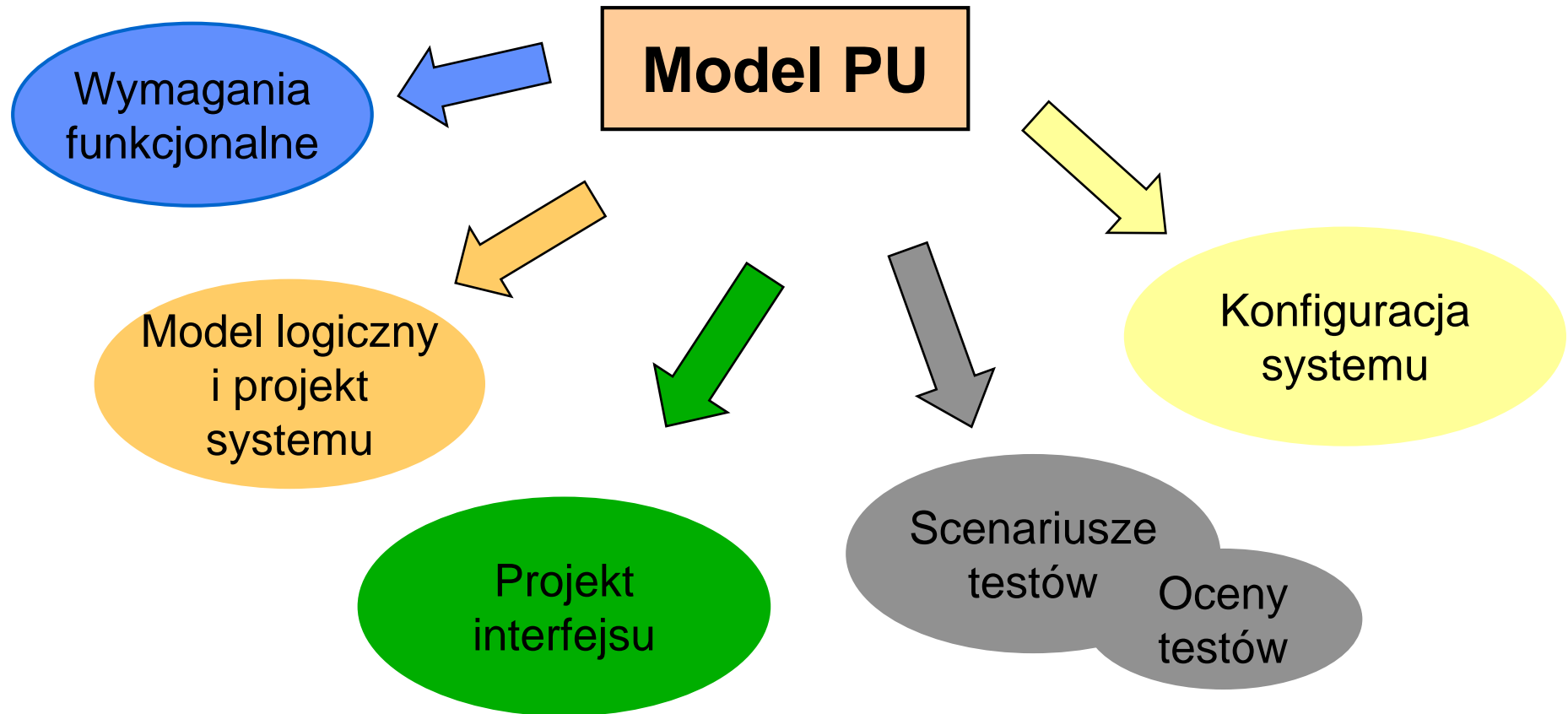
T. Łutowicz: System monitorowania problemów procesu dydaktycznego




