



PROGRAMOWANIE APLIKACJI MOBILNYCH

WYKŁAD 10

Publikacja aplikacji mobilnej

dr inż. Mateusz Pomianek

Katedra Informatyki i Automatyki, Politechnika Rzeszowska

m.pomianek@prz.edu.pl // [D.103](#) // <http://www.kia.prz.edu.pl> // <https://materialy.kia.prz.edu.pl>



1. Dostępność
2. SDLC - Proces tworzenia oprogramowania
3. Projekt aplikacji mobilnej
4. Brief aplikacji
5. BRD – Analiza wymagań biznesowych
6. FRD – Analiza wymagań funkcjonalnych
7. TRD - Analiza wymagań technicznych



Tworzenie aplikacji

Testowanie, poprawki

Publikacja aplikacji

- samodzielna
- w firmie

Harmonogram pracy
wyznaczany przez proces
biznesowy



Many of you might not agree but this is how peak career trajectory looks like

Experience



Goose farmer

Self-employed

Jul 2023 - Present · 10 mos

On-site



Microsoft

22 yrs 4 mos



Principal Performance Architect

Full-time

Jul 2022 - Jul 2023 · 1 yr 1 mo

Chehalis, Washington, United States



Principal Software Development Engineer

Apr 2001 - Jul 2022 · 21 yrs 4 mos



„Ogólnie przyjęte” nie oznacza, że opisana wiedza i praktyki są lub powinny być stosowane jednolicie we wszystkich projektach; zespół zarządzający projektem jest zawsze odpowiedzialny za określenie, co jest odpowiednie dla danego projektu.

- Project Management Institute in its
Guide to the Project Management Body of Knowledge



Dostępność



1. Sklep Play → Konsola Play

2. API Key

3. Konsola play

4. Google analytics

- Google Analytics Certification
- [github/googleanalytics/](https://github.com/googleanalytics/)





Test zamknięty

Google Play Console

← Wszystkie aplikacje

Panel

Statystyki

Przegląd publikowanych zmian

Testuj i publikuj

Monitoruj i ulepszaj

Zwiększaj liczbę użytkowników

Zarabiasz w Google Play

Panel

Dokończ konfigurację aplikacji

Podaj informacje o aplikacji i utwórz jej stronę

Poinformuj nas o zawartości aplikacji i zarządzaj sposobem jej prezentowania w Google Play

Ukryj zadania ^

OKREŚL TREŚĆ APLIKACJI

- Określ politykę prywatności >
- Dostęp aplikacji >
- Reklamy >
- Ocena treści >
- Odbiorcy docelowi >
- Bezpieczeństwo danych >
- Aplikacje instytucji państwowych >
- Funkcje finansowe >
- Zdrowie >

ZARZĄDZAJ SPOSOBEM PREZENTOWANIA APLIKACJI

- Wybierz kategorię aplikacji i podaj dane kontaktowe >
- Dodaj informacje o aplikacji >

Produkcyjna



Wykryj problemy w aplikacji, zbierz opinie i uzyskaj dostęp produkcyjny

W ramach testów zamkniętych możesz udostępnić aplikację dużej grupie użytkowników, nad którą masz kontrolę. Dzięki temu możesz wykryć problemy, zebrać opinie i sprawdzić, czy wszystko jest gotowe do wprowadzenia aplikacji na rynek. Zanim poprosisz o możliwość udostępnienia wersji produkcyjnej wszystkim użytkownikom, musisz przeprowadzić test zamknięty. [Więcej informacji](#)

Aby rozpocząć test zamknięty, dokończ konfigurowanie aplikacji

Ukryj zadania ^

SKONFIGURUJ ŚCIEŻKĘ TESTÓW ZAMKNIĘTYCH

Wybierz kraje i regiony

Wybierz testerów

UTWÓRZ I OPUBLIKUJ WERSJĘ

Utwórz nową wersję

Wyświetl podgląd wersji i potwierdź ją

Wyślij wersję do Google w celu sprawdzenia

Prześlij prośbę o możliwość udostępnienia aplikacji w wersji produkcyjnej

Wersja produkcyjna aplikacji to ta, którą udostępniasz miliardom użytkowników w Google Play. Zanim poprosisz o możliwość udostępnienia aplikacji w wersji produkcyjnej, musisz przeprowadzić test zamknięty, który spełnia nasze kryteria. Podczas składania przez Ciebie wniosku poprosimy Cię także o odpowiedź na kilka pytań dotyczących testu zamkniętego. [Wyświetl podgląd pytań](#)

- Opublikuj wersję do testów zamkniętych
- Upewnij się, że do testu zamkniętego zgłosiło się co najmniej 12 testerów
W teście bierze obecnie udział 0 testerów
- Przeprowadź test zamknięty z udziałem co najmniej 12 testerów trwający minimum 14 dni

Poproś o opublikowanie wersji produkcyjnej

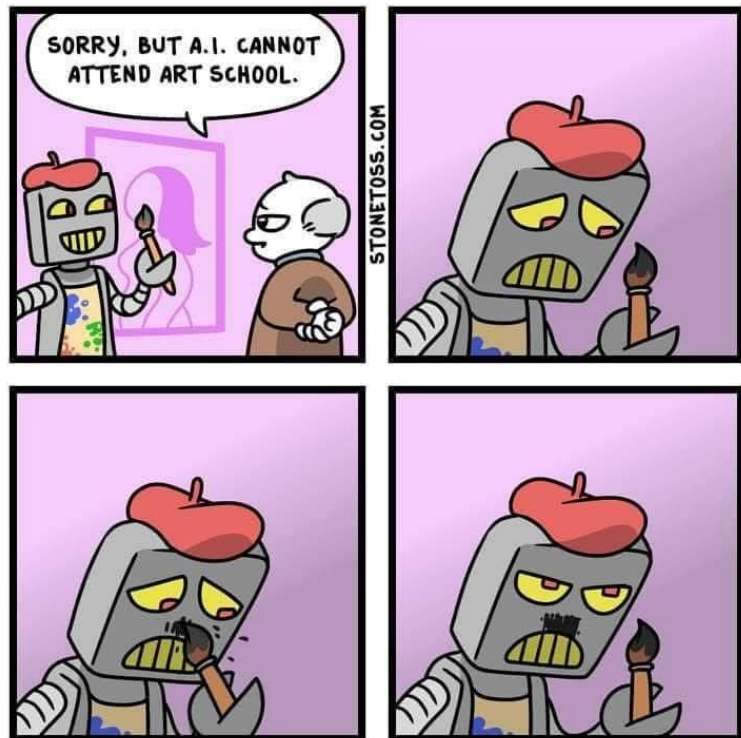
[Więcej informacji](#)



