

## Wprowadzenie do inżynierii wymagań

dr hab. inż. Michał Śmiałek, prof. uczelni  
dr inż. Kamil Rybiński



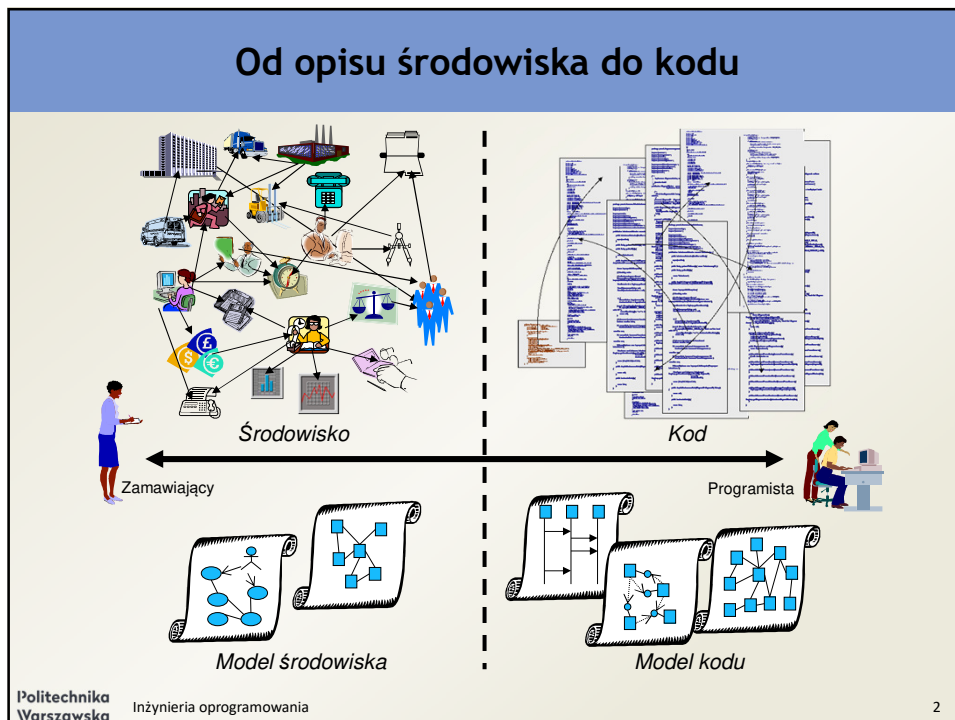
Wydział  
Elektryczny  
POLITECHNIKA WARSZAWSKA