Edytowanie
problemu

**Edycja istniejącego (wyświetlonego)
problemu**

Kontekst użycia modelu przypadków użycia



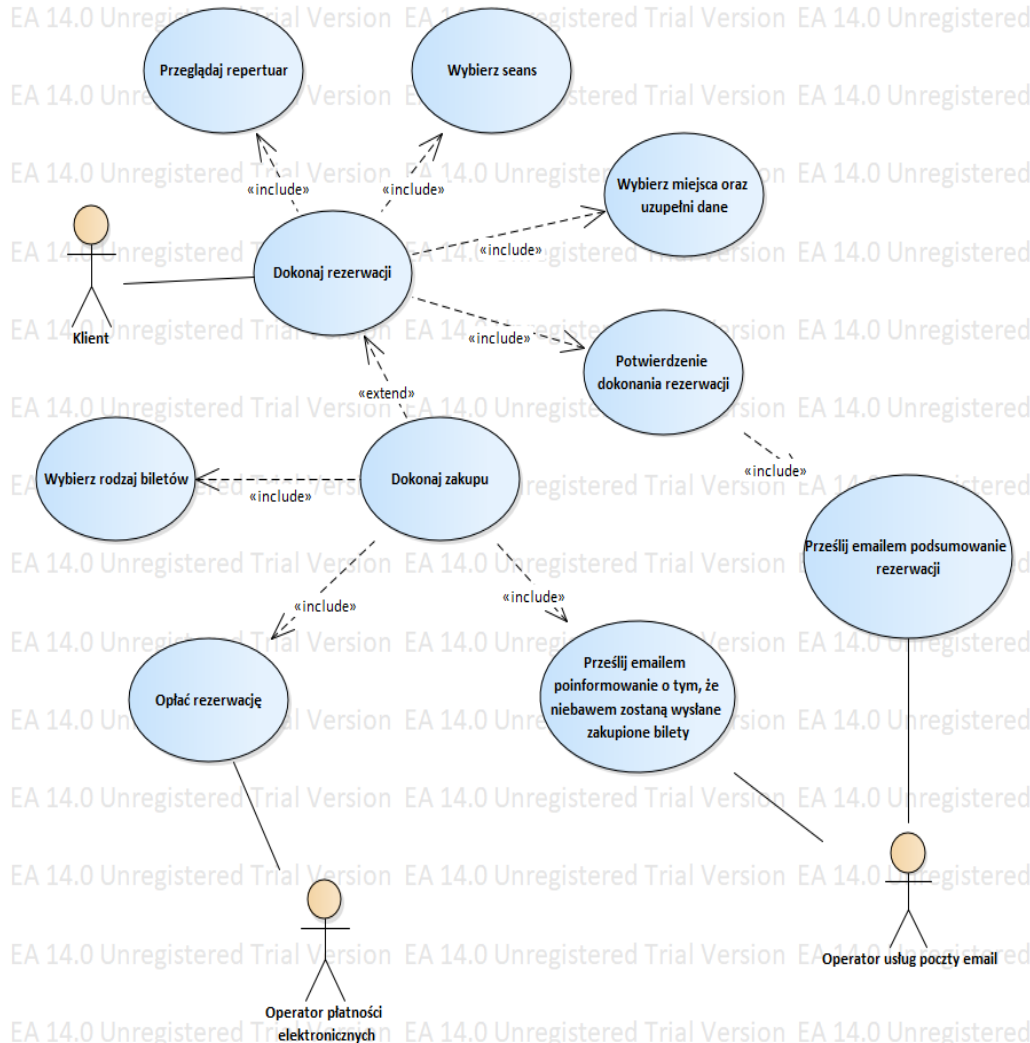
Dlaczego modelowanie przypadków użycia jest ważne

- ⇒ Definiują zakres systemu, jego granicę
- ⇒ Określają użytkowe funkcjonalności systemu
 - powinny też definiować reakcję na sytuacje wyjątkowe
- ⇒ Ukazują otoczenie systemu i interakcje między nimi
- ⇒ „Są podobne” do wymagań, ale dają znacznie lepszy przekaz, są mniej podatne na sprzeczności, nadmiarowości, niespójności,
- ⇒ Są zrozumiałe i komunikatywne
- ⇒ Pozwalają usprawnić komunikację klient – analityk – deweloper
- ⇒ Przydatne dla celów prezentacji i dokumentacji
- ⇒ Przydatne w różnych fazach cyklu 