Aby pomóc użytkownikom w zakresie dostępności, system Android pozwala na **Stwórz usługę dostępności**, która może prezentować treści od aplikacji dla użytkowników, a także obsługiwać aplikacje w ich imieniu. Android zapewnia kilka usług dostępności systemu:

1. **TalkBack:** Pomaga osobom, które mają słabe widzenie lub są niewidome. Ogłasza treść poprzez Syntezował głos i wykonuje czynności w aplikacji w odpowiedzi na gesty użytkownika.
2. **Switch Access:** Pomaga osobom z niepełnosprawnością motorową. Podkreśla elementy interaktywne Wykonuje działania w odpowiedzi na naciśnięcie przez użytkownika przycisku. To pozwala Kontrolowanie urządzenia przy użyciu tylko jednego lub dwóch przycisków.

1. Etykiety treści
2. Ułatwienia dostępu w aplikacjach
3. Material 3 Expressive
4. Accessible design
5. Layout basics
6. Material A-Z
7. Testowanie ułatwień dostępu w aplikacji





1. Label elements
2. Add accessibility actions
3. Extend system widgets
4. Use cues other than color
5. Make media content more accessible





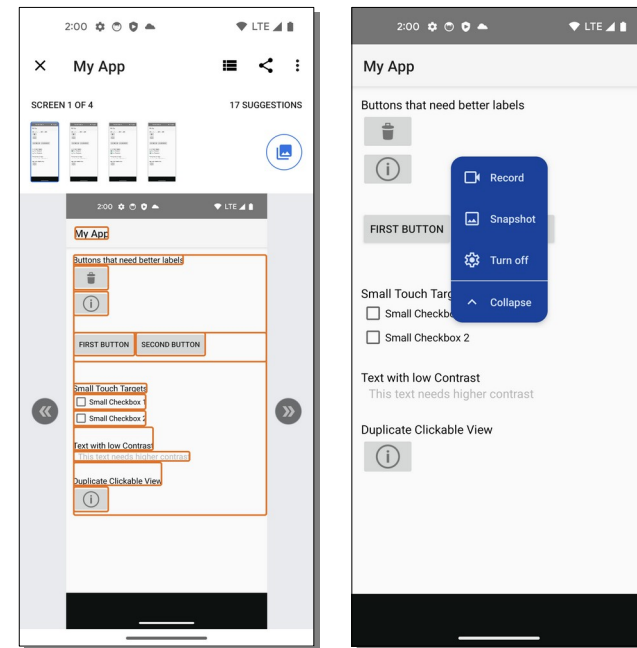
Ważne jest, aby zapewnić użytkownikom przydatne i opisowe etykiety dla każdego interaktywnego elementu interfejsu użytkownika w aplikacji. Każda etykieta musi wyjaśniać znaczenie i cel danego elementu. Czytniki ekranu, takie jak **TalkBack**, mogą ogłaszać te etykiety użytkownikom.

W większości przypadków opis elementu interfejsu użytkownika określa się w pliku zasobów układu, który zawiera element. Zazwyczaj etykiety dodaje się za pomocą atrybutu **contentDescription**. Istnieje kilka innych technik etykietowania opisanych w poniższych sekcjach.

Podczas etykietowania edytowalnych elementów, takich jak obiekty EditText, pomocne jest wyświetlanie tekstu, który podaje przykład prawidłowego wprowadzania danych w samym elemencie, a także udostępnianie tego przykładowego tekstu czytnikom ekranu. W tej sytuacji obiekt **View** musi mieć atrybut **android:labelFor** ustawiony na ID elementu **EditText**. Można użyć atrybutu **android:hint**, jak pokazano poniżej:

```
<!-- The hint text for en-US locale would be
      "Apartment, suite, or building". -->
<EditText
  android:id="@+id/addressLine2"
  android:hint="@string/aptSuiteBuilding" ... />
```

Accessibility Scanner to narzędzie, które skanuje interfejs aplikacji, aby podpowiedzieć, jakie ulepszenia związane z dostępem można do niej wprowadzić. Pozwala on, szybko i łatwo określić wiele typowych ulepszeń w zakresie ułatwień dostępu, takich jak powiększenie niewielkich obiektów dotykowych, wzmocnienie kontrastu tekstu i obrazów czy dodanie opisów treści do nieoznakowanych elementów graficznych.





Web Accessibility Initiative | WAI

Strategies, standards, resources to make the Web accessible to people with disabilities

Get Involved | About W3C WAI

Accessibility Fundamentals

Planning & Policies

Design & Develop

Test & Evaluate

Teach & Advocate

Standards/Guidelines

[Home](#) / [Accessibility Fundamentals](#) / [Digital Accessibility Courses](#) / **W3C Foundations Online Course**

Accessibility Fundamentals

Introduction to Accessibility

Accessibility is About People

Components of Web Accessibility

Accessibility Principles

Research

Digital Accessibility Courses

W3C Foundations Online Course

Course List

Digital Accessibility Foundations Free Online Course

Motivations

The “[Introduction to Web Accessibility](#)” online course provides **the foundation you need to make your digital technology accessible**, so that it:

- works well for people with disabilities
- meets international standards
- enhances the user experience for everyone
- achieves goals

Accessibility is essential to create high-quality apps, websites, and other digital content — and to not exclude people from using your products and services.

Page Contents

- [Who the course is for](#)

“Accessibility skills are increasingly in demand. With this course, you can build your accessibility skills and immediately apply what you learn in practice.”

— Ms. Natalia Amelina, [UNESCO Institute for Information Technologies in Education \(IITE\)](#)



Przerwa



Inżynieria wymagań (RE) jest procesem definiowania, dokumentowania i utrzymywania wymagań w procesie projektowania inżynierskiego. Jest to wspólna rola w inżynierii systemów i inżynierii oprogramowania.

Wraz z publikacją **IEEE Computer Society** w marcu 1997 r. oraz utworzeniu serii konferencji na temat inżynierii wymagań, która przekształciła się w **międzynarodową konferencję inżynierską**.

Koncepcja operacji (CONOPS) to dokument opisujący charakterystykę proponowanego systemu z punktu widzenia osoby, która będzie korzystać z tego systemu. Służy do przekazywania wszystkim interesariuszom ilościowych i jakościowych cech systemu. **CONOPS** są szeroko stosowane w wojsku, służbach rządowych i innych dziedzinach.