Inżynieria oprogramowania



1

### Od opisu środowiska do kodu

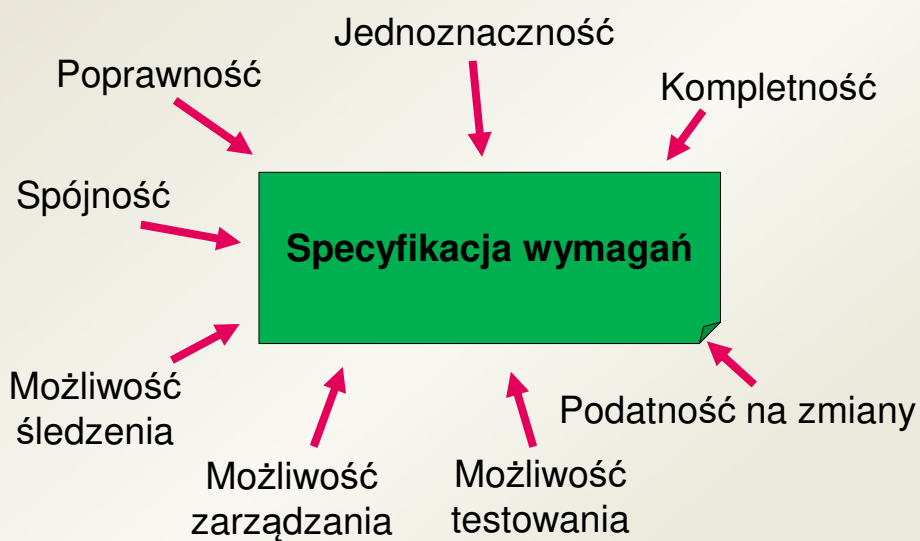


2

## Czym jest wymaganie?

- Wymaganie to własność produktu końcowego (systemu oprogramowania), którą musi on posiadać, aby spełnić oczekiwania zamawiającego
  - Sposób funkcjonowania systemu
  - Cecha jakościowa narzucona na system
- System spełnia wymagania jeśli zostanie potwierdzone spełnienie wymaganych własności
- Inżynieria wymagań
  - Dyscyplina polegająca na zbieraniu, analizowaniu, negocjowaniu i specyfikowaniu wymagań
  - Główny produkt to specyfikacja wymagań
- Jakość wymagań jest kluczowa dla powodzenia projektu

## Cechy dobrej specyfikacji wymagań



## Cechy dobrej specyfikacji wymagań (2)

- **Kompletność**
  - Obejmuje cały zakres potrzeb zamawiającego, opisany w niezbędnych szczegółach
  - Uwzględnia wszystkie obszary wymagań
- **Jednoznaczność i poprawność**
  - Definiuje jeden możliwy zakres systemu
  - Nie pozostawia pola do różnych interpretacji
- **Spójność**
  - Brak sprzeczności między różnymi wymaganiami
  - Używa wspólnego słownika pojęć

## Cechy dobrej specyfikacji wymagań (3)

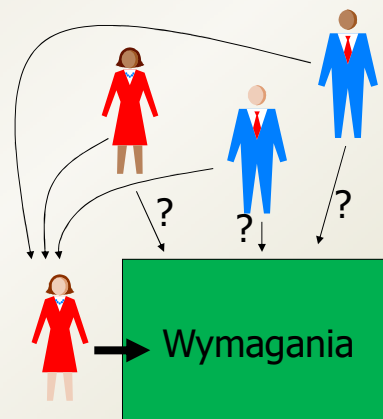
- **Możliwość zarządzania**
  - Podzielona na dobrze określone jednostki (wymagania)
  - Stertuje procesem wytwarzania systemu
  - Wymagania posiadają atrybuty
  - Umożliwia śledzenie produktów wynikających z wymagań
  - Umożliwia śledzenia zmian wymagań
- **Testowalność**
  - Możliwość „zmierzenia” spełnienia wymagania
  - Możliwość utworzenia scenariuszy testów funkcjonalnych
  - Możliwość opracowania metryk dla cech jakościowych
- **Uwaga: dobrej jakości wymagania = zadowolony klient pod koniec projektu**

## Źródła wymagań

- Istniejące procesy
  - Opisy procesów biznesowych obowiązujące w organizacji
- Docelowe procesy
  - Specyfikacja zmodyfikowanych procesów biznesowych, biorących m.in. pod uwagę zastosowanie nowego systemu
- Podręczniki użytkownika
  - Specyfikacja opisująca funkcjonalność istniejących systemów
- Zamawiający
  - Osoby decydujące o dokonaniu zmian w organizacji i budowie nowego (lub rozbudowanego) systemu
- Użytkownicy
  - Osoby, które będą bezpośrednio pracować z systemem

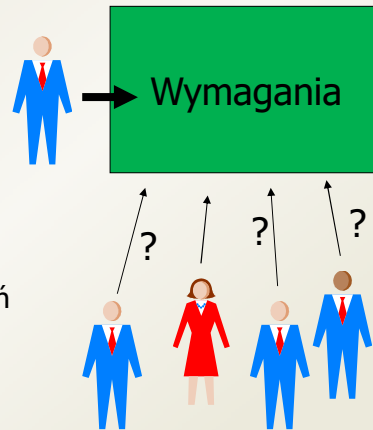
## Odbiorcy wymagań - zamawiający

- Sponsorzy
  - Podejmują decyzję o potrzebie zamówienia systemu
- Klienci/dostawcy (współpracownicy) zamawiającego
  - Będą musieli współpracować z systemem
- Przygotowujący przetargi
  - Muszą zorganizować przetarg na budowę systemu zgodnego z wymaganiami
- Użytkownicy końcowi
  - Będą bezpośrednio korzystać z systemu