Wykorzystanie przypadków użycia w cyklu wytwarzania oprogramowania

- **specyfikacja wymagań funkcjonalnych wobec systemu**
- **weryfikacja poprawności i kompletności specyfikacji wymagań**
- **wspomaganie walidacji wymagań**
- **identyfikacja składowych, obiektów systemu**
- **podstawa do opracowania diagramów interakcji**
- **punkt wyjścia do opracowania testów (w tym akceptacyjnych) systemu**
- **możliwość śledzenia, jak wymagania przeradzają się w klasy obiektów i operacje systemu**
- **projekt logiczny interfejsu, projekt nawigacji, kształt i zawartość interakcji**
- **podstawa do oceny rezultatów testów**
- **konfigurowanie składników systemu i ich dostępności**

Problem granulacji PU

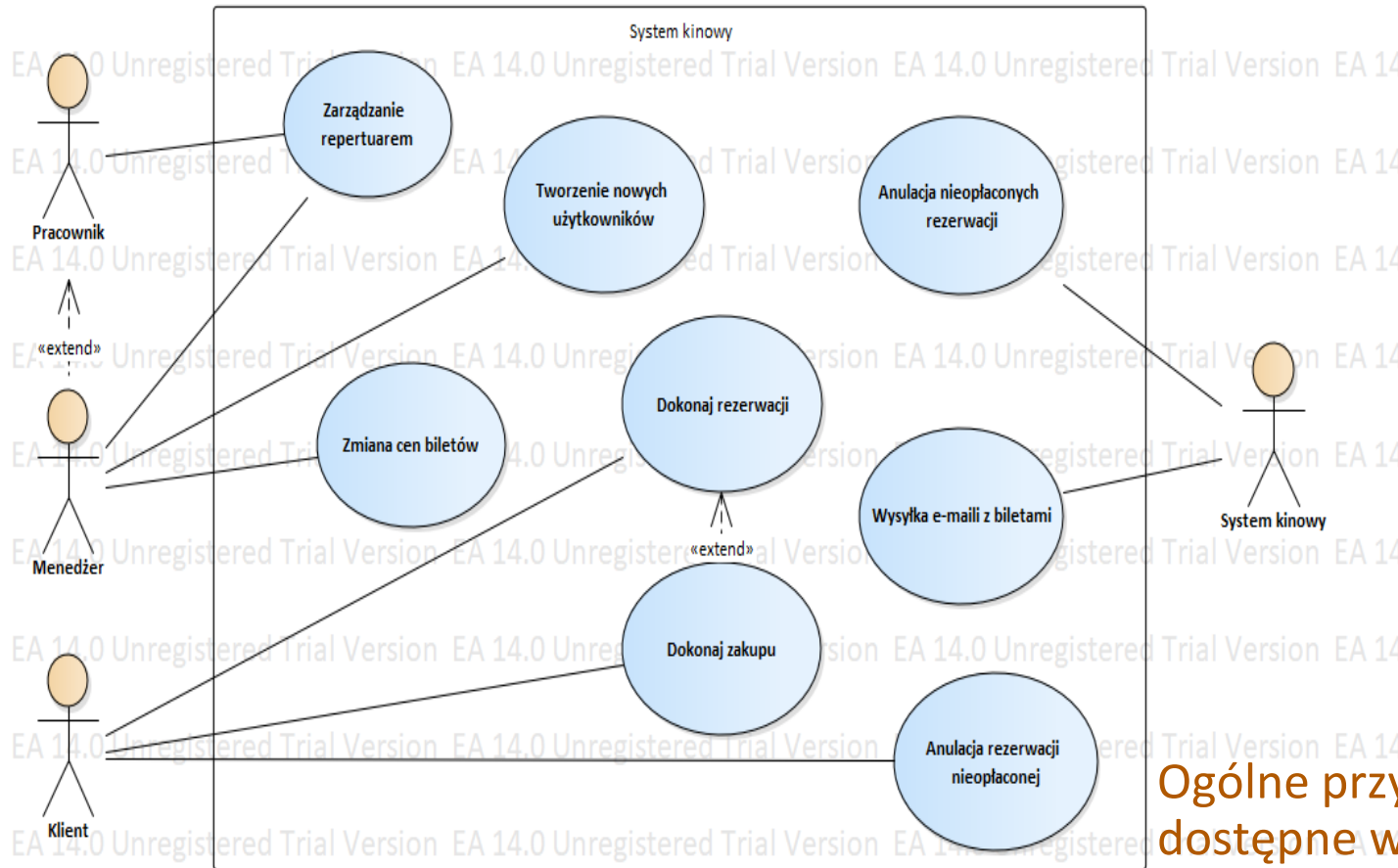


Czy (kiedy) przypadek użycia powinien reprezentować każdą interakcję użytkownika z systemem, a kiedy koncentrować się na głównym zadaniu aktora?

Całościowy przypadek użycia związany z *Rezerwacją biletów*

M. Sielawa, M. Niewiarowski, M. Groth,
S. Zblewski: SUPREME - System wspomagający
pracę kina. Praca dypl. PJATK 2018

Problem granulacji PU



Ogólne przypadki użycia
dostępne w systemie

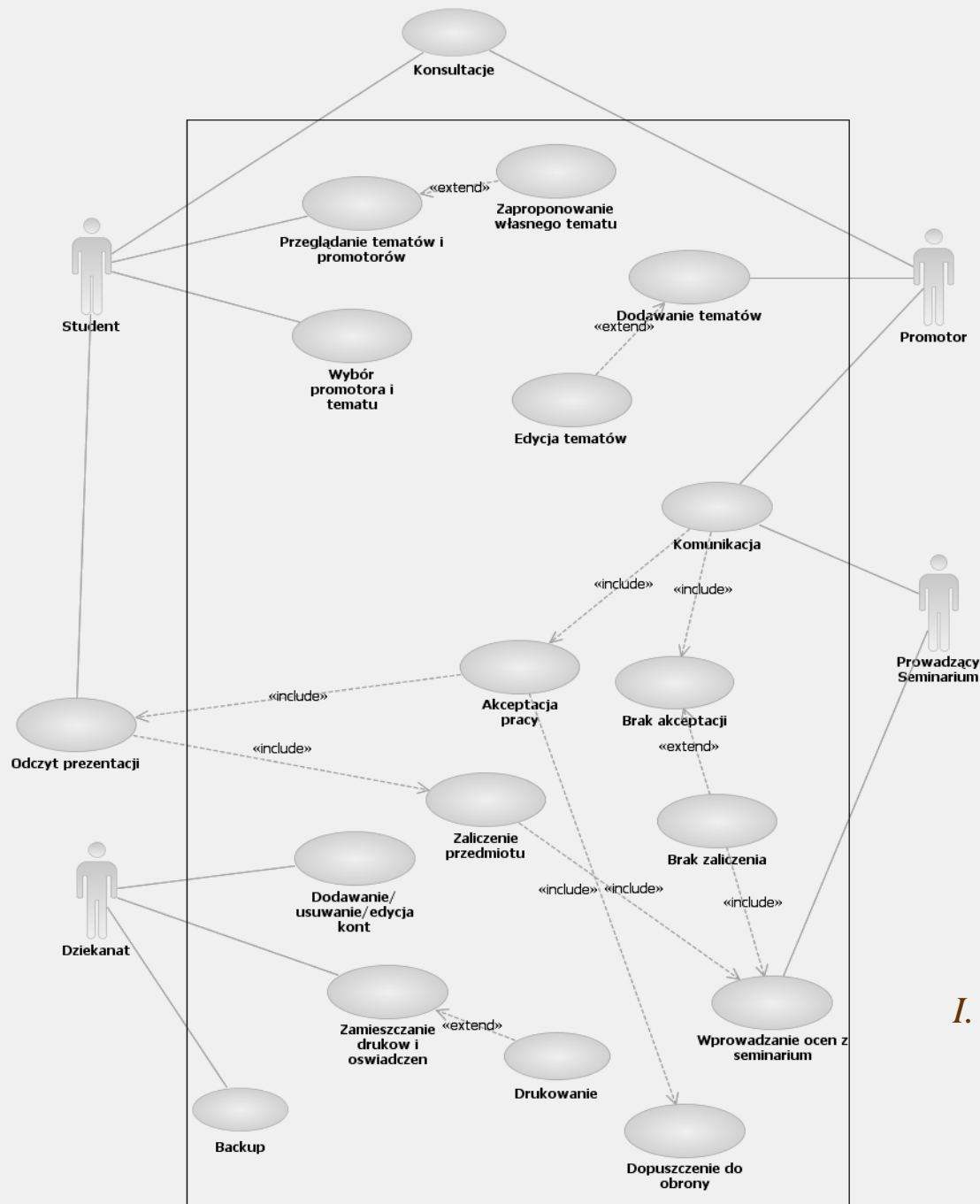
M. Sielawa , M. Niewiarowski, M. Groth,
S. Zblewski: SUPREME - System wspomagający
pracę kina. Praca dypl. PJATK 2018

