

INŻYNIERIA OPROGRAMOWANIA

**wykład 3:
FAZA WSTĘPNA**

dr inż. Leszek Grocholski

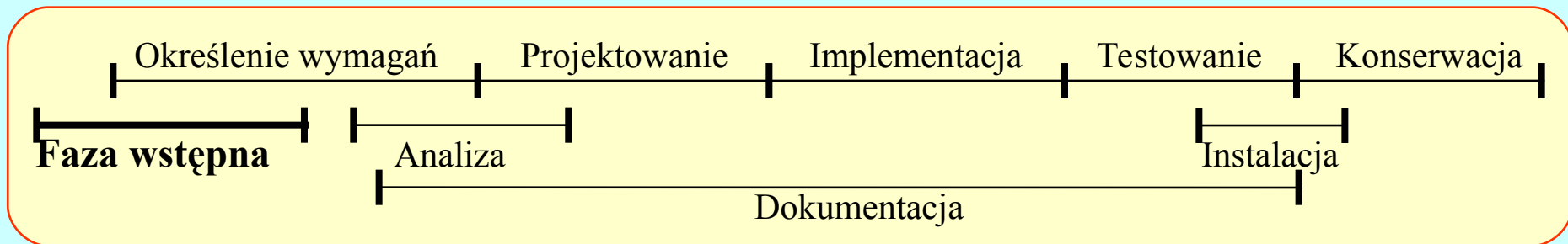
Zakład Inżynierii Oprogramowania
Instytut Informatyki
Uniwersytet Wrocławski

Plan wykładu

- 1. Analiza biznesowa**
- 2. Analiza potrzeb klienta**
- 3. Zakres i kontekst przedsięwzięcia**
- 4. Podejmowane decyzje**
- 5. Harmonogram przedsięwzięcia**
- 6. Szacowanie kosztu oprogramowania**
- 7. Studium wykonalności**

Faza wstępna (studium wykonalności)

feasibility study



Celem FAZY WSTĘPNEJ jest podjęcie decyzji o rozpoczęciu projektu wytworzenia (zamówienia) i wdrożenia oprogramowania.

Nazywana także **strategicznym planem rozwoju informatyzacji (SPRI)** lub **studium osiągalności**.

Czynności w fazie wstępnej

-> strona zamawiającego

PRZEPROWADZENIE ANALIZY BIZNESOWEJ,

jej etapy to:

1. Zidentyfikowanie problemu i jego źródła (problem, potrzeba zmiany)
2. Opisanie działania organizacji po wyeliminowaniu problemu
3. Opis możliwych rozwiązań problemu (nie tylko system informatyczny !)
4. Jeżeli problem ma być rozwiązany przy pomocy systemu informatycznego to:
 - Określenie najważniejszych wymagań
 - Propozycja kilku możliwych rozwiązań (sposobów realizacji systemu)
 - Oszacowanie kosztów oprogramowania i projektu
 - Analiza rozwiązań
 - Prezentacja wyników analizy biznesowej
5. Podjęcie decyzji o zamówieniu (ew. nie zamówieniu) oprogramowania

