

Metodologia

- Brak spójnej i szeroko akceptowanej metodyki
- Proces RUP określa ogólny schemat przebiegu projektu
- Język UML definiuje notację i zestaw modeli

1. Analiza dziedziny problemu

(faza rozwinięcia)

Cel:

- pozyskanie wiedzy na temat dziedziny aplikacji
- poznanie i udokumentowanie wymagań użytkownika

Metoda:

- studiowanie specjalistycznej literatury
- udziału w prezentacjach i rozmowach z ekspertami
- wywiady z przyszłymi użytkownikami

Wyniki:

- dokumentacja regulacji prawnych
- określenie sprzęgów zewnętrznych
- **model przypadków użycia**

2. Wyodrębnienie funkcji systemu

Cel:

- wyodrębnienie działań wspieranych przez system
- określenie interfejsu użytkownika

Metoda:

- dalsza analiza dziedziny aplikacji i wymagań użytkownika
- analiza scenariuszy przypadków użycia
- implementacja prototypu interfejsu użytkownika

Wyniki:

- specyfikacja funkcji systemu
- prototyp interfejsu użytkownika

3. Opracowanie modelu dziedziny problemu.

(wykonywane równolegle z określaniem funkcji)

Cel:

- identyfikacja i klasyfikacja podstawowych bytów dziedziny
- określenie zachowania kluczowych obiektów

Metoda:

- dalsza analiza dziedziny aplikacji i wymagań użytkownika
- analiza scenariuszy przypadków użycia
- wykorzystanie pojęć opisywanych w literaturze
- językowa analiza opisu

Wyniki:

- diagramy klas (model pojęciowy)
- diagramy stanu

4. Analiza wybranego fragmentu systemu.

(faza konstrukcji)

Proces RUP nie dzieli fazy konstrukcji na analizę i projekt — przeciwnie, traktuje łącznie czynności analizy i projektu

Cel:

- określenie sposobu działania systemu

Metoda:

- weryfikacja i uzupełnienie diagramu klas
- określenie logicznego modelu bazy danych
- budowa modelu zachowania (diagramy współpracy)

Wyniki:

- diagramy klas (model specyfikacyjny)
- diagramy współpracy

5. Projektowanie wybranego fragmentu systemu.

(przeplata się z czynnościami analizy)

Cel:

- określenie budowy systemu

Metoda:

- modyfikacja modelu klas
- określenie fizycznego modelu bazy danych
- określenie fizycznej struktury programów (komponenty)

Wyniki:

- diagramy klas (model implementacyjny)
- diagramy komponentów i rozmieszczenia

6. Implementacja wybranego fragmentu systemu.

Cel:

- opracowanie programów

Metoda:

- definicja klas programu
- wykorzystanie technologii systemowych
- wykorzystanie narzędzi CASE

Wyniki:

- kod programu
- dokumentacja techniczna
- podręczniki użytkownika