W inżynierii oprogramowania **proces tworzenia oprogramowania** (*SDLC, software development life cycle*) jest procesem planowania i zarządzania rozwojem oprogramowania. Zazwyczaj wiąże się to z podziałem pracy nad oprogramowaniem na mniejsze, równoległe lub sekwencyjne etapy lub podprocesy w celu zarządzania projektem.

Specyfikacja wymagań oprogramowania (*SRS, Software requirements specification*) to opis systemu oprogramowania, który ma być opracowany. Jest wzorowany na specyfikacji wymagań biznesowych (*CONOPS*). Specyfikacja wymagań oprogramowania określa wymagania funkcjonalne i нефункционаłne i może zawierać zestaw przypadków użycia, które opisują interakcje użytkownika, które oprogramowanie musi zapewnić użytkownikowi w celu uzyskania idealnej interakcji.



Prototypowanie oprogramowania to czynność polegająca na tworzeniu prototypów aplikacji oprogramowania, tj. niekompletnych wersji opracowywanego programu. Jest to czynność, która może wystąpić podczas tworzenia oprogramowania i jest porównywalna do prototypowania znanego z innych dziedzin, takich jak inżynieria mechaniczna lub produkcja.

Prototyp zazwyczaj symuluje tylko kilka aspektów i może całkowicie różnić się od produktu końcowego.



Minimalny funkcjonalny produkt (MVP) to wersja produktu z wystarczającą liczbą funkcji, aby mogła być używana przez pierwszych klientów, którzy mogą następnie przekazywać informacje zwrotne na temat przyszłego rozwoju produktu. Termin ten stworzył dyrektor generalny SyncDev, Frank Robinson w 2001 r. i spopularyzował Eric Ries w książce The Lean Startup z 2011 r. Trzy kluczowe słowa, które tworzą **MVP** są w nim nie bez znaczenia:

- **Minimum**, czyli minimalny. Docelowo ma ono posiadać tylko kilka niezbędnych funkcji
- **Viable** - w języku angielskim słowo to oznacza „wykonalny”, „zdolny do funkcjonowania”. Rozwiązanie musi być funkcjonalnie i działać tak, aby pierwsi użytkownicy mogli z niego łatwo skorzystać i podzielić się feedbackiem.
- **Product** – w tym przypadku dodatkowe wyjaśnienia chyba nie są potrzebne.

BABOK

Jest to kompleksowe **opracowanie zagadnień** związanych z **Analizą Biznesową**. Znajdziemy tam opisy związanych z nią obszarów wiedzy, a także katalog **umiejętności oczekiwanych od profesjonalistów** w tej dziedzinie.

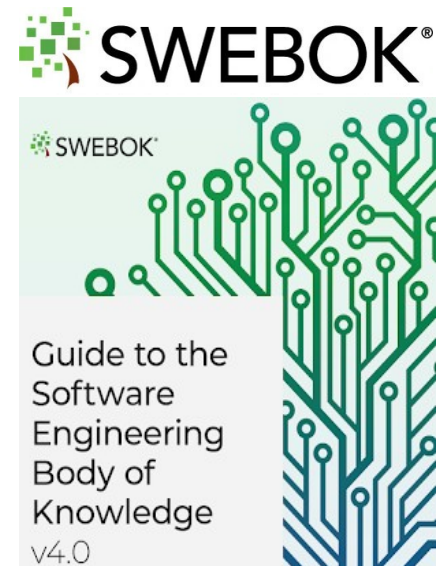
Definiowane są również **określone standardy** i **dobrze praktyki**, pozytywnie wpływające na **jakość** osiągniętych rezultatów i **efektywność działań**. Warto się z nimi zapoznać, aby „nie odkrywać koła na nowo”, ale uczyć się na błędach i doświadczeniach innych.



SWEBOK

Przewodnik po materiale wiedzy, oparty na inżynierii oprogramowania (**SWEBOK, V4.0, Software Engineering Body of Knowledge**), opublikowany przez IEEE Computer Society (IEEE CS), odzwierciedla obecny stan inżynierii oprogramowania. 18 obszarów wiedzy (KAs) podsumowuje kluczowe pojęcia i zawiera listę referencyjną dla szczegółowych informacji.

- Definiuje zawartość dyscypliny inżynierii oprogramowania;
- Promuje zrozumienie inżynierii oprogramowania;
- Kataloguje granice inżynierii oprogramowania;
- Zapewnia podstawy do szkolenia i rozwoju programu nauczania;
- Wspiera certyfikację i licencjonowanie inżynierów oprogramowania;
- Odzwierciedla obecne praktyki i integruje nowe technologie.





Introduction: <https://waseda.box.com/v/swebok-webinar-introduction>

Chapter 1: Software Requirements, <https://waseda.box.com/v/swebok-webinar-c1-requirements>

