Uniwersytet Ekonomiczny we Wrocławiu

STUDIA PODYPLOMOWE

# Narzędzia informatyczne wspierające przedsięwzięcia e-commerce

## Zarządzanie projektami e-commerce

*Marcin Moś*

**Marcin Moś, Meblini.pl, UE we Wrocławiu**

**Oleśnica ☺ Online 2021-05-30**

# O sobie

- Studia
- PlugIT
- Ceneo
- Meblini.pl

**Wasze doświadczenie?**

**Analiza biznesowa?**

**Zarządzanie wymaganiami?**

**Narzędzia IT?**

1. **Cykl życia projektu**
2. **Pomysł / Planowanie**
3. **Analiza projektu**
4. **Wymaganie projektowe**
5. **Modelowanie zakresu projektu**
   - ❑ Model procesu
   - ❑ Makieta interfejsu
   - ❑ Inne modele
6. **Planowanie projektu**
7. **Implementacja projektu**

Cykl życia projektu (Model fazowy projektu):

Dla wygody i poprawy skuteczności zarządzania wydziela się w projekcie jego etapy (**fazy**), wynikają one z czasowego podziału jego realizacji. Wynik każdej fazy to **kamień milowy** (punkt kontrolny).

Okresy pomiędzy kamieniami milowymi to fazy projektu. Suma faz to **cykl życia projektu**.

Każda faza **projektu** też jest **projektem**

Projekty e-commerce (informatyczne):

- Pomysł / Planowanie
- Analiza
- Projektowanie
- Implementacja
- Testowanie
- Wdrożenie
- Promocja / Sprzedaż
- Posprzedażowa obsługa Klienta

# Faza 0 (Pomysł / Planowanie projektu)

- Zebranie pomysłów / myśli
- Wstępne badanie rynku / zapotrzebowania
- Wstępne badanie konkurencji

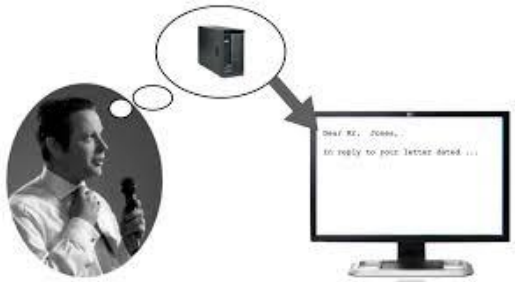
- Pretotyping
- Design sprint

Pretotypig:

„Przetestowanie wstępnego zainteresowania i wykorzystania potencjalnie nowego produktu poprzez symulację jego głównego działania przy nakładzie minimalnego czasu i pieniędzy”

„Upewnij się – najszybciej i najtaniej jak się da – że tworzysz właściwy produkt, zanim stworzysz go właściwie”

IBM „speech-to-text”



Palm Pilot

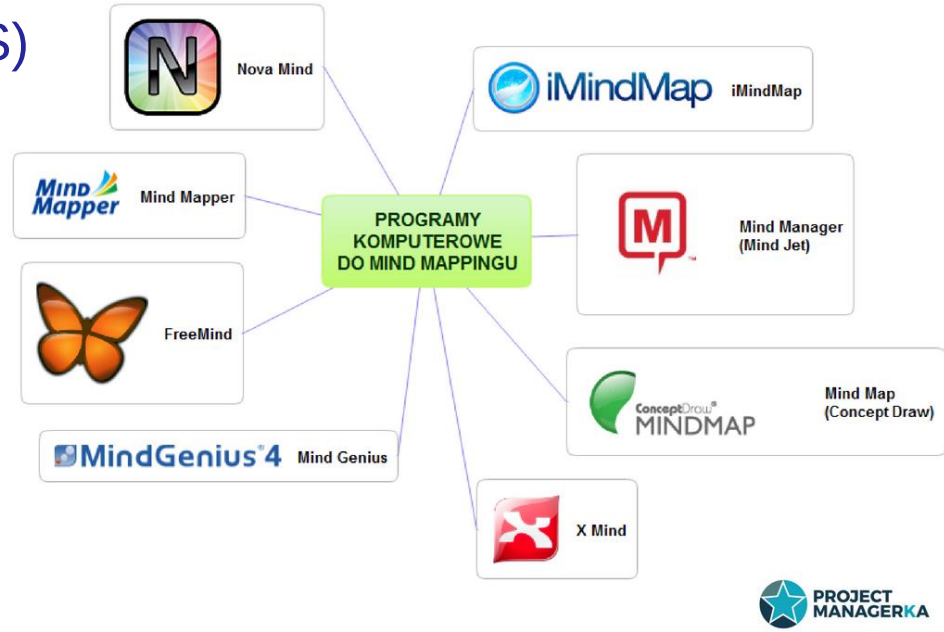
[Źródło: https://marketerplus.pl/pretotyping-ograniczyc-straty-uniknac-fiaska-nowych-produktachpomyslachuslugach/]