## Odbiorcy wymagań - wykonawcy

- Przygotowujący oferty
  - Muszą złożyć ofertę na budowę systemu zgodnego z wymaganiami
- Projektanci systemu
  - Muszą zaprojektować i zrealizować zadany system
- Testerzy
  - Muszą napisać testy, które zweryfikują spełnienie wymagań
- Kierownicy projektów
  - Muszą pokierować realizacją systemu tak, aby był zgodny z wymaganiami



## Zarządzanie wymaganiami - co to jest?

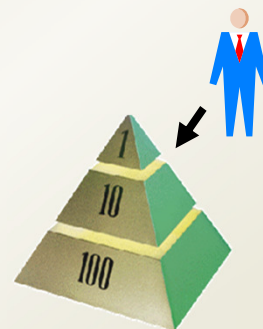
Zarządzanie wymaganiami polega na ciągłym upewnianiu się, że

- rozwiązujemy właściwy problem
- budujemy właściwy system

poprzez systematyczne podejście do

- zbierania
- organizowania
- dokumentowania
- wykorzystywania

zmieniających się wymagań na system oprogramowania.



## Zarządzanie wymaganiami - po co?

- Wymagania nie są oczywiste i mogą pochodzić z wielu źródeł
- Wymagania są różnego rodzaju i mają różny poziom szczegółowości
- Wymagań jest bardzo dużo i trudno je wszystkie objąć
- Wymagania zależą od siebie nawzajem i od innych produktów procesu dla oprogramowania
- Wymagania mają różne właściwości, np. znaczenie dla systemu
- Wymagania muszą pogodzić różne zainteresowane strony
- Wymagania się zmieniają (niestety!).

Politechnika  
Warszawska      Inżynieria oprogramowania

11

- |                            |                           |    |
|----------------------------|---------------------------|----|
| Politechnika<br>Warszawska | Inżynieria oprogramowania | 11 |
|----------------------------|---------------------------|----|

# Modelowanie środowiska systemu oprogramowania

- Opisujemy fragment otaczającego nas świata
- Setki lub tysiące różnych elementów oddziałujących na siebie w złożony sposób
- System informatyczny jest naturalnym składnikiem większości współczesnych organizacji
- Opis biznesu powinien wskazywać na możliwości usprawnienia pracy przy pomocy systemu

Politechnika  
Warszawska

Inżynieria oprogramowania

12

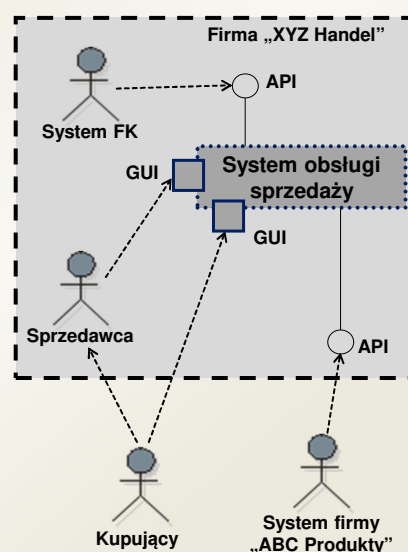
-

## Co modelujemy? Biznes i systemy informatyczne

- Podstawowe elementy organizacji biznesowej:
  - Współpracownicy, pracownicy, jednostki organizacyjne
  - Produkty, surowce, podzespoły, dokumenty
  - Systemy informatyczne
  - Procesy biznesowe
- Model biznesu powinien być „miniaturową kopią” rzeczywistego biznesu. System informatyczny jest jedynie elementem składowym biznesu.
- Model może dotyczyć stanu obecnego, lub stanu postulowanego (po dokonaniu zmian organizacyjnych). Zmiany mogą wynikać m.in. z uruchomienia nowego systemu informatycznego.

