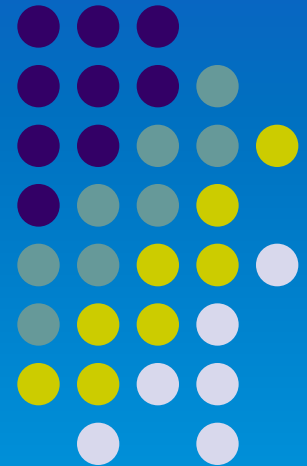


Projektowanie obiektowe

Wzorce projektowe



Wprowadzenie do
wzorców projektowych



Zagadnienia



- Katalog wzorców projektowych wg „Gang of Four”
- Zasady projektowania obiektowego
 - SOLID
- MVC - Model-Widok-Kontroler

Forma zaliczenia i konsultacje



- **Przedmiot kończy się egzaminem**
 - test (c.a. 20 pytań) jednokrotnego i wielokrotnego wyboru (może pojawić się również pytanie opisowe)
 - 3 terminy w sesji (+ ewentualnie zerówka)
- **Strona www – coach.kis.p.lodz.pl**

Roadmap

- Wzorce projektowe
- Schematy opisu wzorców
- Cechy wzorców



Wzorce



- „Każdy wzorzec opisuje pewien regularnie napotykaną problem i łączy go z ogólnym opisem jego rozwiązania w sposób, który pozwala stosować to rozwiązanie miliony razy, ale za każdym razem nieco inaczej.”

Christopher Aleksander

Wzorce projektowe – the beginning



- **Model-View-Controller (MVC) jako szkielet dla języka Smalltalk (T. Reenskauga – początek lat 80-tych) w przypadku aplikacji Webowych nazywany Model-2**
- **„Gang of Four”: Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides**
 - **„Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku”**
(„Design Patterns - Elements of Reusable Object-Oriented Software”, Addison Wesley 1995)

Opis wzorca wg. „Gang of Four” (GoF)



- Nazwa – używana w celu jednoznacznej identyfikacji, ujednolicenia języka,
- Problem – opis problemu, który może być rozwiązany, lista warunków, które muszą być spełnione, aby stosowanie wzorca miało sens
- Rozwiązanie – elementy składające się na projekt, ich związki, zobowiązania i współpraca, nie opisuje konkretnego projektu – stanowi pewien szablon
- Konsekwencje – zyski i straty, wady i zalety stosowania wzorca.

Zalety stosowania wzorców



- Wynikają z wielu praktycznych doświadczeń.
 - Wzorce projektowe ustalają terminologię
 - Ułatwia komunikację z innymi projektantami i programistami
 - Narzuca określoną terminologię projektowania
 - Upraszczają restrukturyzację istniejących systemów.
 - Umożliwiają wielokrotne użycie sprawdzonych rozwiązań.
-
- Ale ...
 - Wzorzec projektowy to półprodukt
 - Muszą zostać obrobione i osadzone w całości projektu

Wzorce GoF



- **Materiałów w Internecie jest sporo:**

- <http://www.vincehuston.org/dp/>
- <http://hillside.net/patterns/patterns-catalog>
- [http://en.wikipedia.org/wiki/Design_pattern_\(computer_science\)](http://en.wikipedia.org/wiki/Design_pattern_(computer_science))

The Sacred Elements of the Faith

the holy origins

the holy behaviors

the holy structures

107 FM Factory Method								139 A Adapter
117 PT Prototype	127 S Singleton				223 CR Chain of Responsibility	163 CP Composite		175 D Decorator
87 AF Abstract Factory	325 TM Template Method	233 CD Command	273 MD Mediator	293 O Observer	243 IN Interpreter	207 PX Proxy	185 FA Façade	
97 BU Builder	315 SR Strategy	283 MM Memento	305 ST State	257 IT Iterator	331 V Visitor	195 FL Flyweight	151 BR Bridge	

Pierwszym waszym zadaniem jest zapoznanie się z tymi wzorcami

Klasyfikacja wzorców wg. GoF



- Strukturalne
 - Łączenie istniejących obiektów
 - Adapter, Decorator, Facade, Composite, Bridge, Proxy, Flyweight.
- Behavioralne
 - Umożliwienie elastycznej zmiany zachowania
 - Interpreter, Iterator, Chain of responsibility, Mediator, Template method, Observer, Visitor, Memento, Command, State, Strategy.
- Kreatywne
 - Ułatwienie procesu tworzenia obiektów
 - Builder, Abstract factory, Factory method, Prototype, Singleton.

Inne wzorce



- Wzorce zaczynają się i nie kończą się na GoF
 - **Wszystkie wzorce bazują na pewnych fundamentach obiektowości**
 - Dziedziczenie i polimorfizm
 - Intefejsy
 - Delagacja
 - **Istnieją również wzorce:**
 - Współbieżności, (np. Active Object, Thread Specific Storage, Thread Pool Pattern, Monitor Object,...)
 - Architektoniczne (SOA, Client-Server, Three-tier, Pipeline, ...),
 - Specyficzne dla konkretnej dziedziny zastosowań (Active Record, Domain Model, Metadata mapping, ...)
 - ...

„Software framework” czyli wzorce
zaimplementowane w projekcie powtórnego użycia



- **Software framework - (rama projektowa, szkielet)**
- struktura wspomagająca tworzenie, rozwój i testowanie powstającej aplikacji
 - .NET Framework
 - Spring Framework
 - Ruby on Rails
 - Eclipse Framework, NetBeans Framework
 - Symfony Framework
 - Struts
 - Hibernate
 - ...

Nasza literatura:



- **Podstawowa:**

- **Oczywiście GoF jako odnośnik do podstawowych wzorców**
- Steven John Metsker , „C#. Wzorce projektowe”, Helion, 2005
- Martin Fowler, „Architektura systemów zarządzania przedsiębiorstwem. Wzorce projektowe”, Helion, 2005.

- **Dodatkowa:**

- Rod Johnson, Juergen Hoeller, Alef Arendsen, Thomas Risberg, Colin Sampaleanu, „Spring Framework. Profesjonalne tworzenie oprogramowania w Javie”, Helion 2006
- Joshua Kerievsky, „Refaktoryzacja do wzorców projektowych”, Helion 2005.
- Alan Shalloway, James R. Trott, „Projektowanie zorientowane obiektowo. Wzorce projektowe. Wydanie II”, Helion 2005