# Planowanie:

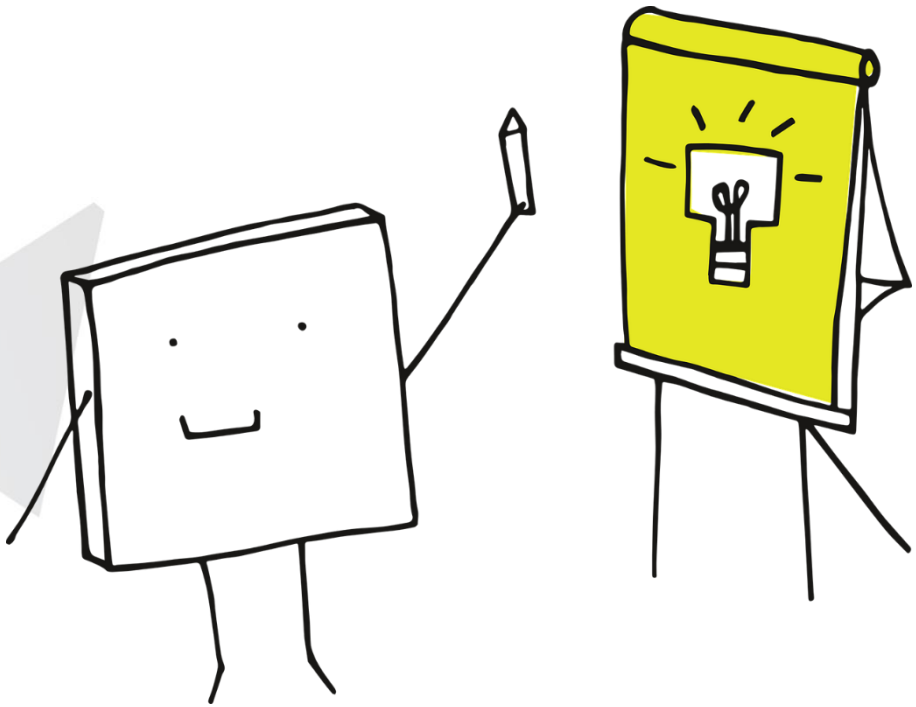
- Przygotowanie zarysu WBS (Work Breakdown Structure) - zasięg przedsięwzięcia określa się przy pomocy hierarchicznej struktury drzewa
- Określenie wstępnego zakresu projektu
- Uzyskanie akceptacji interesariuszy

# Faza 0 - Aplikacje

- **Mind mapping** (PC / Android / IOS)
  - iMindMap, SimpleMind, MindMemo, …
- **Zakres**
  - Google Docs, Microsoft Office, OpenOffice,…



# Wasza doświadczenia – pomysł / planowanie?



# Faza 1 (Analiza projektu)

- Cele projektu
- Interesariusze projektu
- Rozpoznanie i definiowanie potrzeb
- Przekształcenie potrzeb w wymagania
  - Funkcjonalne
  - Niefunkcjonalne
  - Techniczne
- Doprecyzowanie / Modelowanie wymagań
  - Model procesu
  - Makiety interfejsu
  - Modele UML
  - Wzory dokumentów / raportów…
- Produkty projektu