Czynności w fazie wstępnej

→ strona wykonawcy

1. Analiza co i dlaczego klient potrzebuje

Najlepiej przez dokonanie serii rozmów (wywiadów) z przedstawicielami klienta

2. Określenie celów przedsięwzięcia z punktu widzenia klienta

3. Określenie zakresu oraz kontekstu przedsięwzięcia

4. Ogólne określenie wymagań, wykonanie zgrubnej analizy i koncepcji systemu

5. Propozycja kilku możliwych rozwiązań (sposobów realizacji systemu)

6. Orientacyjne oszacowanie kosztów rozwiązań

7. Analiza rozwiązań

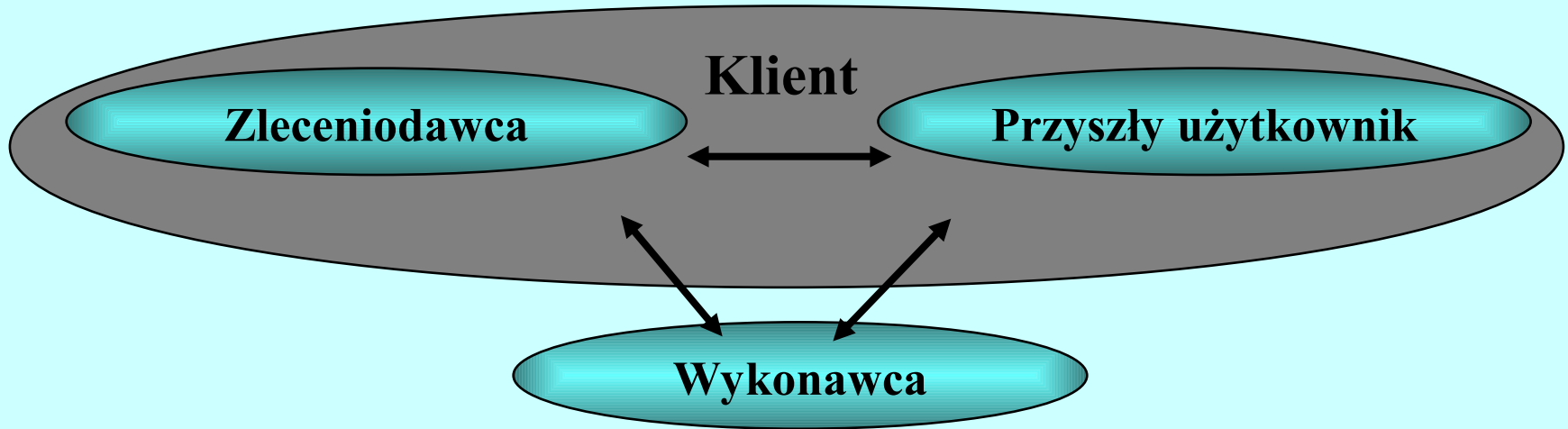
8. Określenie wstępnego harmonogramu przedsięwzięcia oraz struktury zespołu realizatorów

9. Określenie standardów, zgodnie z którymi realizowane będzie przedsięwzięcie

10. Prezentacja wyników fazy wstępnej przedstawicielom klienta oraz korekta wyników

11. Podjęcie decyzji o wytworzeniu oprogramowania dla klienta

Współpraca z klientem



Po stronie klienta wśród interesariuszy warto wyróżnić:

zleceniodawcę (sponsora) i przyszłych użytkowników.

Należy starać się uwzględnić kryteria wielu stron, ale należy pamiętać, że system będzie głównie oceniany przez przyszłych użytkowników.

Ważnym elementem fazy wstępnej jest jasne określenie **celów** przedsięwzięcia z punktu widzenia klienta. Nie zawsze są one oczywiste, co często powoduje nieporozumienia pomiędzy klientem i wykonawcą.

Równie ważne jest określenie **ograniczeń klienta** (np. finansowych, infrastruktury, zasobów ludzkich, czasu wdrożenia, itd.)

Przykład: program podatkowy

Firma rachunkowa zajmuje się m.in. przygotowaniem formularzy zeznań podatkowych (PIT-ów) dotyczących podatku dochodowego dla indywidualnych podatników.

Ponieważ liczba klientów tego rodzaju usługi jest duża, a w dodatku muszą być obsłużeni w większości w marcu i kwietniu, firma widzi konieczność opracowania systemu komputerowego wspomagającego ten typ działalności.

Cele systemu:

Ogólnie świadczenie udoskonalonych usług, tzn w tym przypadku

- przyśpieszenie obsługi klientów,
- zmniejszenie ryzyka popełnienia błędów.

Przykład: system informacji geograficznej - SIG

Firma TURYSTYCZNA widzi możliwość sprzedaży rynkowej prostego systemu informacji geograficznej (mapy komputerowej).

Miałby to być system łączący w sobie możliwość przeglądania bitowej mapy pewnego obszaru (np. mapy fizycznej, zdjęcia satelitarnego) wraz z umieszczonymi na tym tle dodatkowymi informacjami opisującymi pewne obiekty znajdujące się na prezentowanym obszarze.

Cele systemu:

- możliwość łatwego, dialogowego projektowania mapy,
- możliwość łatwego i wygodnego przeglądania mapy wraz z informacjami.

Przykład: system harmonogramowania zleceń

Przedsiębiorstwo farmaceutyczne zleciło wykonanie analizy krytycznych procesów funkcjonowania jednego z wydziałów. Jednym z nich jest harmonogramowanie zleceń, które wydział otrzymuje z działu marketingu. Zlecenie oznacza wyprodukowanie pewnej ilości konkretnego produktu, przy czym możliwe są dodatkowe wymagania, np. ograniczenie terminu wykonania.