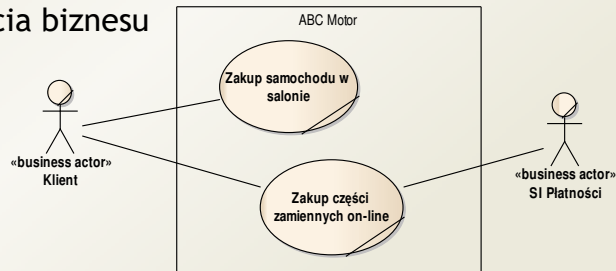
## Granica systemu, granica organizacji

- Jak biznes korzysta z systemu?
  - Interfejs użytkownika (GUI, e-commerce)
  - Interfejs systemowy (API, e-business)
- Jak biznes komunikuje się z otoczeniem?
  - Interfejs „ludzki” (kontakt osobisty)
  - Interfejs użytkownika (GUI, e-commerce)
  - Interfejs systemowy (API, e-business)



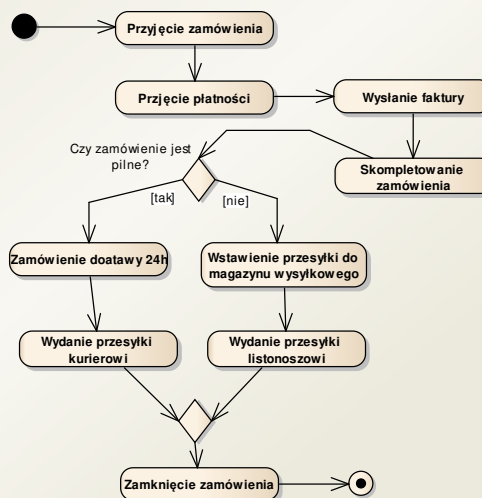
## Proces biznesowy - przypadek użycia biznesu

- Co to jest proces biznesowy?
  - posiada jasno określone granice, wejście i wyjście,
  - składa się z sekwencji uporządkowanych czynności,
  - tworzy wartość dodaną dla określonego odbiorcy (beneficjenta procesu).
- Procesy biznesowe na diagramie
  - Aktorzy biznesowi
  - Przypadki użycia biznesu



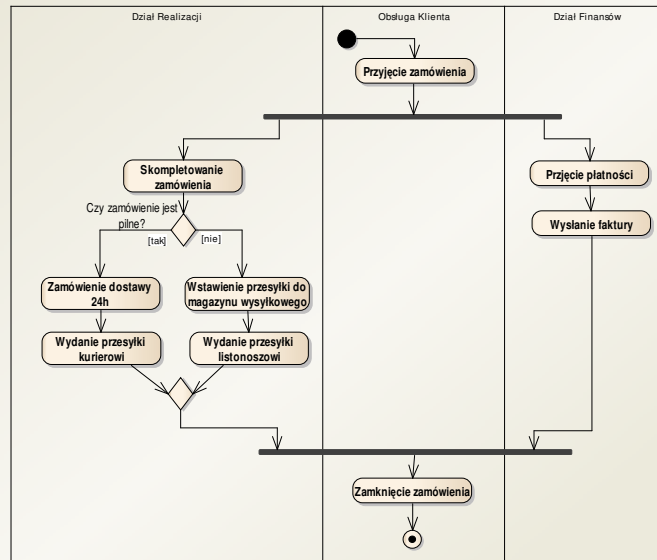
## Proces biznesowy w języku UML

- Diagram czynności
  - Węzeł początkowy
  - Akcje
  - Węzły decyzyjne
  - Przebiegi sterowania
  - Węzły końcowe