# Formułowanie celów SMART projektu

## Główny cel projektu oraz cele szczegółowe powinny być:

- Specific / konkretny, sprecyzowany – jednoznacznie i prosto zapisany, ale jednocześnie szczegółowo
- Measurable / mierzalny – cel musi być mierzalny, aby nim zarządzać i móc stwierdzić, czy został zrealizowany i w jakim stopniu
- Achievable / osiągalny – muszę mieć pewność, że cel przy posiadanych zasobach (czas, finanse, ludzie, wiedza, etc.) jest realny do wykonania
- Relevant / istotny – skutek realizacji celu wprowadzi ważną zmianę, co motywuje, determinuje i działa na ambicję – musisz czuć, że realizacja celu jest ważna i potrzeba
- Time-defined / określony w czasie – muszę określić, na kiedy mam wykonać zadania, z jaką częstotliwością i na kiedy mają być gotowe poszczególne etapy realizacji celu

[Źródło: https://www.intellect.pl/blog/okresl-swoje-cele-biznesowe-i-zacznij-projekt-www/]

# CELE PROJEKTU

**Co** chcesz osiągnąć?

**Po co** chcesz to osiągnąć?

**Jak** chcesz to osiągnąć?

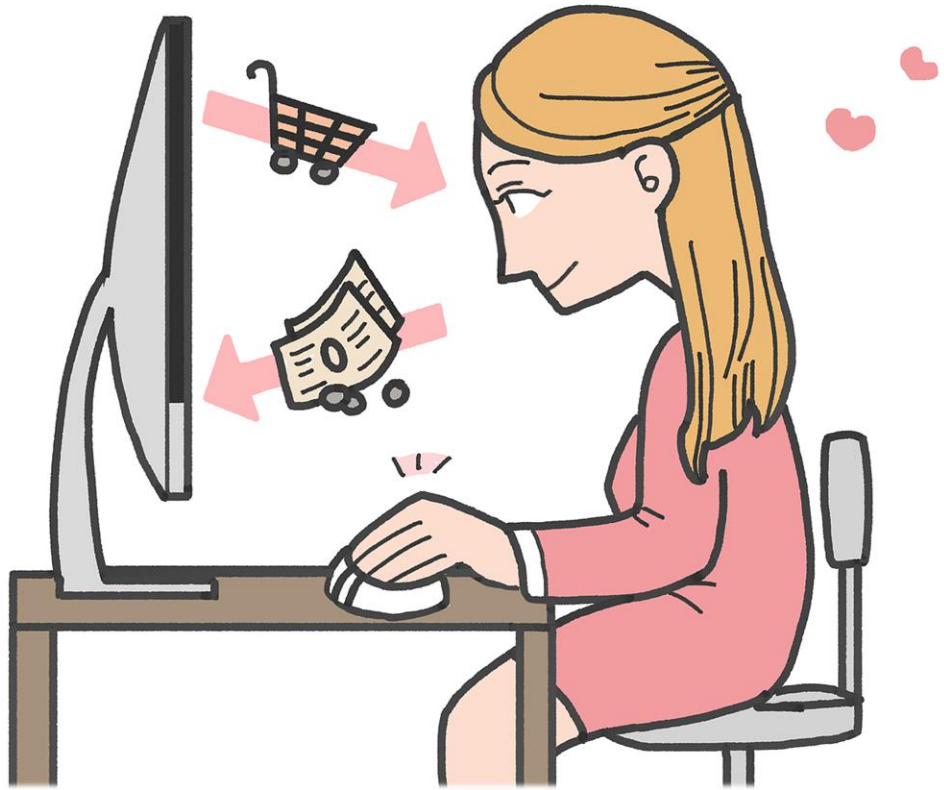
## Przykładowe CELE:

- zmniejszenie kosztów dystrybucji informacji dla prasy i materiałów handlowych o 50 %
- zmniejszenie liczby połączeń telefonicznych odbieranych przez pracowników o 30 % dzięki implementacji rozwiązania do przyjmowania zgłoszeń online
- dotarcie do potencjalnych klientów za pośrednictwem Internetu – 10 leadów/ tydzień
- zwiększenie kwartalnej liczby wartościowych leadów o 15% (wartościowy lead to: uzupełnione wszystkie pola formularza, zgoda na kontakt marketingowy
- usprawnienie i przyśpieszenie procesowania zamówień o 30 % w ciągu pół roku
- zwiększenie konwersji o 8,5% przy tym samym poziomie ruchu na stronie
- zwiększenie kwartalnej sprzedaży online o 17%
- zwiększenie konwersji sprzedaży o 12% przy zachowaniu tego samego ruchu na stronie