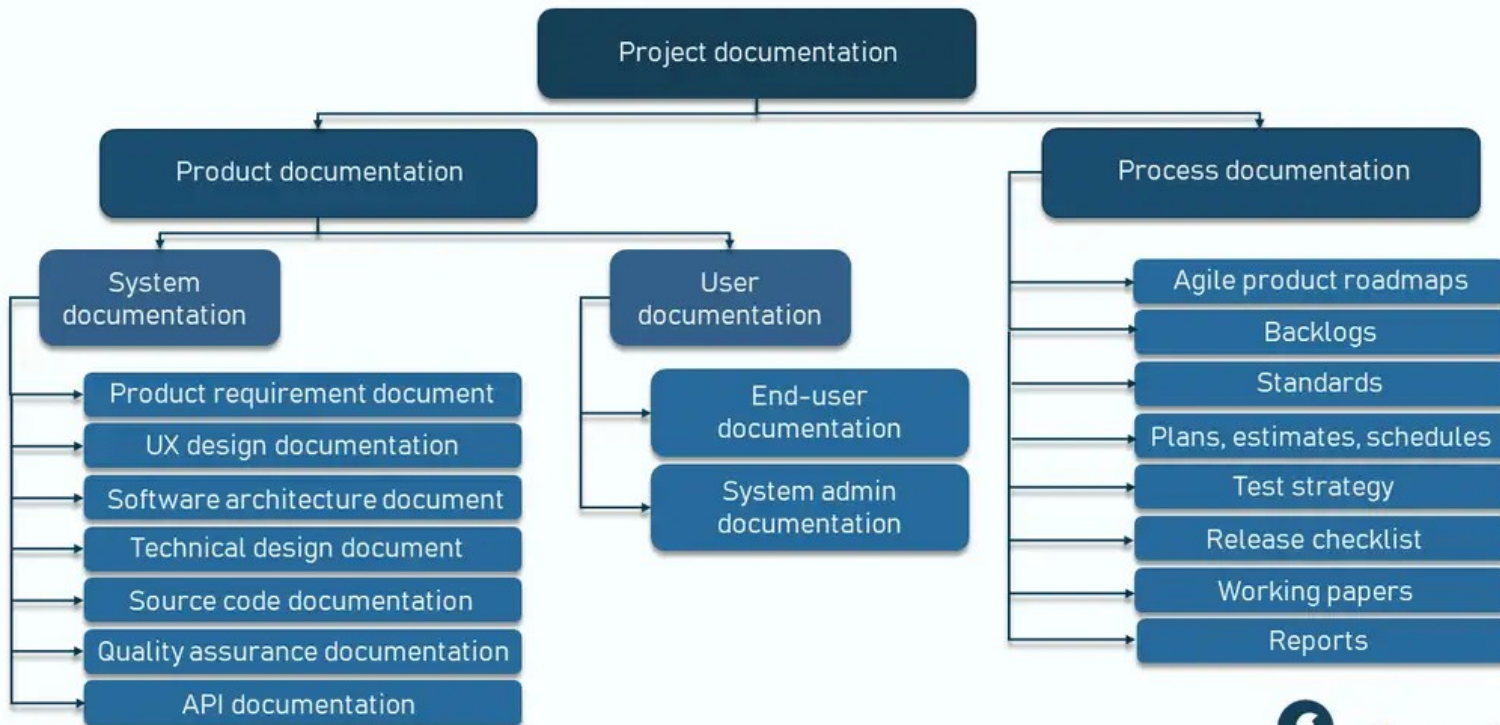
Chapter 2: Software Architecture, <https://waseda.box.com/v/swebok-webinar-c2-architecture>

Chapter 3: Software Design, <https://waseda.box.com/v/swebok-webinar-c3-design>

Chapter 4: Software Construction, <https://waseda.box.com/v/swebok-webinar-c4-construction>



SOFTWARE TECH DOCUMENTATION TYPES



1. Zespół wymagań biznesowych (**BRD**)

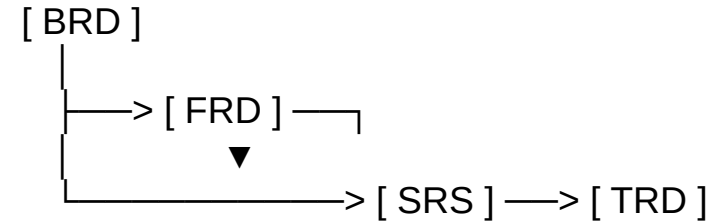
- Zdefiniowanie potrzeb biznesowych i celów projektu

2. Zespół wymagań funkcjonalnych (**FRD**)

- co dokładnie ma robić aplikacja?

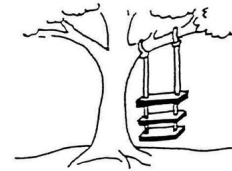
3. Zespół wymagań technologicznych (**TRD**)

- jak to zostanie zaimplementowane?

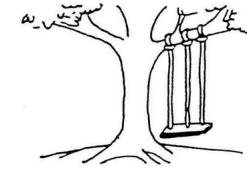


- Brief aplikacji
- ~~Software Requirements Specification (SRS)~~
- Zespół wymagań biznesowych (BRD)
- Zespół wymagań funkcjonalnych (FRD)
- Zespół wymagań technologicznych (TRD)

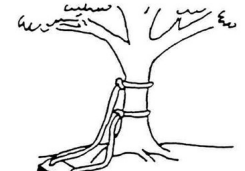
“Problem solving is an art form not fully appreciated by some”