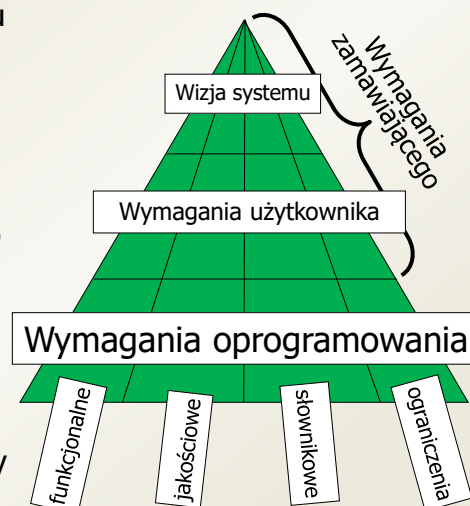


## Proces biznesowy w torami i synchronizacją



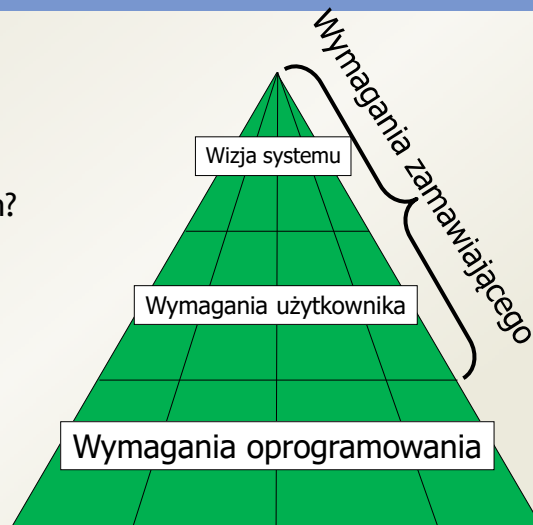
## Piramida wymagań

- Z punktu widzenia poziomu szczegółowości
  - Wizja systemu
  - Wymagania użytkownika
  - Wymagania oprogramowania
- Wizja i wymagania użytkownika stanowią wymagania zamawiającego
- Z punktu widzenia znaczenia wymagań
  - Wymagania funkcjonalne
  - Wymagania jakościowe
  - Wymagania słownikowe
  - Ograniczenia środowiskowe
- Im niższy poziom „piramidy wymagań” tym większa jego objętość



## Wymagania zamawiającego

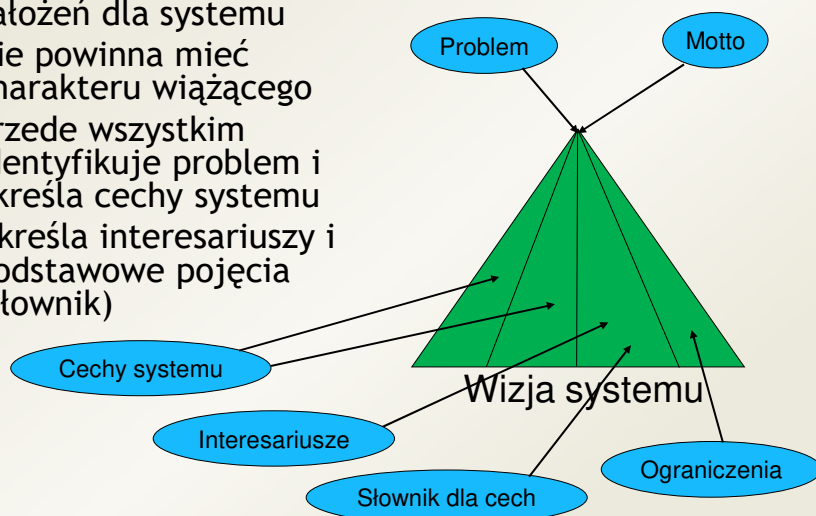
- Wymagania zamawiającego tworzymy po to, aby zamówić system
  - Po co nam ten system?
  - Jaki zakres ma mieć system?
  - Ile system będzie kosztował?
- Wymagania zamawiającego powstają na samym początku projektu