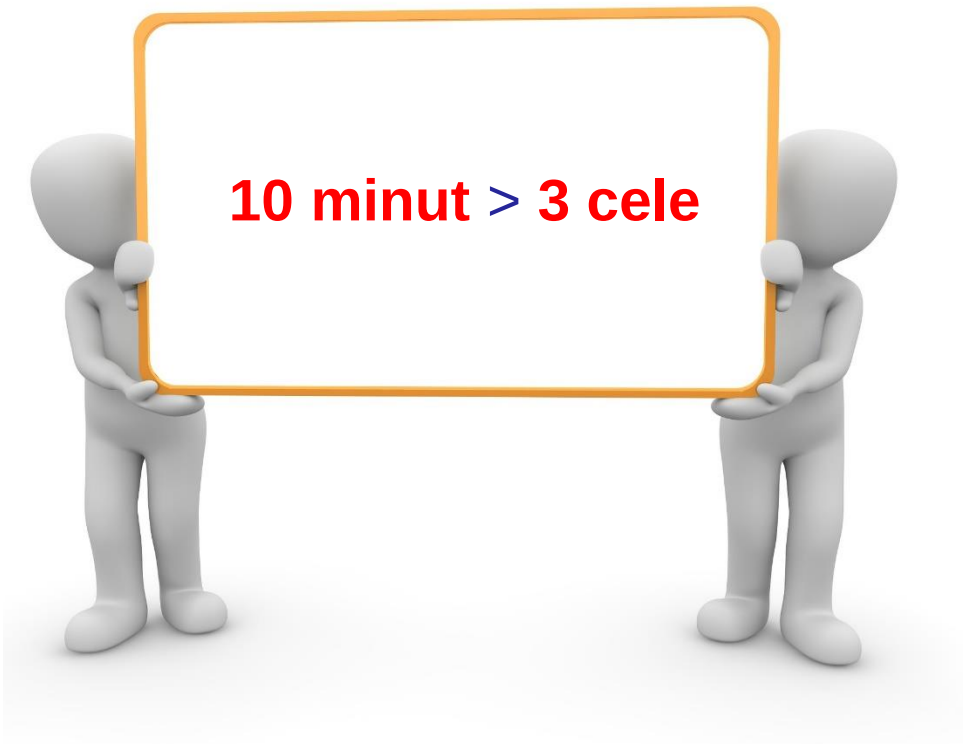
[Źródło: https://www.intellect.pl/blog/okresl-swoje-cele-biznesowe-i-zacznij-projekt-www/]

[Źródło: https://www.depcore.pl/artykul/przyklady-celow-smart-dla-stron-internetowych]

# Cele w sklepie internetowym?



# TWÓJ PROJEKT / POMYSŁ - TWOJE CELE?



# Analiza biznesowa

Celem analizy biznesowej jest uzyskanie fundamentalnych informacji o planowanym projekcie: jego celach, priorytetach, powiązaniach z długofalową polityką firmy. Rezultaty analizy pozwalają ustalić, jakie wymierne korzyści uzyskuje firma dzięki wdrażanym rozwiązaniom

Analiza biznesowa jest procesem, którego celem jest zidentyfikowanie potrzeb Klienta w zakresie funkcjonalności oraz innych biznesowych aspektów dotyczących systemu informatycznego.

Techniki pozyskiwania wymagań:

- Burza mózgów
- Analiza dokumentacji
- Analiza interfejsów
- Wywiad
- Obserwacja
- Prototypowanie
- Warsztaty

Problemy z definiowaniem wymagań:

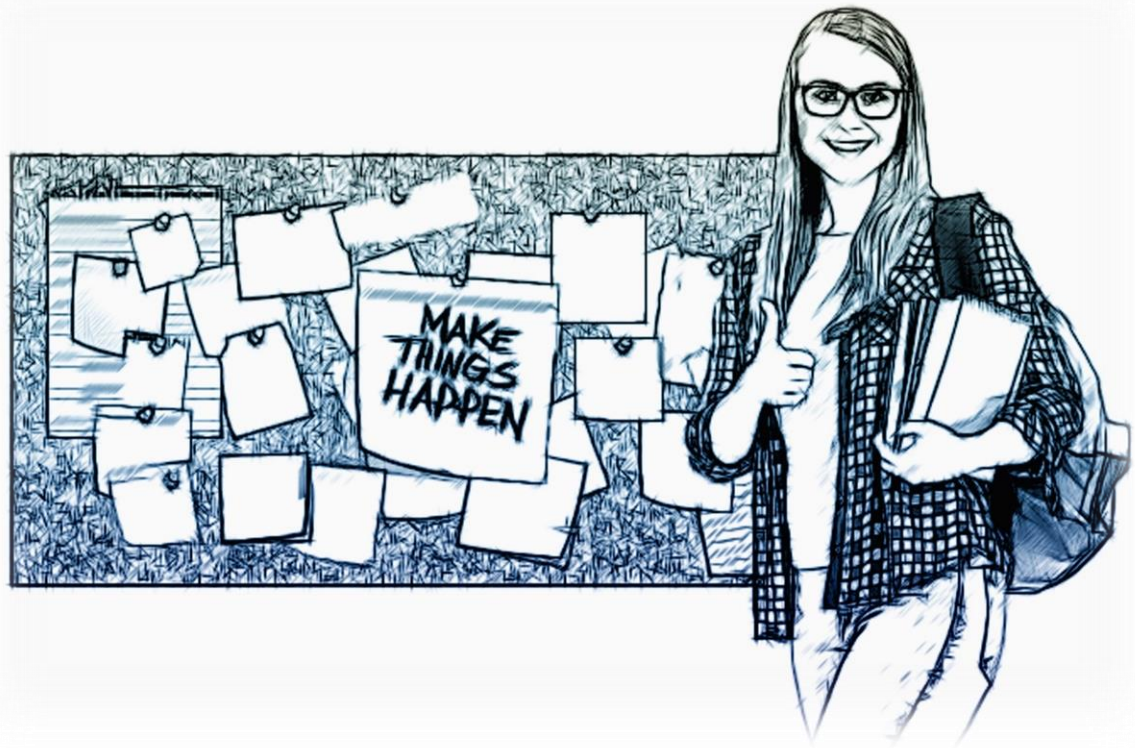
- zdefiniowane nieprawidłowo
- zdefiniowane nieprecyzyjnie / niejednoznacznie
- są zmienne

Jedno spotkanie lub rozmowa telefoniczna często wiążą się z koniecznością wprowadzenia istotnych korekty we wcześniej przygotowanym opracowaniu. W rezultacie specyfikacja / product backlog jest dokumentem dynamicznym – na bieżąco powinny być tam rejestrowane wymagania i wszelkie zmiany wcześniejszych założeń

Wymaganie funkcjonalne

Wymagania niefunkcjonalne

Wymagania techniczne



Analiza wymagań funkcjonalnych umożliwia zidentyfikowanie i opisanie pożądanego zachowania systemu.

Wymaganie funkcjonalne:

*„stwierdzenie, jakie usługi ma oferować system, jak ma reagować na określone dane wejściowe oraz jak ma się zachowywać w określonych sytuacjach. W niektórych wypadkach wymagania funkcjonalne określają, czego system nie powinien robić”*

Dobrze napisane wymagania:

- Prosto
- Spójnie
- Zrozumiale
- Zwięźle
- Precyzyjnie - jeśli wymaganie można błędnie zinterpretować, to tak się stanie (Murphy)

Korzystajmy z wizualizacji! ☺

Wymagania opisane w formie tekstowej powinny obejmować

- opis **możliwości** systemu,
- **warunków**, jakie muszą zaistnieć, by wymaganie mogło zostać spełnione i zrealizowane , oraz
- wszelkich **ograniczeń**, jakie uniemożliwiają spełnienie wymagania w projektowanym systemie

Wymagania funkcjonalne, to m.in.:

- ✓ opisana ogólna wizja systemu, cele jakie ma spełniać oraz kontekst jego powstania (np. jakiego typy problemów zostały zdefiniowane w firmie, z jakiego typu trudnościami wiąże się brak aplikacji, itd.)
- ✓ opis użytkowników (aktorów, grup) i ich uprawnień (ról) – innego typu uprawnienia mogą mieć Administratorzy, którzy będą opiekować się aplikacją, inne kierownicy, a inne szeregowi pracownicy;
- ✓ opis funkcji systemu oraz sposoby korzystania z niego przez każdego z aktorów, tj. jakie procesy w nim wykonuje;
- ✓ opis formularzy, wyświetlanych danych i operacji na nich wykonywanych;
- ✓ zakres domyślnych raportów, które będą mogły być automatycznie generowane.