Co zrobić gdy diagram przypadków użycia „wychodzi” mocno rozbudowany i mało czytelny?

- **Tworzyć diagramy hierarchiczne**
np. na diagramie ogólnym umieścić PU *Administrowanie systemem*, a na kolejnych rozwinąć ten PU o przypisane mu (include, extends) PU szczegółowe, jak *Administrowanie użytkownikami*, *Zarządzanie uprawnieniami*, *Edycja itp.*
- **Zweryfikować możliwość uogólnienia, związki, np. pod kątem dziedziczenia funkcji przez aktorów**
- **Tworzyć diagramy dla poszczególnych podsystemów logicznych**
Wówczas nieujęte na diagramie podsystemy modelujemy jako aktorów!
- **Tworzyć diagramy dla poszczególnych aktorów**
np. odrębnie dla zwykłego użytkownika, odrębnie dla administratora,...

Co zrobić gdy w analizie pojawiają się zadania spoza zakresu? albo nienależące do aktualnie rozpatrywanego wydania?

- Można przedstawić je jako podsystemy (logiczne) reprezentowane przez aktorów
- Rysując pewne PU poza ramką określającą zakres systemu *przykład* →
- Wyodrębniamy – np. cieniujemy (kolorujemy) na diagramie przypadki użycia nienależące (albo należące) do zakresu *przykład* →