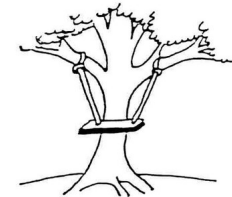
*As proposed by
the project sponsors*



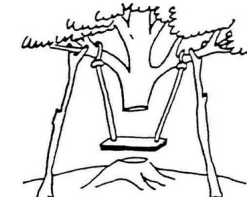
*As specified in
the project request*



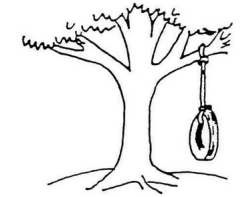
*As designed by
the senior analyst*



*As produced by
the programmers*



*As installed at
the user's site*

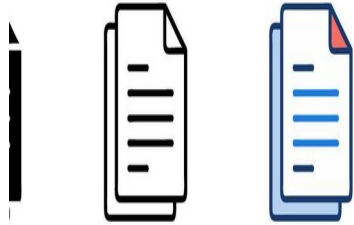


*What the user
wanted*

System Design Tree Swing Metaphor



Brief aplikacji



= BRD + FRD + TRD





Brief aplikacji



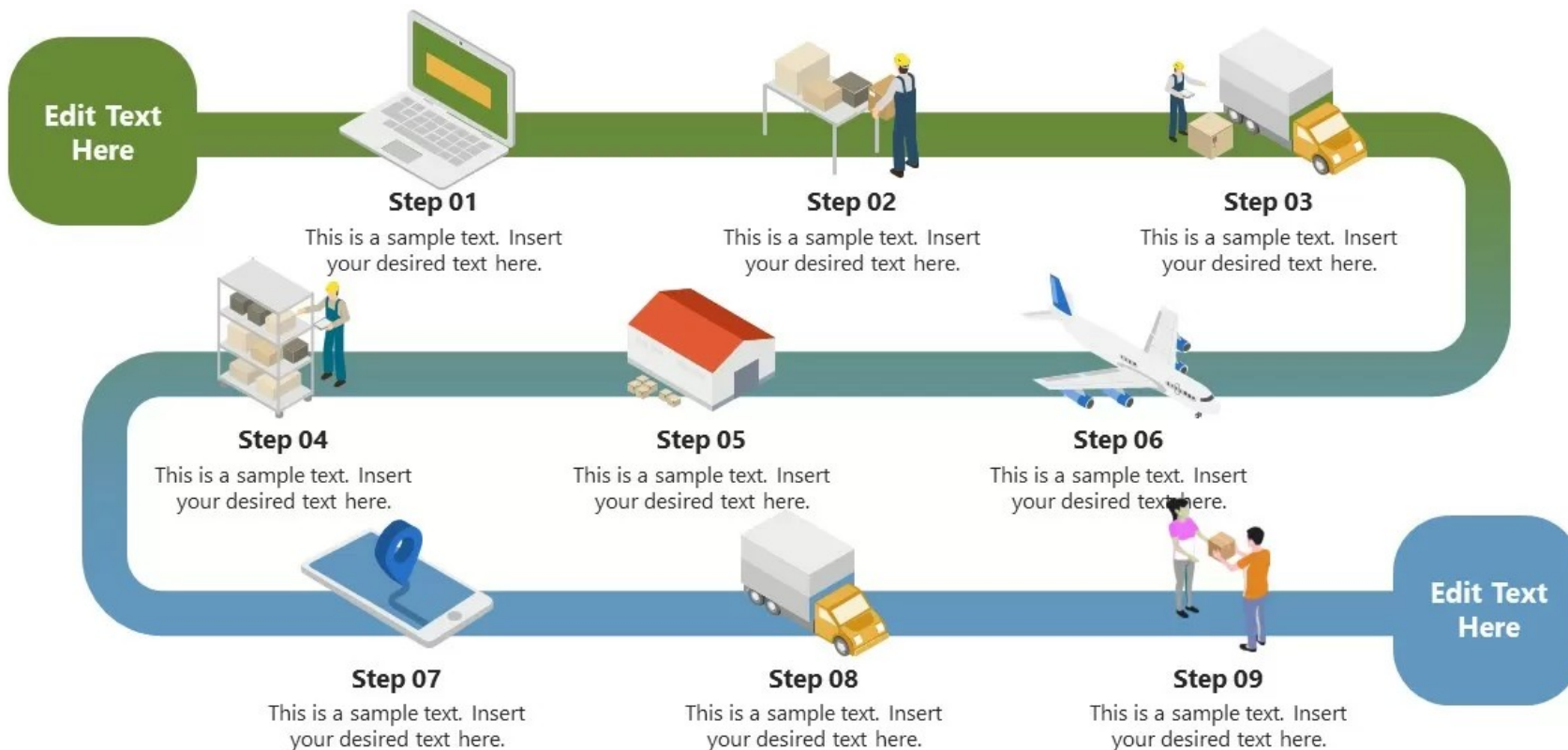
1. Scharakteryzuj grupę docelową aplikacji i opisz przeciętnego użytkownika. W jaki sposób i w jakich okolicznościach aplikacja będzie używana? W jakim kontekście, kiedy i po co? Do czego ma służyć użytkownikowi?
2. Wymień aplikacje konkurencji i dlaczego Twoja jest lepsza.
3. Przewidywany termin realizacji oraz budżet przeznaczony na ten cel.
4. Scharakteryzuj model biznesowy aplikacji. Aplikacja ma być płatna/bezpłatna? Zamierzasz zarabiać na reklamach?
5. W jaki sposób aplikacja będzie promowana? Jaki jest przeznaczony budżet na promocję?



6. Opisz planowaną funkcjonalność aplikacji (2-3 zdania).
7. Określ kolorystykę aplikacji. Jeśli posiadasz grafiki koncepcyjne lub zrzuty ekranu, to załącz je przedstawić.
8. Określ systemy operacyjne, na których ma działać aplikacja.
9. Czy aplikacja będzie komunikować się z zewnętrznymi bazami danych, jakie to będą bazy? Czy aplikacja będzie zależna od innych systemów (logowanie, wysyłka do innych kanałów itd.)? Jeśli tak, to określ punkty integracji.
10. Czy i jaki rozwój aplikacji jest planowany?



Process Flow Diagram to graficzno-opisowa metoda prezentowania procesu produkcyjnego od początku do końca





Analiza wymagań **biznesowych**

Cel: Zdefiniowanie potrzeb biznesowych i celów projektu.

Zawartość:

- Uzasadnienie biznesowe (business case)
- Cele i KPI
- Interesariusze
- Opis problemu i wartości dodanej



BRD - Business Requirements Document

Business requirements document template

Project name: Tech Marketing Blog
Project manager: Sally Brown
Date submitted: January 31, 2022
Document status: Draft Proposed Validated Approved

1. Executive summary
Our tech company plans to start a company marketing blog to educate current users, attract a new audience, and promote products.

2. Project objectives

- Improve client retention by 10% year over year
- Increase website views by 1,000 per month
- Increase conversion rates by 20% on average

3. Project scope
The marketing team will write the content for the blog and will collaborate with the SEO and design team to optimize it. We'll distribute four in-depth posts per month.

4. Business requirements

Priority level	Critical level	Requirement description
1	High	Content scheduling
2	Medium	Website analytics
3	Medium	CRM tracking
4	Medium	SEO Analytics

5. Key stakeholders

Name	Job role	Duties
Sally Brown	Project manager	Head of blog project
Tom Roberts	Department head	Approves blog initiative
Pete Hall	Marketing team member	Ideates/writes blog content
Kristin Watson	SEO team member	Optimizes blog

6. Project constraints

Constraint	Description
Timeline	Complete four posts per month (one post per week)
Budget	Stick to \$2,000 monthly budget
Team availability	Must stick to team member schedules
Project risks	May not receive views if we don't rank in Google

7. Cost-benefit analysis

Cost	Benefit
Team member time	Team members create lasting deliverables
SEO software	Provides insight to rank posts and increase visibility
Blog design/management	Keeps audience on page and increases conversions
Blog software/hosting	Required to run blog