Wymaganie niefunkcjonalne to:

- ograniczenia nakładane na budowane rozwiązanie, które nie wiążą się bezpośrednio z zadaniami, które wykonuje
- są one nie mniej istotne niż wymagania funkcjonalne.

Przykład: parametry jakościowe stawiane przed aplikacją, np. określenie średniego czasu, w jakim ma się otwierać wywołany przez użytkownika interfejs (może on się otwierać 1 sekundę lub 10).

- maksymalna ilości użytkowników, którzy będą mogli w jednym czasie swobodnie korzystać z aplikacji
- maksymalny czas wdrożenia oraz deadline na uruchomienie
- wymagania wobec dokumentacji, która ma zostać opracowana przez wykonawcę (np.: projekt architektury informatycznej, projekt migracji danych, specyfikacja sprzętowa, specyfikacja funkcjonalna, harmonogram wdrożenia, instrukcje dla użytkowników itp.)
- oczekiwania wobec dostawcy, co zadań przez niego wykonywanych po samym wdrożeniu (np. szkolenia dla pracowników, gwarancja, zakres wsparcia powdrożeniowego, SLA, itp.)
- zasady bezpieczeństwa (back-up danych, itp.)
- sposób obsługi niedostępności serwera
- kolorystyka interfejsów (lub zasady umieszczania logo firmy)

Wymaganie techniczne, np:

- opis środowiska, w którym nowym system będzie funkcjonował, np.: systemy operacyjne stacji roboczych przyszłych użytkowników (Windows, Linux, iOS), przeglądarki internetowe (jakie?), klienci pocztowi wykorzystywani w organizacji;
- opis systemów, z którymi będzie zintegrowany (wymagania, ograniczenia);
- opis sposobu logowania użytkowników,
- zasady dostępu przez VPN i urządzenia mobilne, itd.

Uzupełnienie specyfikacji:

- modele
- makiety
- szkice formularzy
- wzorce dokumentów generowanych automatycznie (raporty, faktury,…)
- przykładowe dokumenty, które są obecnie „w obiegu”, a które mają być zastąpione przez aplikację
- procedury obowiązujące w firmie (np. procedura bezpieczeństwa)

Najczęstsze błędy:

- brak jasno sprecyzowanych priorytetów stawianych przed aplikacją / powinno być jasno opisane, które elementy mają kluczowe znaczenie, a które drugorzędne
- brak dokładnej analizy problemu, który aplikacja ma rozwiąza / bywa tak, że aplikacja wcale nie jest rozwiązaniem, gdyż problem leży gdzie indziej

Najczęstsze błędy:

- nadmierność wymagań – w trakcie konsultacji wewnątrz firmy i spotkań z różnymi pracownikami można bardzo łatwo natrafić na prawdziwy koncert życzeń i prześciganie się w coraz to bardziej oryginalnych pomysłach

  Cele powinny być jasno zdefiniowane – jeżeli jakieś wymaganie nie jest odpowiedzią na żaden z celów to powinno być usunięte z listy / bardzo łatwo stworzyć skomplikowaną aplikację

Najczęstsze błędy:

- brak umiejętności przełożenia „myśli na papier” – pisanie w sposób ogólnikowy i skrótowy / często autor specyfikacji wychodzi z założenia, że wszyscy się domyślą o co mu chodzi
  W trakcie realizacji nie ma już czasu na pytania i doprecyzowanie. Brak jednoznaczności wymagań prowadzi często do licznych nieporozumień i nerwów. W sytuacjach spornych z dostawcą, w pierwszej kolejności odwołuje się do specyfikacji, jeżeli coś zostanie tam niedokładnie opisane to istnieją duże szanse na to, że zostanie także niedokładnie odwzorowane w aplikacji

Najczęstsze błędy:

- niekonsultowanie dokumentu z innymi działami firmy.
  Należy zdefiniować wszystkich interesariuszy, których będzie dotyczyć budowane rozwiązanie - każdy z nich powinien się wypowiedzieć, a następnie zatwierdzić zakres wymagań
- pomijanie wymagań niefunkcjonalnych
- skoncentrowanie się na wymaganiach technicznych, a nie na ludziach i ich potrzebach. Zbudowana aplikacja jest poprawna technicznie, ale nie spełnia oczekiwań w niej pokładanych (mówiąc krócej – jest bezużyteczna)

Najczęstsze błędy:

- powtórzenia, sprzeczności, brak uporządkowanej struktury dokumentu (np. pomieszczane wymagania z różnych obszarów) Dokument specyfikacji powinien mieć swoją logikę, jasny podział na rozdziały – np. wg. poszczególnych modułów aplikacji
- brak zebranych potwierdzeń („okejek') ze strony managmentu dla końcowej wersji specyfikacji. Praktyka pokazuje, że wiele osób ma zaskakującą skłonność do częstego zmieniania zdania – i nie ma co z nimi bazować na ustnych ustaleniach. Szczególnie, gdy w grę wchodzą duże pieniądze

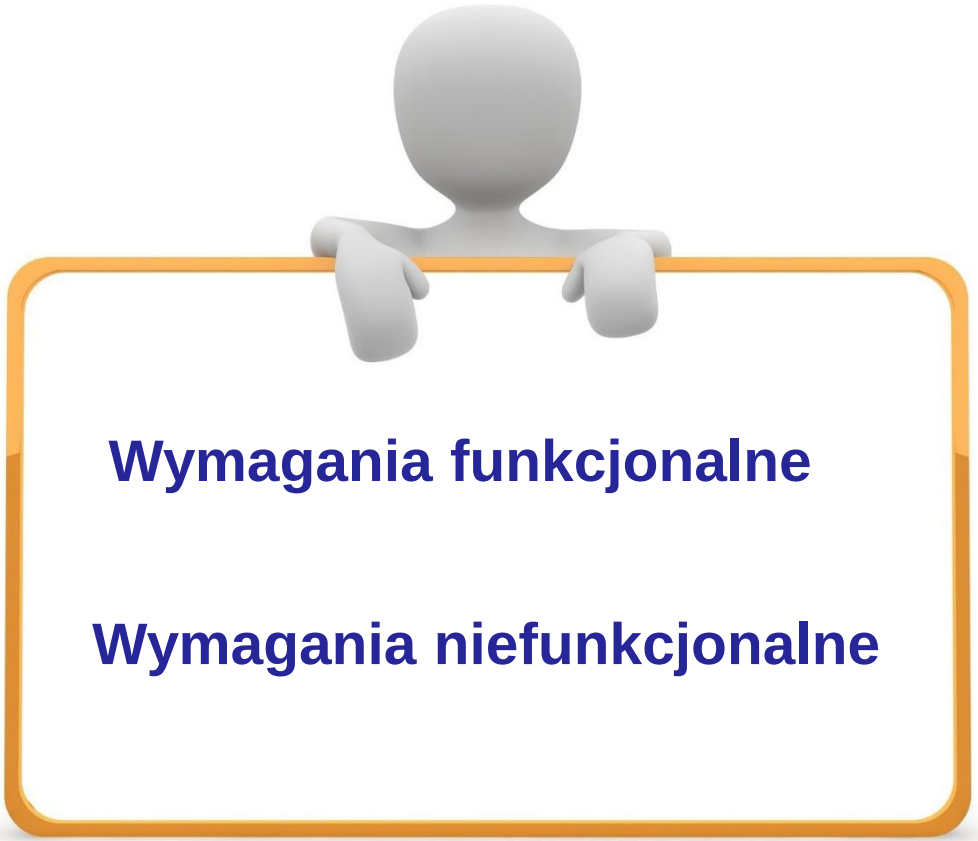
Najczęstsze błędy:

- używanie niezrozumiałej terminologii i brak objaśnień / słownika na końcu dokumentu
- niekompletność, wychodzenie z założenia, że wszystkie wymagania spisze się dopiero na etapie projektu, ewentualnie nie będzie problemu z dodaniem kolejnych funkcjonalności (problemu zwykle nie ma tylko, że po tym jak zostanie zatwierdzony zakres i budżet projektu, każda dodatkowe wymaganie to dodatkowe koszty, na które niekoniecznie będzie zgoda)

Przykładowa lista wymagań:

- System umożliwia zalogowanie się użytkownika za pomocą loginu i hasła
- System waliduje poziom uprawnień użytkownika
- System blokuje dostęp po 3 nieudanych próbach logowania

# Twój pomysł / projekt – Twoje wymagania?



**Wymagania funkcjonalne**

**Wymagania niefunkcjonalne**

System rozumieć można jako obiekt obsługujący wybrane zdarzenia mające miejsce w rzeczywistym świecie. Efektem analizy określonej dziedziny może być **lista powiązanych zdarzeń**.