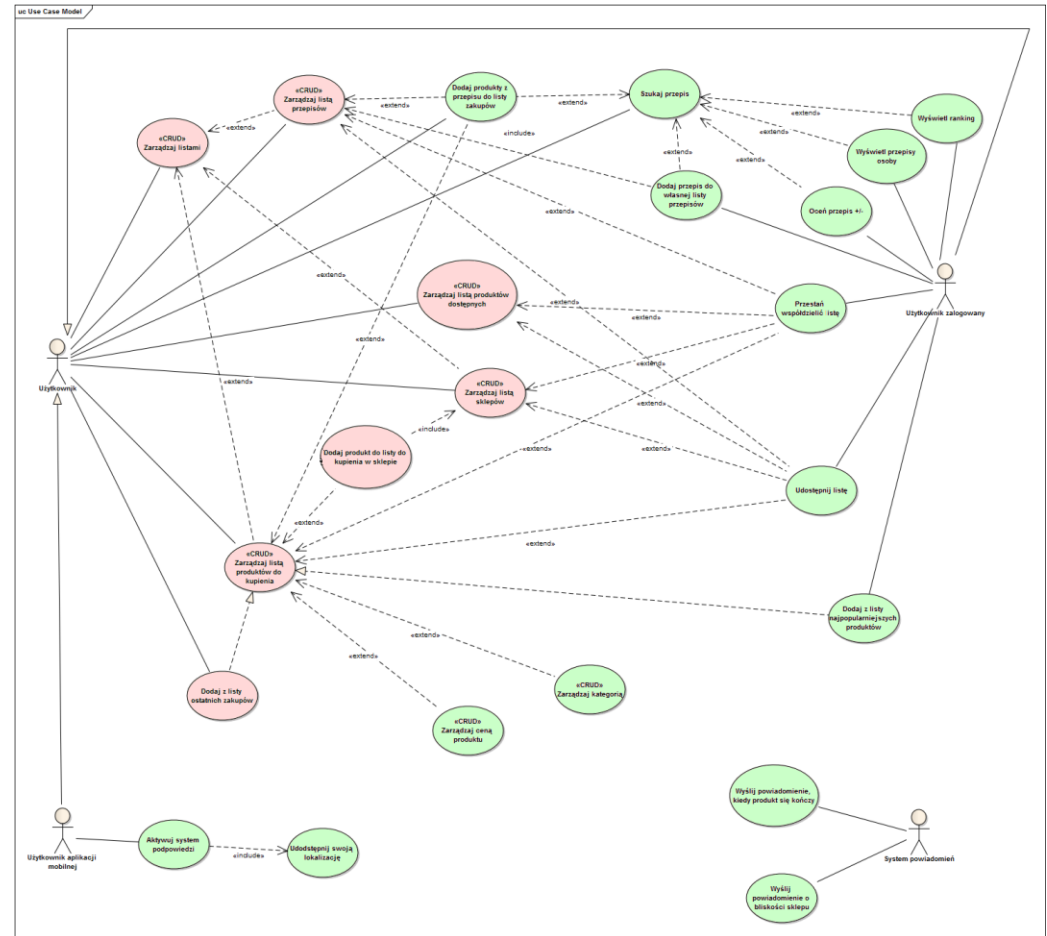
Przypadki użycia – modelowanie przypadków pozostających poza systemem

I. Derengowska: System monitorowania przebiegu realizacji prac dyplomowych

Przypadki użycia – wyodrębnianie

Nadawanie wag lub priorytetów

- **w celu ujęcia całościowego kontekstu**
- **dla wyróżnienia wycinka pracy lub podsystemu**
- **dla potrzeb realizacyjnych**
 - **wskazanie kolejności**
 - **wydaniami**



RYSUNEK 3 - DIAGRAM PRZYPADKÓW UŻYCIA (KOLOR CZERWONY – I WYDANIE, KOLOR ZIELONY – II WYDANIE)

A. Jackiewicz, A.Twórz, B. Nowaczyk, P. Jabłoński: Projekt i realizacja interaktywnego asystenta zakupów – Concierge. Praca dypl. ZWI PJATK, 2015

⇒ **walidacja wymagań (SWS, stories, PU)**

- **praca z udziałowcami, przede wszystkim – z Klientem (Zamawiającym, Użytkownikiem)**



Pomocna technika: specyfikacje formalne

Przykład

specyfikacji formalnej operacji wyszukiwania położenia w tablicy a elementu e

operation Wyszukaj (a : Array_of _Elem; $int1$, $int2$: Integer; e : Elem)

returns Integer

Pre: exists i in $int1..int2$ that $a(i) = e$

Post: $a' = a$ and a' (Wyszukaj (a , $int1$, $int2$, e)) = e

Error: Wyszukaj (a , $int1$, $int2$, e) < $int1$
or Wyszukaj (a , $int1$, $int2$, e) > $int2$

Zalety

- Formalne określenie, jednoznaczność
- Możliwość automatycznego badania właściwości
- --”– weryfikacji niesprzeczności, kompletności,...