19

## Wizja systemu

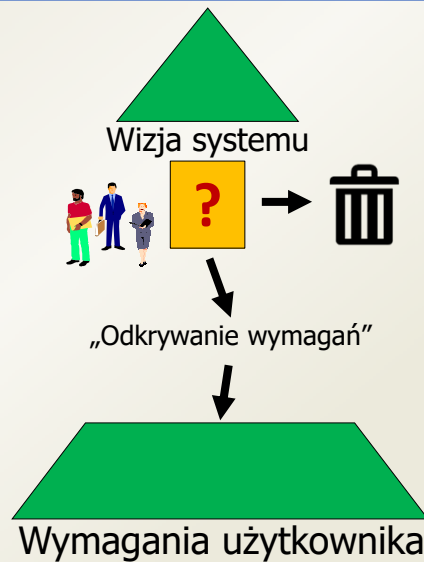
- Przedstawienie ogólnych założeń dla systemu
- Nie powinna mieć charakteru wiążącego
- Przede wszystkim identyfikuje problem i określa cechy systemu
- Określa interesariuszy i podstawowe pojęcia (słownik)



20

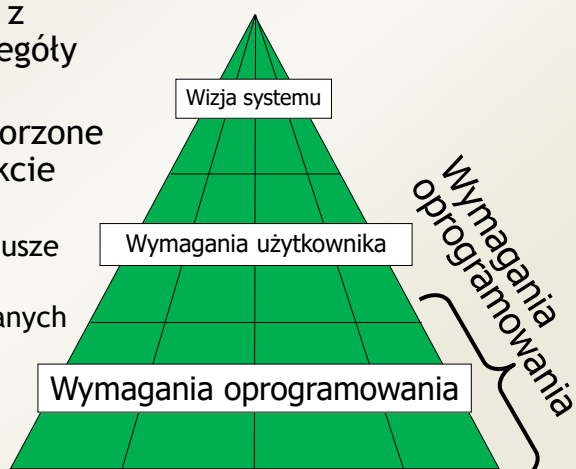
## Wymagania użytkownika

- Określenie zakresu systemu na podstawie wizji
- Cel: specyfikacja będąca podstawą kontraktu (umowy formalnej lub nieformalnej) między zamawiającym a wykonawcą systemu
- Ważne cechy wymagań użytkownika
  - Precyzja, eliminująca nieścisłości co do zakresu systemu
  - Ogólność opisu, wykluczająca sugerowanie rozwiązań technicznych (technologie dla interfejsu użytkownika, system bazy danych, itp.)



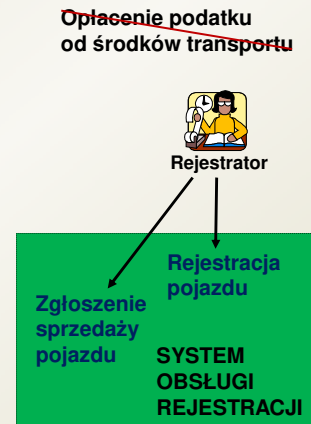
## Wymagania oprogramowania

- Wymagania oprogramowania tworzymy po to, aby uzgodnić z zamawiającym szczegóły
- Wymagania oprogramowania tworzone są na bieżąco w trakcie całego projektu
  - Szczegółowe scenariusze
  - Projekty ekranów
  - Szczegóły modelu danych



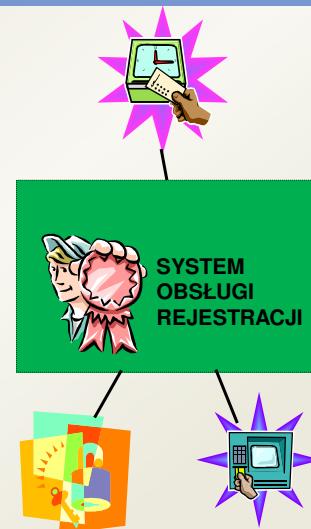
## Wymagania funkcjonalne

- Wymagania funkcjonalne określają sposób zachowania się systemu oprogramowania w odpowiedzi na interakcje użytkownika
  - Jakie usługi ma oferować system?
  - Jak reagować na określone komunikaty wejściowe?
  - Jak się zachowywać w określonych sytuacjach?
- Zestaw wymagań funkcjonalnych określa zakres budowanego systemu
- Zakres systemu możemy też określić stwierdzając, czego system nie powinien robić