Total cost = \$2,000/month Expected ROI = \$3,000/month

asana

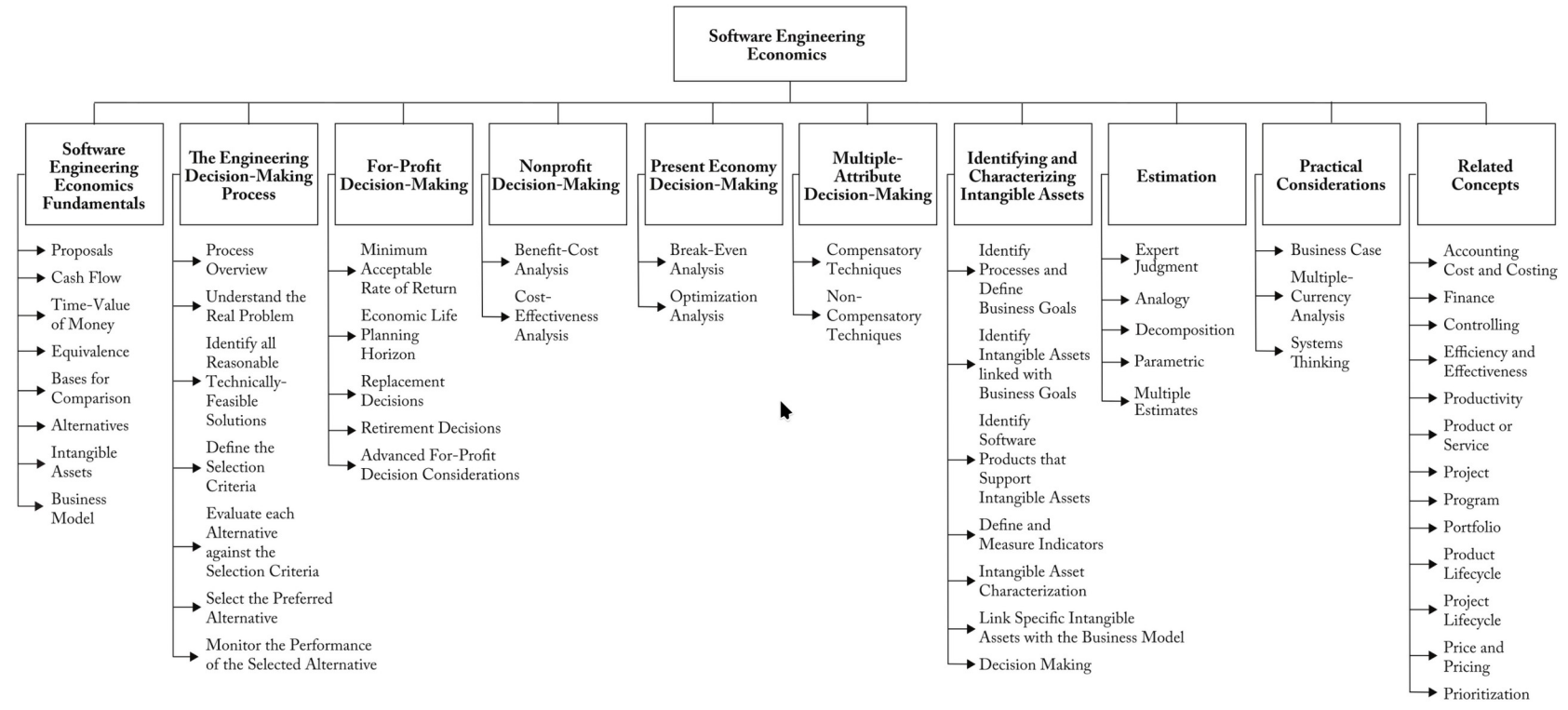


Figure 15.1. Breakdown of Topics for the Software Engineering Economics KA

Wycena jakości uwzględnia tylko wskaźniki jakości typu wartości niematerialnej i oblicza jej ogólną wycenę.

Aby ocenić podzbiór wskaźników jakości, biorąc pod uwagę zestaw q wskaźników jakości dla wartości niematerialnej i prawnej n , wycena jakości jest podana zgodnie z równaniem 1.

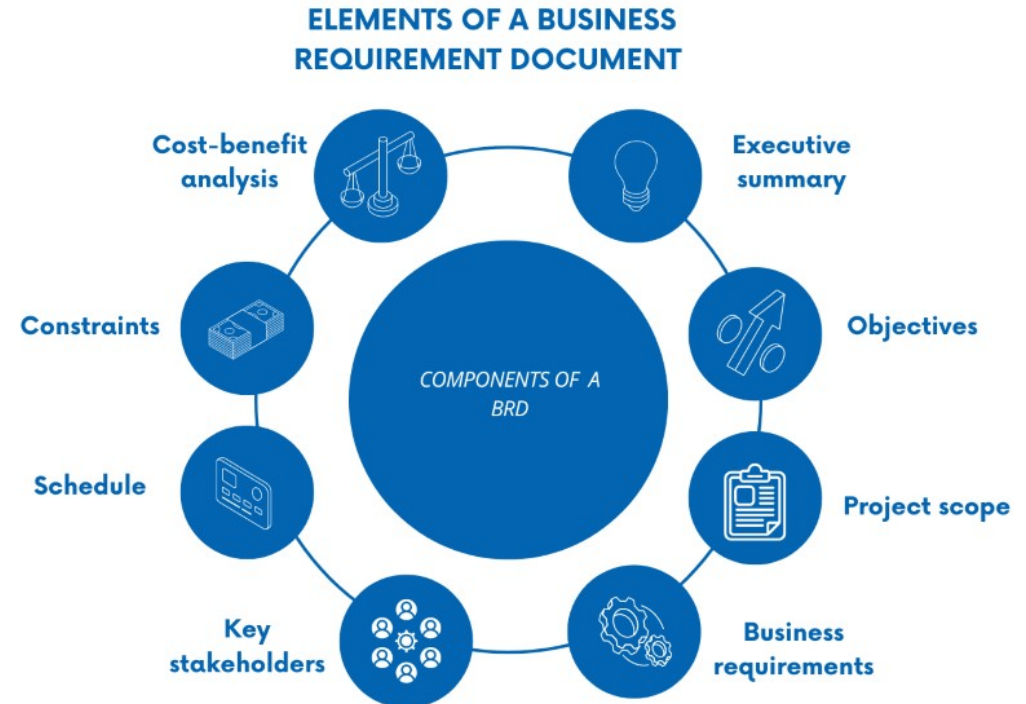
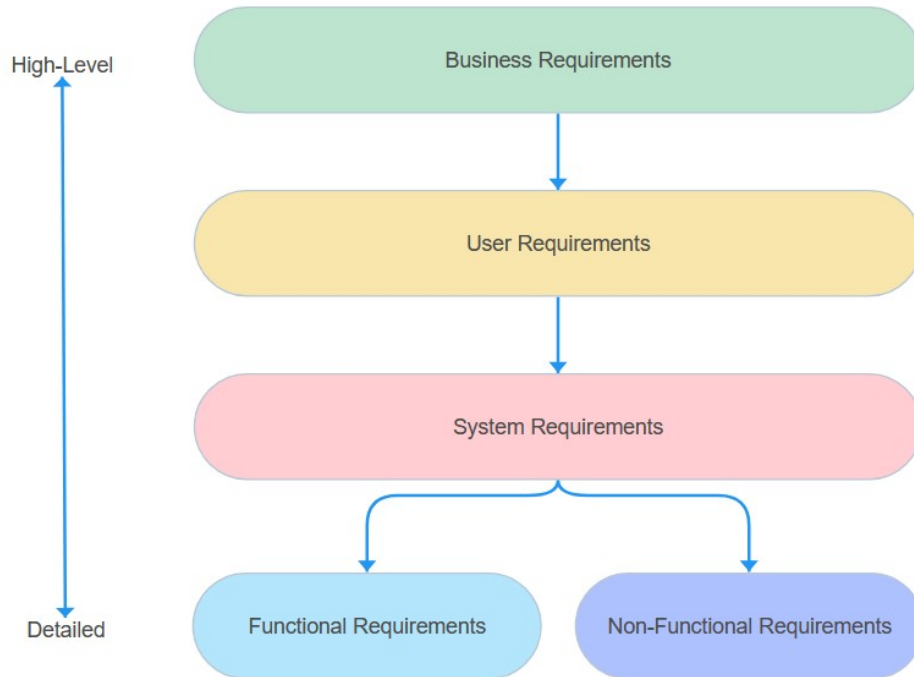
Gdzie X_i^n to każdy z q znormalizowanych wskaźników jakości, które posiada składnik wartości niematerialnych n .