Przykład specyfikacji ADT (abstrakcyjnego typu danych)

Sygnatura *Stosu*

sorts *stack (elem)*

opns

empty: --> stack

push: elem, stack --> stack

top : stack --> elem

axioms

top (empty) --> **undefined**

top (push (elem, stack)) = elem

push (elem, stack) = stack

end sorts

⇒ Pielęgnacja wymagań

Pielęgnacja –

usuwanie usterek

aktualizacja

zarządzanie zmianami

i kontrola ich wprowadzenia

➤ Konieczne jest posiadanie **procedury zarządzania zmianami wymagań**

- dokumentowania propozycji zmian
- analizowania i oceny ich akceptowalności, zgodności z celami, spójności i niesprzeczności z innymi;
- szacowania kosztu ich wprowadzenia

oba powyższe punkty także świetle możliwości zrealizowania też innych wymagań

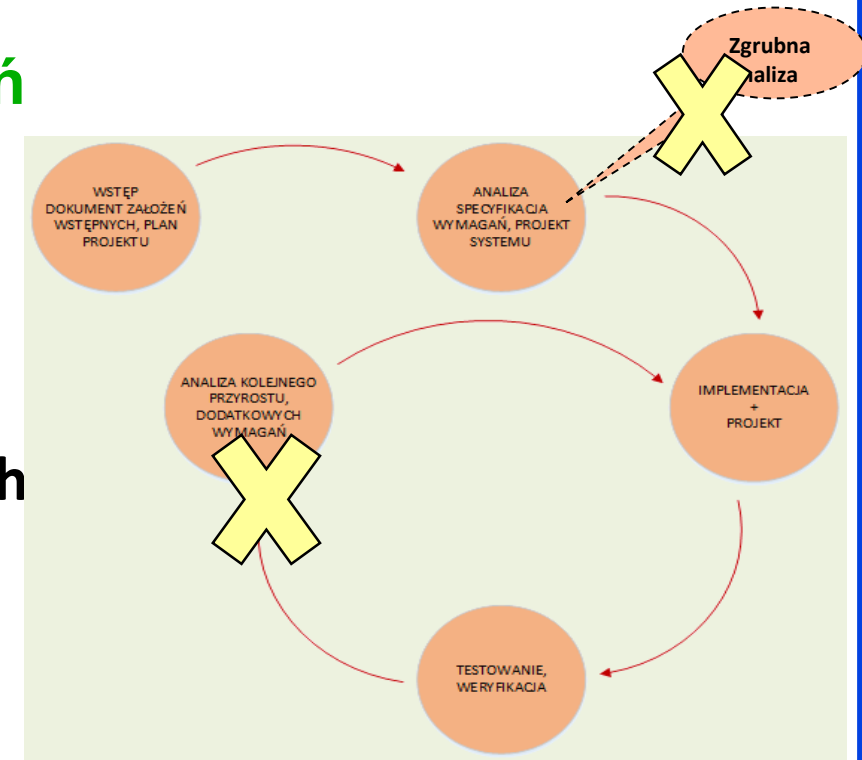
- formalnego zaakceptowania i wprowadzenia do SWS
- odpowiedzialności za powyższe

Zmienność jest immanentną cechą wymagań!

Przy pielęgnacji wymagań ważne jest zachowanie spójności wszystkich reprezentacji

❑ Jak zrobić specyfikację wymagań gdy w projekcie planujemy realizację przyrostową

- Zgrubna „standardowa” + specyfikacje wymagań w przyrostach
- Całościowa „standardowa”
 - SWS + przyp. użycia + model logiczny
- Specyfikacja agile’owa
 - Wersje jak wyżej
 - Uwaga: w PJATK domagamy się modelowania przypadków użycia



Projekt PRO

Analiza agilowa (SCRUM) scenariusze (historijki) definiowane w sprintach

➤ Najłatwiej – w powiązaniu z kolejnymi przyrostami odpowiadającymi sprintom realizującym produkty rejestru zaległości (ang. *Product Backlog*)

Profiles

•As a site member, I want to describe myself on my own page in a semi-structured way. That is, I can fill in predefined fields, but also have room for a free-text field or two. (It would be nice to let this free text be HTML or similar.)