Wymagania:

- uwzględnienie wszelkich ograniczeń, zapewniające praktyczną wykonalność proponowanych harmonogramów
- zapewnienie możliwości “ręcznego” modyfikowania harmonogramu
- opracowanie harmonogramu w formie łatwej do wykorzystania przez kadrę kierowniczą wydziału oraz automatyzacja przygotowania zamówień dla magazynu na półprodukty

Cele przedsięwzięcia z punktu widzenia klienta:

- zwiększenie wydajności pracy wydziału,
- zmniejszenie opóźnień w realizowaniu zleceń.

Zakres i kontekst przedsięwzięcia

Zakres przedsięwzięcia: określenie fragmentu procesów informacyjnych zachodzących w organizacji, które będą objęte przedsięwzięciem. Na tym etapie może nie być jasne, które funkcje będą wykonywane przez oprogramowanie, a które przez personel, inne systemy lub standardowe wyposażenie sprzętu.

Kontekst przedsięwzięcia: systemy, organizacje, użytkownicy zewnętrzni, z którymi tworzony system ma współpracować.

Bardzo ważne jest wczesne określenie co ma zostać objęte systemem a co nie.

Przykłady zakresu/kontekstu przedsiębiorstwa

Program podatkowy

Zakresem przedsiębiorstwa jest działalność jednej firmy rachunkowej, która może mieć dowolną liczbę klientów. Pracownik firmy jest jedynym **systemem zewnętrznym**.

System informacji geograficznej

Zakresem przedsiębiorstwa jest projektowanie i przeglądanie prostej mapy komputerowej.

Systemami zewnętrznymi, z którymi system ma współpracować jest projektant mapy i osoba przeglądająca mapę.

System harmonogramowania zleceń

Zakresem przedsiębiorstwa jest funkcjonowanie komórki wydziału obejmującego przygotowanie harmonogramu wykonywania zleceń.

Systemami zewnętrznymi są: system zarządzający zleceniami, System zawierający informacje o technologicznych możliwościach wydziału produkcyjnego, kadra kierownicza.

Decyzje strategiczne - wykonawca

- Wybór modelu, zgodnie z którym będzie realizowane przedsięwzięcie
- Określenie architektury
- Wybór technik stosowanych w fazach analizy i projektowania
- Wybór środowiska (środków) implementacji
- Wybór narzędzia CASE
- Określenie stopnia wykorzystania gotowych komponentów
- Podjęcie decyzji o współpracy z innymi producentami lub zatrudnieniu ekspertów

Zidentyfikowane ograniczenia:

Maksymalne nakłady, jakie można ponieść na realizację przedsięwzięcia

- Dostępny personel
- Dostępne narzędzia
- Ograniczenia czasowe

Po prezentacji wyników dla klienta końcowym wynikiem może być przyjęcie lub odrzucenie oferty twórcy oprogramowania. Faza wstępna jest nieodłączną częścią cyklu produkcji oprogramowania, wobec czego nie powinna być wykonywana na koszt i ryzyko producenta oprogramowania