Powyżej, Q_{val} jest średnią znormalizowanych wartości wskaźników jakości odpowiednich konkretnych wartości niematerialnych i prawnych.

$$Q_{Val}^n = \frac{\sum_{i=1}^q X_i^n}{q}$$

Equation 1. Quality Assessment for a Knowledge Asset

BRD - Business Requirements Document





Analiza wymagań funkcjonalnych

FRD - Functional Requirement Document

Analiza wymagań funkcjonalnych umożliwia zidentyfikowanie i opisanie pożądanego zachowania systemu. Zgodnie z jedną z definicji, wymaganie funkcjonalne to „stwierdzenie, jakie usługi ma oferować system, jak ma reagować na określone dane wejściowe oraz jak ma się zachowywać w określonych sytuacjach.

W odróżnieniu od wymagań biznesowych wymagania funkcjonalne odnoszą się bezpośrednio do tego co ma się dziać w aplikacji.

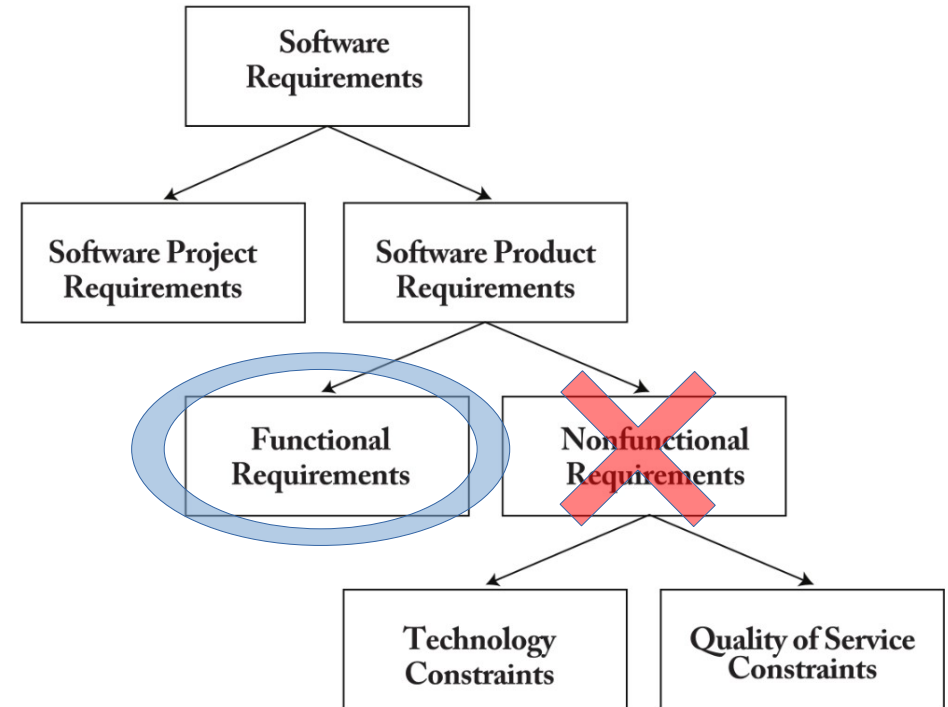
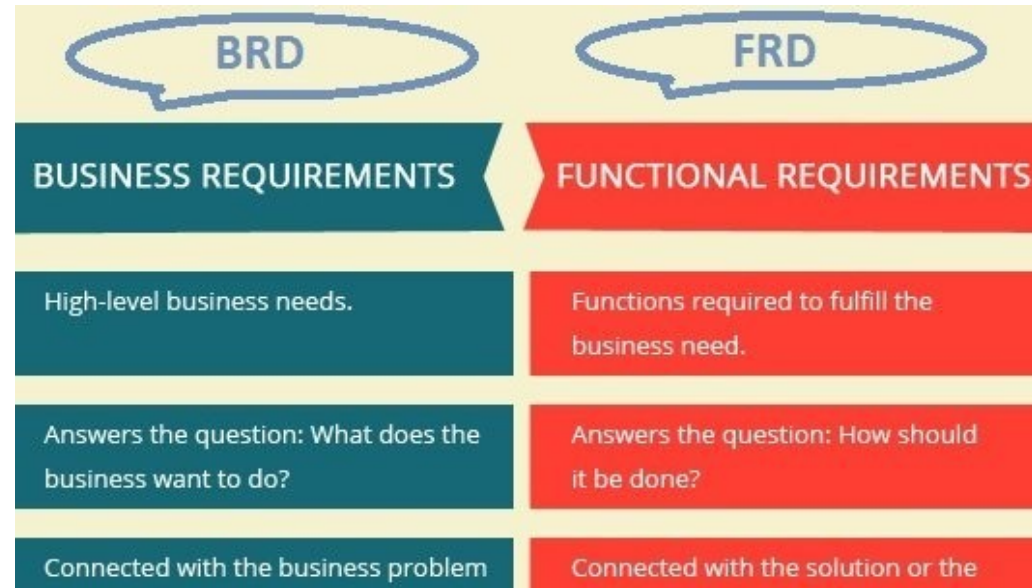


Figure 1.2. Categories of Software Requirements

FRD - Functional Requirement Document

Zawiera uporządkowane procesami wymagania klienta i propozycje ich realizacji (rozwiązania). Dokument używany jest do opisu procesów biznesowych.

Analiza GAP/FIT - Standard dokumentacji analizy wymagań. Dokument w postaci arkusza kalkulacyjnego. Zawiera wymagania i ich rozwiązania określone w dokumencie FRD wraz z oszacowaniem czasochłonności prac.