•.....

News

•As a site visitor, I can read current news on the home page.

•As a site visitor, I can access old news that is no longer on the home page.

•.....

Courses and Events

•As a site visitor, I can see a list of all upcoming "Certification Courses." I can page through them if there are a lot.

•As a site visitor, I can see a list of all upcoming "Other Courses" (non-certification courses). I can page through them if necessary.

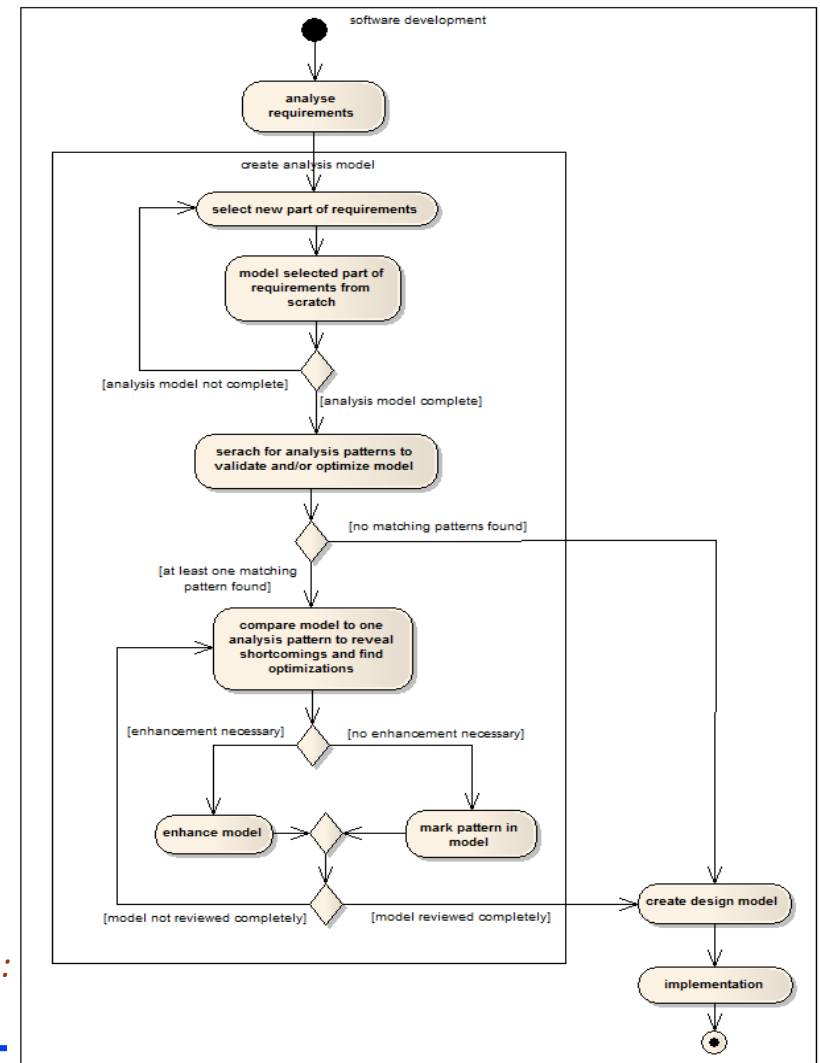
•.....

<http://www.mountaingoatsoftware.com/agile/scrum/product-backlog/example>

⇒ Weryfikacja i walidacja wymagań

Weryfikacja i walidacja wymagań - jak?

- *weryfikacja – także modeli i ich zgodności z reprezentacją tekstową*
- *walidacja logiczna*
- *walidacja przez klienta*
- *przydatność metod specyfikacji formalnej*
- *przydatność wzorców analizy*



Blaimer N., Bortfeldt A., Pankratz G.:
Patterns in Object Oriented Analysis