Studium wykonalności

feasibility study

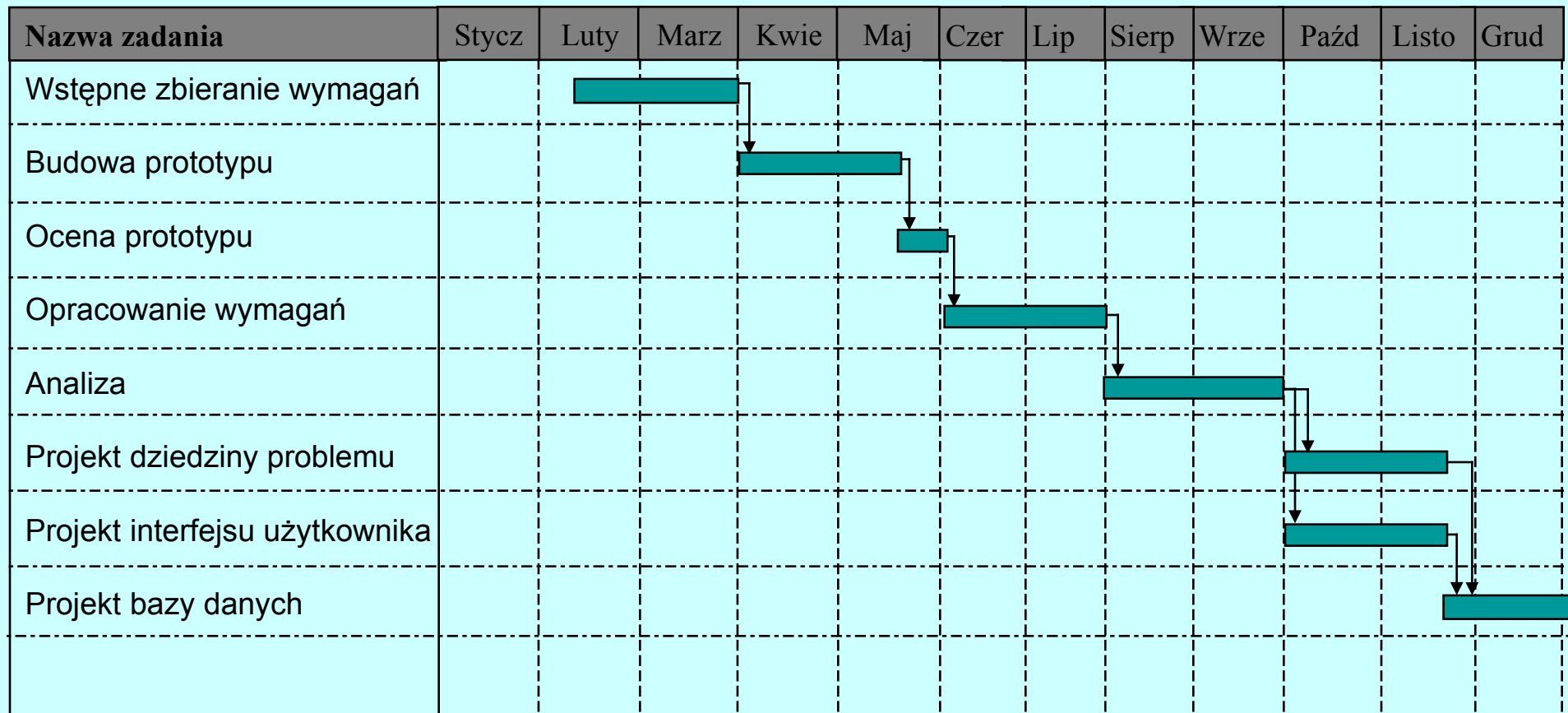
Wynikiem studium wykonalności są m in:

- Rozmiar projektu (np. w punktach funkcyjnych) w porównaniu do rozmiaru zakładanego zespołu projektowego i czasu
- Dostępność zasobów (budżet, personel, kadra)
- Ograniczenia czasowe (krańcowe daty ukończenia projektu, wdrożenia, itd.)
- Warunki wstępne niezbędne do realizacji projektu
- Dostępność oprogramowania oraz narzędzi do rozwoju oprogramowania
- Dostępność sprzętu i sieci
- Dostępność technologii oraz know-how
- Dostępność specjalistów wewnątrz firmy oraz zewnętrznych ekspertów
- Dostępność usług zewnętrznych, kooperantów i dostawców
- Dostępność powierzchni biurowej, środków komunikacyjnych, zaopatrzenia, itd.

Harmonogram przedsięwzięcia

Ustalenie planu czasowego (i/lub wykorzystania zasobów) dla poszczególnych etapów i zadań.

Diagram Gantta



Ocena rozwiązań

Z powodów: wielości celów przedsięwzięcia (czyli kryteriów oceny) lub niepewności (niemożliwości precyzyjnej oceny spodziewanych rezultatów) czy paru możliwych rozwiązań **w fazie wstępnej często rozważa się kilka rozwiązań.**

Częste kryteria oceny:

- koszt
- czas realizacji
- niezawodność
- możliwość ponownego użycia
- przenośność na inne platformy
- wydajność (szybkość)

Prezentacja i porównanie poszczególnych rozwiązań w postaci tabelarycznej

Rozwiązanie	A	B	C
Koszt (tys. zł)	120	80	175
Czas (miesiące)	33	30	36
Niezawodność (błędy/tydzień)	5	9	13
Ponowne użycie (%)	40	40	30
Przenośność (%)	90	75	30
Wydajność(transakcje/sek)	0.35	0.75	1

Wybór rozwiązania

Podczas wyboru optymalnego rozwiązania pomija się rozwiązania zdominowane, tj. gorsze wg danego kryterium (lub wszystkich/prawie wszystkich).