## Wymagania jakościowe („pozafunkcjonalne”)

- Definiują charakterystyki określające sposób oceny działania systemu
  - Jak sprawny ma być system?
  - Jak niezawodny ma być system?
  - Jak bezpieczny ma być system?
  - Jak przyjazny ma być system?
  - Jakie normy ma spełniać system?
- Mogą być bezpośrednio związane z poszczególnymi wymaganiami funkcjonalnymi lub dotyczyć całości systemu (wybranej jego części)
- Bardzo istotne jest zdefiniowanie precyzyjnych metryk



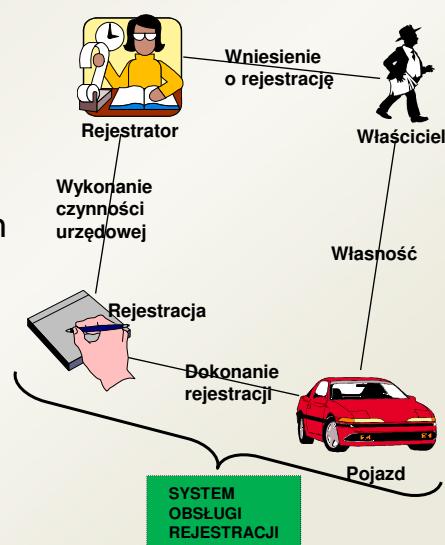
## Ograniczenia środowiskowe i techniczne

- Określają warunki, jakim powinien podlegać system ze względu na środowisko sprzętowe, software'owe i biznesowe
- Przykłady ograniczeń:
  - System musi pracować w określonej konfiguracji sieci lokalnej
  - System musi uwzględnić konieczność wykorzystania mało wydajnych maszyn jako stacji roboczych
  - System musi wykorzystywać posiadane zasoby systemu zarządzania bazami danych
  - System musi uwzględnić konieczność pracy w terenie

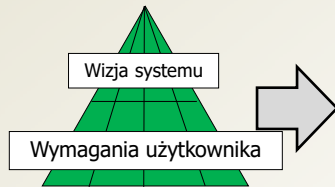


## Wymagania słownikowe (model dziedziny)

- Wymagania słownikowe definiują zakres pojęć i danych, jakimi system oprogramowania ma operować
- Słownik zawiera definicje pojęć używanych w ramach wymagań funkcjonalnych i jakościowych
- Oprócz samej definicji pojęć opisujemy też zależności między nimi. Zależności te mogą być wyrażone w formie graficznej



## Struktura specyfikacji wymagań



### I. Wprowadzenie

(Cel, streszczenie dla kierownictwa)

### II. Opis dziedziny problemu

#### 1. Model biznesu

(Otoczenie, procesy biznesowe)

#### 2. Wizja systemu

(Problem, cechy systemu, motto)

### III. Zakres systemu

#### 3. Wymagania funkcjonalne

##### 3.1. <Pakiet funkcjonalności systemu>

(Opis pakietu, diagramy, opisy przypadków użycia)

#### 4. Wymagania jakościowe

(Lista wymagań z metrykami)

#### 5. Ograniczenia

(Lista ograniczeń środowiskowych i technicznych)

#### 6. Słownik

##### 6.1. Użytkownicy i systemy zewnętrzne

(Diagram, lista z definicjami)

##### 6.2. <Pakiet słownika>

(Diagram, lista z definicjami)

- Formalizacja dokumentu zależy od przyjętej metodyki i uzgodnień z klientem