



Technologie obiektowe

Obiektowe projektowanie systemów

Wprowadzenie do podstawowych elementów inżynierii oprogramowania tak szybkie jak zupka chińska



Cykl życia oprogramowania w skrócie

Planowanie

Analiza

Projektowanie

Implementacja

Testowanie

Wdrożenie

Utrzymanie



Analiza

Rozpoznanie problemu

Dekompozycja

Określenie wymagań

Określenie zachowań

Model logiczny

Model fizyczny



Analiza: przykład

Rozpoznanie problemu: Kantor wymiany walut

Dekompozycja: dostęp do danych, logika, prezentacja danych

Określenie wymagań: automatyczne pobieranie danych, przeliczanie dowolnych walut, wykorzystanie graficznego interfejsu użytkownika

Określenie zachowań: akwizycja danych, parsowanie, budowa zbioru, pobór danych od użytkownika, wykonanie operacji, zwrot wyników

Model logiczny: obiekty zewnętrzne systemu - repozytorium danych, obiekty wewnętrzne systemu: warstwy funkcjonalne, zależności między nimi



Analiza obiektowa

Warstwowość dostosowana do poziomów abstrakcji

Ukrywanie informacji

Abstrakcja

Hierarchia klas

Dziedziczenie

Ponowne wykorzystanie kodu

Polimorfizm



Analiza obiektowa c.d.

Spójność modelu

Hermetyzacja

Grupowanie obiektów w klasy

Komunikacja między obiektami



Analiza obiektowa c.d.

Co program ma wykonywać?

W jakich ograniczeniach ma działać?

Budowa modelu logicznego

Logika biznesowa

Proces biznesowy

Faza modelowania



Proces biznesowy

Zbiór działań

Posiada określony początek i koniec (warunki)

Dostarcza usługi lub produktu



Proces biznesowy c.d.

Wewnętrzny

Zewnętrzny

Najważniejsze: procesy zewnętrzne (biznesowe), użytkownik-przedsiębiorstwo, przynosi korzyści

Zrozumienie potrzeb użytkownika



Projektowanie

Architektury

Interfejs (użytkownika)

Komunikacja

Bezpieczeństwo

Prototyp



Projektowanie: architektura

Wybór architektury systemu

Podział na elementy funkcjonalne: warstwy, moduły, komponenty, etc...

Grupowanie funkcjonalności

Określenie zależności

Określenie interfejsów (między elementami systemu)

High level design



Projektowanie: architektura c.d.

Określenie klas

Określenie struktur tabel

Określenie szczegółów interfejsów

Identyfikacja wszystkich problemów

Wybór języka programowania

Low level design



Projektowanie: Komunikacja

Sposób przekazywania informacji

REST API

Sockets

etc...



Projektowanie: Bezpieczeństwo

Szyfrowanie

Kontrola danych

Ochrona haseł

Podpis cyfrowy

etc.



Implementacja

Budowanie kodu źródłowego

JAVA

C#

Python

PHP

SQL

etc...



Testowanie

Testy jednostkowe

Testy walidacyjne

Testy integracyjne

Testy penetracyjne

etc...

Poprawa implementacji i ponowne testowanie



Wdrażanie

Przygotowanie dokumentacji

Konfiguracja infrastruktury technicznej

Instalacja

Migracja

Testowanie

Oddanie do użytku



Utrzymanie

Konserwacja

Usuwanie usterek

Aktualizacje

Kopie zapasowe



Analiza przypadku

Kantor wymiany walut



Kantor wymiany walut

Rozpoznanie problemu

Dekompozycja

Określenie wymagań

Określenie zachowań

Model logiczny



Projektowanie

Architektury

Interfejs (użytkownika)

Komunikacja

Bezpieczeństwo

Prototyp



Narzędzia

Notacje

UML

Diagramy klas

Diagramy obiektów

Diagramy sekwencji

Diagramy przypadków użycia

Diagram czynności

Diagram komponentów



Narzędzia

RAD - Rapid Application Development

Visual Studio

Embarcadero RAD Studio

IDE

IntelliJ IDEA, NetBeans, Pycharm, DevCpp, Eclipse