Normalizacja wartości dla poszczególnych kryteriów (sprowadzenie do przedziału $[0,1]$)

Przypisanie wag do kryteriów (również może być trudne).

Przykład: łączna ocena za pomocą sumy ważonej

Rozwiązanie	A	B	C	Waga
Koszt (tys. zł)	0.58	1	0	3
Czas (miesiące)	0.5	1	0	2
Niezawodność (błędy/tydzień)	1	0.5	0	3
Ponowne użycie (%)	1	1	0	1
Przenośność (%)	1	0.75	0	1
Wydajność(transakcje/sek)	0	0,62		11.5
Łączna ocena	7.74	9.17	1.5	

Szacowanie kosztu oprogramowania

Szacowanie kosztów przeprowadza się dla każdego z alternatywnych rozwiązań.

Na koszt oprogramowania składają się następujące główne czynniki:

- robocizna ! → czyli informatycy
- koszt sprzętu będącego częścią tworzonego systemu
- koszt wyjazdów i szkoleń
- koszt zakupu narzędzi

Trzy pierwsze czynniki są dość łatwe do oszacowania.

Oszacowanie kosztów oprogramowania jest praktycznie utożsamiane z oszacowaniem nakładu pracy.

Techniki oszacowania pracochłonności

Modele algorytmiczne. Wymagają opisu przedsięwzięcia przez wiele atrybutów liczbowych i/lub opisowych. Odpowiedni algorytm lub formuła matematyczna daje wynik.

Ocena przez eksperta. Doświadczone osoby z dużą precyzją potrafią oszacować koszt realizacji nowego systemu.

Ocena przez analogię (historyczna). Wymaga dostępu do informacji o poprzednio realizowanych przedsięwzięciach. Metoda polega na wyszukaniu przedsięwzięcia o najbardziej zbliżonych charakterystykach do aktualnie rozważanego i znanym koszcie i następnie, oszacowanie ewentualnych różnic.

Wycena dla wygranej. Koszt oprogramowania jest oszacowany na podstawie kosztu oczekiwanego przez klienta i na podstawie kosztów podawanych przez konkurencję.

Szacowanie wstępujące. Przedsięwzięcie dzieli się na mniejsze zadania, następnie sumuje się koszt poszczególnych zadań.

Modele szacowania kosztów

Historycznie, podstawą oszacowania jest rozmiar systemu liczony w liniach kodu źródłowego. Metody takie są niedokładne, zawodne, sprzyjające patologiom, np. sztucznemu pomnażaniu ilości linii, ignorowaniu komentarzy, itp.

Obecnie stosuje się wiele miar o lepszych charakterystykach (z których będą omówione punkty funkcyjne). Miary te, jakkolwiek niedokładne i oparte na szacunkach, są jednak konieczne. Niemożliwe jest jakiekolwiek planowania bez oszacowania kosztów. Miary dotyczą także innych cech projektu i oprogramowania, np. czasu wykonania, jakości, niezawodności, itd.

Jest bardzo istotne uwolnienie się od religijnego stosunku do miar, tj. traktowanie ich jako obiektywnych wartości “policzonych przez komputer”. Podstawą wszystkich miar są szacunki, które mogą być obarczone znacznym błędem, nierzadko o rząd wielkości. Miary należy traktować jako latarnię morską we mgle - może ona nas naprowadzić na dobry kierunek, może ostrzec przed niebezpieczeństwem. Obowiązuje zasada patrzenia na ten sam problem z wielu punktów widzenia (wiele różnych miar) i zdrowy rozsądek.

Metoda COCOMO

*CO*nstructive *CO*st *MO*del

COCOMO jest oparte na kilku formułach pozwalających oszacować całkowity koszt przedsięwzięcia na podstawie oszacowanej liczby linii kodu. Jest to główna słabość tej metody, gdyż:

- liczba ta staje się przewidywalna dopiero wtedy, gdy kończy się faza projektowania architektury systemu; jest to za późno;
- pojęcie “linii kodu” zależy od języka programowania i przyjętych konwencji;
- pojęcie “linii kodu” nie ma zastosowania do nowoczesnych technik programistycznych, np. programowania wizyjnego.

