

Wymagania funkcjonalne i нефункционалне

Z punktu widzenia rodzaju wymagań można je podzielić na:

Wymagania funkcjonalne – dotyczące tego co ma realizować system; jakie ma spełniać funkcje, jakich dostarczać usług, jak zachowywać się w określonych sytuacjach.

Wymagania funkcjonalne powinny być:

- kompletne – opisywać wszystkie usługi żądane od systemu
- spójne – nie zawierać stwierdzeń sprzecznych

Zwykle opis wymagań funkcjonalnych powinien zawierać następujące punkty:

- Nazwa funkcji
- Opis – krótki opis funkcjonalności.
- Wejście – definicja danych wejściowych i ich ewentualnych ograniczeń.
- Wyjście – definicja zwracanych rezultatów.
- Wymagania - czynności i funkcje niezbędne do uruchomienia funkcji
- Efekty uboczne – określenie dodatkowych czynności wykonywanych dla tej funkcjonalności, np. interakcji z innymi funkcjami.

Przykład:

Nazwa funkcji	Gotowanie mleka
Opis funkcji	Przygotowanie ugotowanego mleka
Wejście	Mleko
Wyjście	Ugotowane mleko
Wymagania	Garnek, źródło ciepła o odpowiedniej temperaturze
Efekty uboczne	Brudny garnek do umycia, zużycie energii

Wymagania нефункционалне – dotyczące tego jak system powinien realizować swoje zadania; np. wymagania dotyczące koniecznych zasobów, ograniczeń czasowych, niezawodności, bezpieczeństwa, przenośności, współpracy z określonymi narzędziami i środowiskami, zgodności z normami i standardami, a także przepisami prawnymi, w tym dotyczącymi tajności i prywatności, itp. Opisują ograniczenia, przy zachowaniu których system powinien realizować swoje funkcje, np.:

Cecha	Miara
Wydajność	<ul style="list-style-type: none"> • obsługa do 50 transakcji tygodniowo • równoczesny dostęp 50 użytkowników • maksymalny czas reakcji do 5 sekund
Dostępność	<ul style="list-style-type: none"> • system musi być dostępny on-line 24h na dobę, 7 dni w tygodniu • maksymalny czas niedostępności systemu to 1h w ciągu miesiąca

Inne:

Rozmiar	<ul style="list-style-type: none"> • wymaga pamięci RAM
---------	--

Niezawodność	<ul style="list-style-type: none">• Prawdopodobieństwo błędnego wykonania podczas realizacji transakcji• Częstotliwość błędnych wykonań• Średni czas między błędnymi wykonaniami
Odporność	<ul style="list-style-type: none">• Czas restartu po awarii systemu• Prawdopodobieństwo zniszczenia danych w przypadku awarii systemu
Przenośność	<ul style="list-style-type: none">• Procent kodu zależnego od platformy docelowej• Liczba platform docelowych• Koszt przeniesienia na nową platformę

Wymagania niefunkcjonalne dla wielu systemów są co najmniej równie ważne jak wymagania funkcjonalne (np. szybkość działania wyszukiwarki może być równie ważna jak precyzja wyszukiwania).

Niekiedy wymagania niefunkcjonalne na pewnym poziomie szczegółowości stają się wymaganiami funkcjonalnymi na innym poziomie (np. wymagania dotyczące bezpieczeństwa – stopień bezpieczeństwa może przekształcić się w konieczność realizowania pewnych funkcji związanych z bezpieczeństwem).

Wymagania niefunkcjonalne mogą dotyczyć nie tylko końcowego produktu jakim jest oprogramowanie, ale także samego procesu wytwarzania oprogramowania.

Ważnym przykładem specyficznej grupy wymagań są wymagania dotyczące zabezpieczania systemu. Obejmują one wymagania:

- identyfikacji użytkowników
- autentykacji użytkowników
- autoryzacji
- wykrywania ataków
- odporności na ataki (wirusów, robaków, itp.)
- integralności danych (odporność na niszczenie danych)
- uniemożliwiania zaprzeczaniu uczestnictwa w transakcjach
- prywatności
- sprawdzalności zabezpieczeń
- odporności podsystemu zabezpieczeń na zmiany dokonywane w innych częściach systemu

Aby przekształcić zbyt ogólne wymagania niefunkcjonalne stosuje się wiele miar dla różnych kategorii wymagań, np.:

- dla szybkości działania: liczba transakcji na sekundę, czas reakcji na zdarzenie, czas odświeżenia strony
- dla łatwości użycia: czas konieczny na przeszkolenie obsługi, liczba stron (okien) pomocy
- dla niezawodności: średni czas do awarii, prawdopodobieństwo niedostępności

- dla odporności: procent zdarzeń powodujących awarię, prawdopodobieństwo zniszczenia danych w trakcie awarii
- dla przenośności: procent instrukcji zależnych od np. systemu operacyjnego lub bazy danych, liczba uwzględnionych systemów operacyjnych (baz danych)