

STRATEGIE REALIZACJI SYSTEMU INFORMATYCZNEGO

(MODELE CYKLU ŻYCIA SYSTEMU/OPROGRAMOWANIA SYSTEMU)

1. STRATEGIA OPARTA NA DOKUMENTACH

(ang. document driven)

**Bardzo dokładna realizacja w oparciu o model kaskadowy.
Proces podzielony jest na etapy. Każdy etap kończy się
opracowaniem dokumentu.**

Model kaskadowy

Etap1	Określenie wymagań
Etap2	Opracowanie założeń
Etap3	Projektowanie techniczne
Etap4	Realizacja ...

EtapN	Eksploatacja

2. PROTOTYPOWANIE (ang. prototyping)

**Prototyp – ogólny model przyszłego systemu informatycznego,
który w kolejnych iteracjach jest udoskonalany aż do
osiągnięcia akceptowalnego stopnia szczegółowości.**

**Podstawowy cel tworzenia prototypu – weryfikacja i uściślenie
wymagań**

ZALETY:

- wykrycie nieporozumień pomiędzy klientem i projektantami**
- wykrycie brakujących funkcji w projekcie (w systemie)**
- wykrycie trudnych funkcji**
- wykrycie braków w specyfikacji wymagań**

Ponadto:

- możliwość szybkiej demonstracji pracującej wersji systemu**
- możliwość szkoleń zanim zbudowany zostanie cały system.**

3. MONTAŻ Z GOTOWYCH KOMPONENTÓW

(ang. composition of reusable components)

Przyspieszona realizacji systemu na bazie własnych lub kupionych komponentów, modułów, bibliotek programistycznych..

Zalety: wysoka niezawodność, zmniejszenie ryzyka, realizacja wg standardów, efektywne wykorzystanie nie tylko własnych doświadczeń, jednak mimo zakupów zmniejszenie kosztów.

Wady: zwykle brak odpowiednich, w pełni zgodnych z oczekiwaniami komponentów, uzależnienie się od zewnętrznych firm programistycznych i zewnętrznych standardów, dodatkowy koszt w przypadku tworzenia własnych bibliotek.

4. PROGRAMOWANIE WYNALAZCZE

(ang. exploratory programming)

Wymagania do systemu rozpoznawane są w trakcie tworzenia systemu.

5. PROGRAMOWANIE PRZYROSTOWE

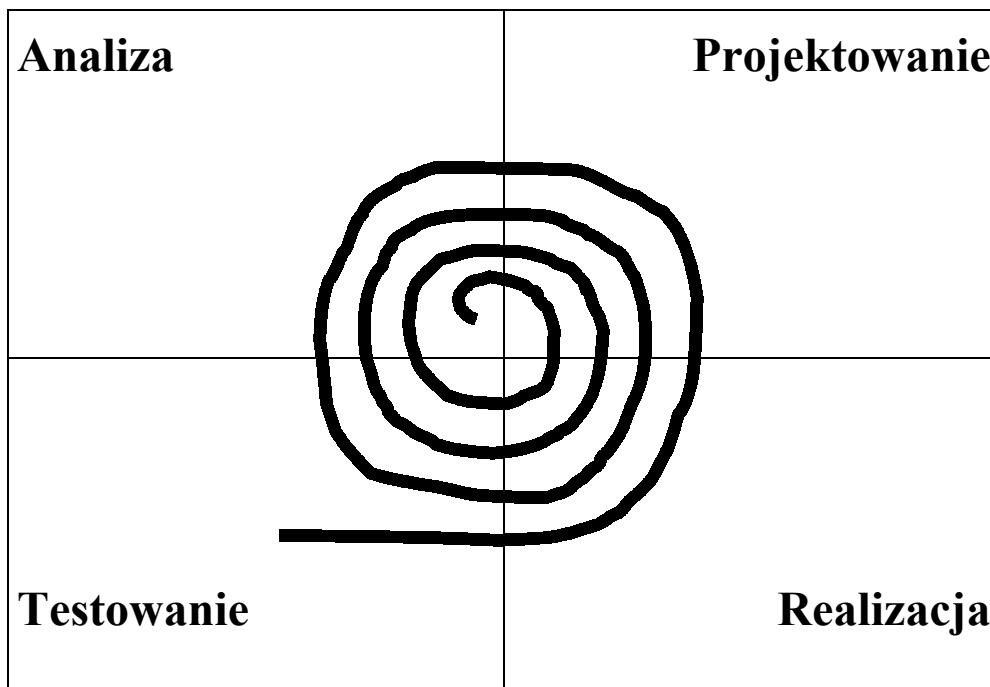
(ang. incremental programming)

Kolejne wymagania rozpoznawane są w trakcie tworzenia systemu i dodawane do sytemu.

Zalety: bardzo szybka możliwość demonstracji pracującej i w pełni sprawnej części systemu, a w związku z tym możliwość szkoleń, a nawet częściowego wdrożenia systemu i użytkowania ograniczonej wersji systemu.

Wada: większe koszty realizacji.

6. REALIZACJA W OPARCIU O MODEL SPIRALNY (ang. spiral model)



7. TRANSFORMACJE FORMALNE (ang. formal transformation)

Rozpoznanie i formalna specyfikacja wymagań.

Postać pośrednia.

Postać pośrednia

.....

Kod informatyczny

Wady: raczej metoda teoretyczna, wąskie zastosowania,
język formalnej specyfikacji jest praktycznie językiem
programowania, mała efektywność.