COCOMO oferuje kilka metod

- **Metoda podstawowa:** prosta formuła dla oceny osobo-miesiący oraz czasu potrzebnego na całość projektu.
- **Metoda pośrednia:** modyfikuje wyniki osiągnięte przez metodę podstawową poprzez odpowiednie czynniki, które zależą od aspektów złożoności.
- **Metoda detaliczna:** bardziej skomplikowana, ale jak się okazało, nie dostarcza lepszych wyników niż metoda pośrednia.

Metoda punktów funkcyjnych

*Function Point Analysis,
FPA*

Metoda punktów funkcyjnych oszacowuje koszt projektu na podstawie funkcji użytkowych, które system ma realizować. Stąd wynika, że metoda ta może być stosowana dopiero wtedy, gdy funkcje te są z grubsza znane.

Metoda jest oparta na zliczaniu ilości wejść i wyjść systemu, miejsc przechowywania danych i innych kryteriów. Te dane są następnie mnożone przez zadane z góry wagi i sumowane. Rezultatem jest liczba „punktów funkcyjnych”.

Punkty funkcyjne mogą być następnie modyfikowane zależnie od dodatkowych czynników złożoności oprogramowania.

Istnieją przeliczniki punktów funkcyjnych na liczbę linii kodu, co może być podstawą dla metody COCOMO.

Metoda jest szeroko stosowana i posiada stosunkowo mało wad. Niemniej, istnieje wiele innych, mniej popularnych metod, posiadających swoich zwolenników.

Metoda Delphi i inne metody

Metoda Delphi zakłada użycie kilku niezależnych ekspertów, którzy nie mogą się ze sobą w tej sprawie komunikować i naradzać. Każdy z nich szacuje koszty i nakłady na podstawie własnych doświadczeń i metod. Eksperti są anonimowi. Każdy z nich uzasadnia przedstawione wyniki.

Koordynator metody zbiera wyniki od ekspertów. Jeżeli znacznie się różnią, wówczas tworzy pewne sumaryczne zestawienie (np. średnią) i wysyła do ekspertów dla ponownego oszacowania. Cykl jest powtarzany aż do uzyskania pewnej zgody pomiędzy ekspertami.

Metoda analizy podziału aktywności (activity distribution analysis):

Projekt dzieli się na aktywności, które są znane z poprzednich projektów. Następnie dla każdej z planowanych aktywności ustala się, na ile będzie ona bardziej (lub mniej) pracochłonna od aktywności już wykonanej, której koszt/nakład jest znany. Daje to szacunek dla każdej planowanej aktywności. Szacunki sumuje się dla uzyskania całościowego oszacowania.

Metody oszacowania pracochłonności: testowania systemu, dokumentacji, wdrożenia, ...

Kluczowe czynniki sukcesu

Szybkość pracy. Szczególnie w przypadku firm realizujących oprogramowanie na zamówienie, opóźnienia w przeprowadzeniu fazy wstępnej mogą zaprzepaścić szansę na wygranie przetargu lub na następne zamówienie. Faza ta wymaga więc stosunkowo niedużej liczby osób, które potrafią wykonać pracę w krótkim czasie.

Zaangażowanie kluczowych osób ze strony klienta. Brak akceptacji dla sposobu realizacji przedsięwzięcia ze strony kluczowych osób po stronie klienta może uniemożliwić jego przyszły sukces.

Uchwycenie (ogólne) całości systemu. Podstawowym błędem popełnianym w fazie wstępnej jest zbyt przywiązanie i koncentracja na pewnych fragmentach systemu. Niemożliwe jest w tej sytuacji oszacowanie kosztów wykonania całości. Łatwo jest też przeoczyć szczególnie trudne fragmenty systemu.

Rezultaty fazy wstępnej

Udostępniony klientowi raport, ew. wewnętrzna analiza, która obejmuje:

- definicję celów przedsięwzięcia
- opis zakresu przedsięwzięcia
- opis systemów zewnętrznych, z którymi system będzie współpracować
- ogólny opis wymagań
- ogólny model systemu
- opis proponowanego rozwiązania
- oszacowanie wymiarów (czasu, kosztów)
- opis metody zarządzania przedsięwzięciem
- wstępny harmonogram prac

Raport oceny rozwiązań, zawierający informację o rozważanych rozwiązaniach oraz przyczynach wyboru jednego z nich.

Opis wymaganych zasobów - pracownicy, oprogramowanie, sprzęt, lokale, ...

Definicje wymaganych standardów.

Ogólny harmonogram.

INŻYNIERIA OPROGRAMOWANIA

Dziękuję za uwagę