Określenie zakresu polega na wybraniu tych zdarzeń, które system będzie obsługiwał (w zakresie) i tych, które nadal będą przebiegały całkowicie w świecie rzeczywistym i nie będą wspierane (poza zakresem).

## Kroki określania zakresu:

- identyfikacja zdarzeń mających miejsce na granicy rozpatrywanego rozwiązania
- wybranie zdarzeń obsługiwanych przez rozwiązanie
- określenie reakcji na zdarzenie – listy czynności wykonywanych w celu jego obsłużenia – czyli procesu biznesowego

# Przykład: Zgłoszenie zamówienia

Proces:

- ✓ Przyjęcie zamówienia (w systemie)
- ✓ Skompletowanie pozycji zamówienia (poza systemem)
- ✓ Przygotowanie paczki (poza systemem)
- ✓ Wysłanie zamówienia (poza systemem)
- ✓ Zmiana statusu zamówienia – widoczna dla klienta (w systemie)

## Kroki określania zakresu:

- identyfikacja zdarzeń mających miejsce na granicy rozpatrywanego rozwiązania
- wybranie zdarzeń obsługiwanych przez rozwiązanie
- określenie reakcji na zdarzenie – listy czynności wykonywanych w celu jego obsłużenia – czyli procesu biznesowego

Uniwersytet Ekonomiczny we Wrocławiu

STUDIA PODYPLOMOWE

# Faza 1 - Aplikacje

- **Zarządzanie przedsięwzięciem / projektem**

Zarządzanie czasem

Zarządzanie wymaganiami

Any.do todoist ZenDay Forward thinking

EVERNOTE Wunderlist Trello

## PRZEGLĄD APLIKACJI

*do zarządzania czasem i projektami*

Nozbe habitica

JIRA asana

EasilyDo MINDLY Google Keep

* http://zorganizowana.com/przeglad-aplikacji/

# Faza 1 - Aplikacje

- **Tworzenie specyfikacji rozwiązania**
  - Makiety
  - Modele procesów
  - …
- **Komunikacja**

* https://webroad.pl/webdesign/706-narzedzi-do-tworzenia-makiet-porownanie

# Twój pomysł / projekt – Twoje wymagania?

Modelowanie zakresu projektu

**Identyfikacja zdarzeń**

**Modelowanie procesu**

**Makiety interfejsu**

## Twój pomysł / projekt – Twoje wymagania?

Modelowanie wyglądu docelowego aplikacji

## Makiety interfejsu (Pencil)

Implementacja:

- Realizacja zakresu projektu
- Testy
- Testy UAT
- Wdrożenie na środowisko produkcyjne

- Systemy wspomagające zarządzanie wymaganiami

# Twój pomysł / projekt – Twoje wymagania?

Zarządzanie wymaganiami projektowymi

## Monitoring i zarządzanie projektem (Trello)

# Przydatne linki / Źródła:

https://www.intellect.pl/blog/okresl-swoje-cele-biznesowe-i-zacznij-projekt-www/

https://www.magazyn-ecommerce.pl/artykul/jak-skutecznie-planowac-i-wyznaczac-cele-w-e-sklepie

https://consider.pl/jak-napisac-specyfikacje-wymagan-biznesowych/

https://consider.pl/aplikacje-i-narzedzia-marketingowe/

https://marketerplus.pl/pretotyping-ograniczyc-straty-uniknac-fiaska-nowych-produktachpomyslachuslugach/

https://productvision.pl/2017/design-sprint-zwaliduj-dowolna-hipoteze-ciagu-5-dni/

https://www.depcore.pl/artykul/przyklady-celow-smart-dla-stron-internetowych

https://zerobs.pl/zarzadzanie-projektami/trello-jak-korzystac/