FRD - Functional Requirement Document

Analiza wymagań funkcjonalnych umożliwia zidentyfikowanie i opisanie pożądanego zachowania systemu. Zgodnie z jedną z definicji, wymaganie funkcjonalne to „stwierdzenie, jakie usługi ma oferować system, jak ma reagować na określone dane wejściowe oraz jak ma się zachowywać w określonych sytuacjach.

W odróżnieniu od wymagań biznesowych wymagania funkcjonalne odnoszą się bezpośrednio do tego co ma się dzieć w aplikacji.

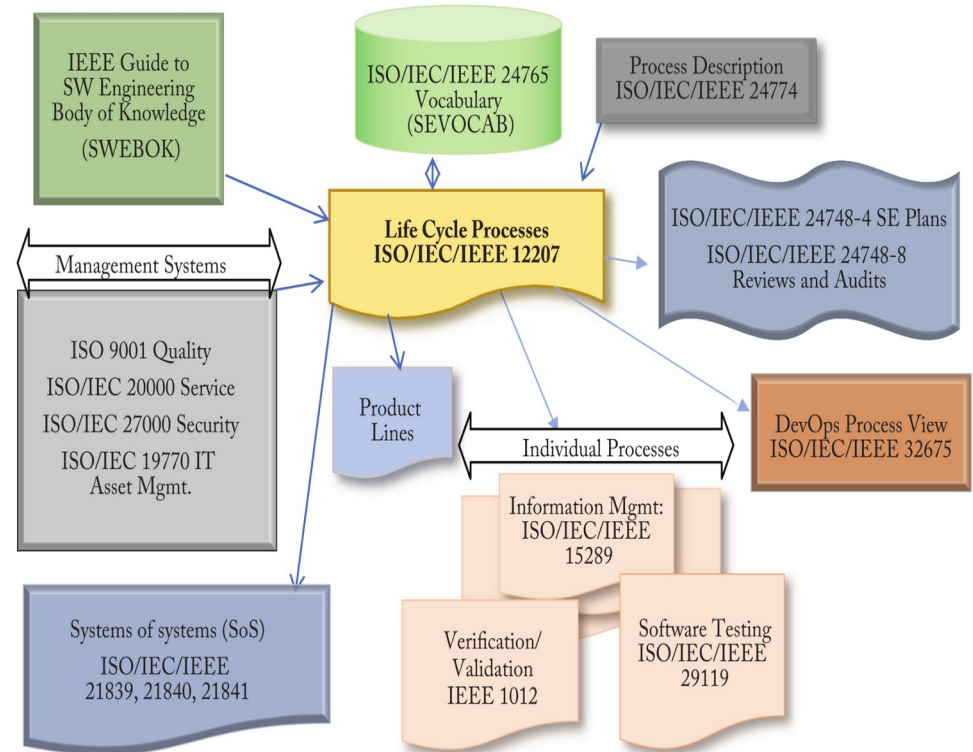
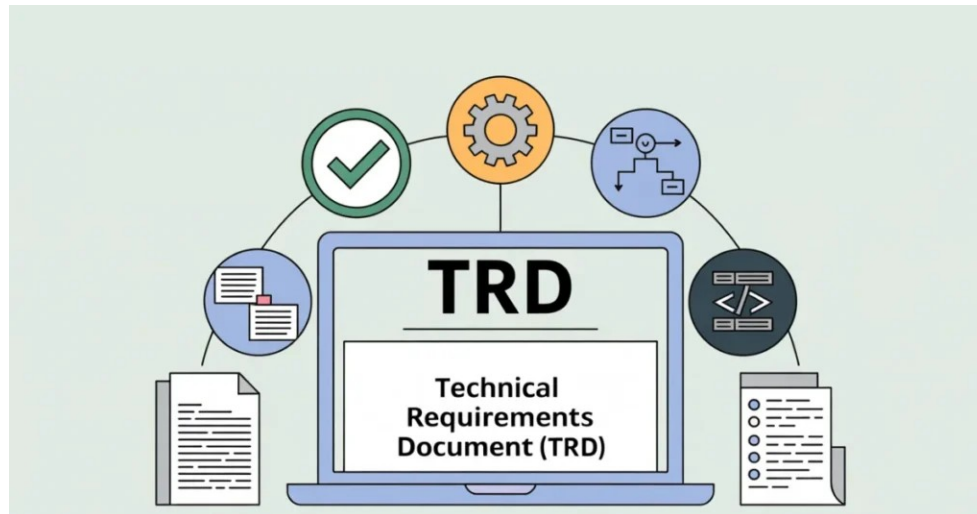


Figure B.1. Software Engineering Standards Landscape



Analiza wymagań technicznych

TRD (*Technical Requirements Document*) to dokument inżynierski definiujący techniczne aspekty realizacji systemu na podstawie wymagań z SRS. Jego celem jest dostarczenie zespołowi deweloperskiemu precyzyjnych wytycznych dotyczących implementacji.



Zawartość TRD (*typowa struktura*)

1. Architektura systemu

- Diagramy komponentów
- Wzorce projektowe
- Podział na moduły/subsystemy

2. Wybór technologii i narzędzi

- Języki programowania
- Frameworki, biblioteki
- Bazy danych, silniki przetwarzania

5. Bezpieczeństwo i wydajność

- Szyfrowanie, autoryzacja
- SLA, czas odpowiedzi, skalowalność

3. Interfejsy systemowe i integracje

- API (wewnętrzne i zewnętrzne)
- Protokoły komunikacji (np. REST, gRPC)
- Format danych (JSON, XML, protobuf)

4. Wymagania sprzętowe i środowiskowe

- Platformy docelowe (iOS, Android, Web)
- Wymagania pamięciowe, obliczeniowe
- Środowiska uruchomieniowe (np. Docker, Kubernetes)

6. Strategie testowania i wdrażania

- I/CD, staging, monitoring
- Strategie rollbacku, testy obciążeniowe



Functional vs. Technical Requirements

Functional Requirements

Focuses on what the system or solution must do to meet business objectives. Describes the features, behaviors, and outcomes expected from the user's perspective.

Technical Requirements

Focuses on how the system will be implemented to achieve the functional requirements. Details the technical specifications, architecture, and infrastructure necessary for the system to function as intended.



BRD: Dlaczego i co — cele biznesowe

FRD: Co dokładnie — funkcje systemu

TRD: Jak — implementacja techniczna



KONIEC