

# Inżynieria Oprogramowania

## „Projekt programu do zarządzania sterowaniem procesami w drukarni”



Opracowane przez:  
Alicja Skoncej  
Piotr Szczodrok  
Mateusz Krusiński

# Cel projektu

Celem projektu było opracowanie systemu wspomagającego zarządzanie zleceniami w drukarni. System ma za zadanie przechowywać informacje na temat pracowników, zleceń oraz ich statusów.



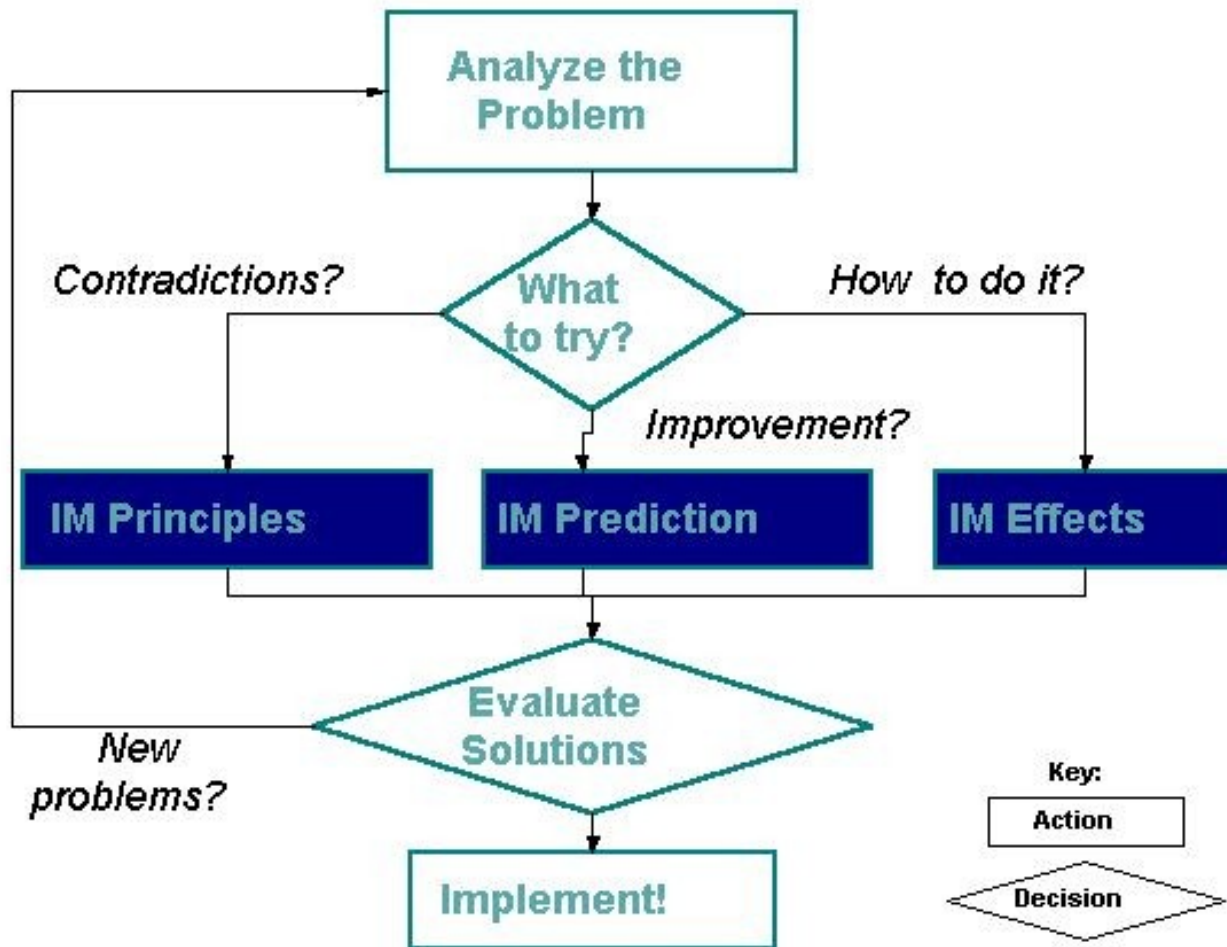
# Wykorzystane narzędzia:

- Visual Paradigm (12.2 Standard Edition)
- QT 5.5
- GitHub



# I etap – specyfikacja

- Przeanalizowanie problemu
- Określenie i ustalenie wymagań systemowych



# II etap - projektowanie

- Ustalenie ogólnej architektury systemu
- Analiza tekstowa:
  - Wstępny opis systemu
  - Wyszczególnienie aktorów
  - Utworzenie słownika pojęć



**CREATING  
CONNECTIONS**

# II etap - projektowanie

- Diagramy przypadków użycia (Use case diagram)
- Przypadki użycia pokazują, co powinien robić system
- Określają wymagania funkcjonalne systemu.
- Wychwytuje fragment funkcji udostępnianych przez system.



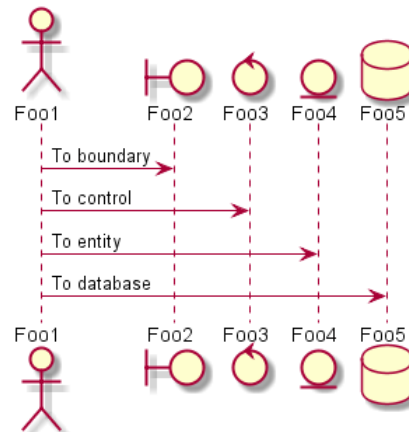


# II etap - projektowanie

- Diagramy aktywności (activity diagram) - umożliwiają określenie tego, w jaki sposób system będzie osiągał swoje zamierzone cele:
  - Jakie akcje?
  - Jak te akcje są połączone?
- Diagramy aktywności z zasady nie pokazują wszystkich szczegółów przetwarzania.
- Pokazują aktywności bez pokazywania bytów, realizujących daną aktywność i dlatego z reguły używane są jako punkt startowy dla procesu modelowania zachowań

# II etap - projektowanie

- Diagramy sekwencji - opisują interakcje pomiędzy częściami systemu w postaci sekwencji komunikatów wymienianych między nimi

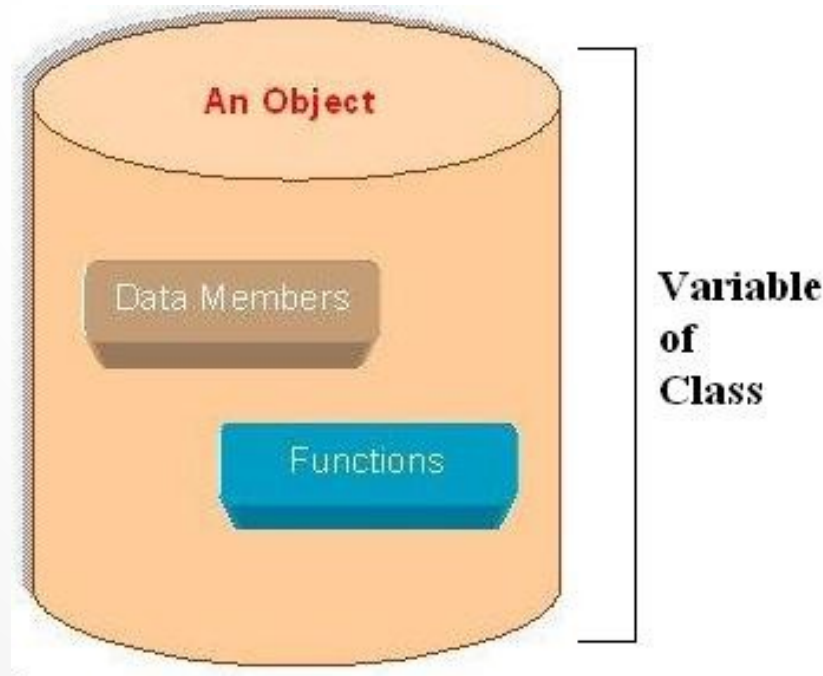


- Obrazuje kolejność w czasie wysyłania komunikatów pomiędzy różnymi obiektami w systemie



# II etap - projektowanie

- Diagram klas - Zawiera informacje o statycznych związkach między elementami (klasami)
- Klasy są ściśle powiązane z technikami programowania zorientowanego obiektowo
- Są jednymi z istotniejszych diagramów w UML



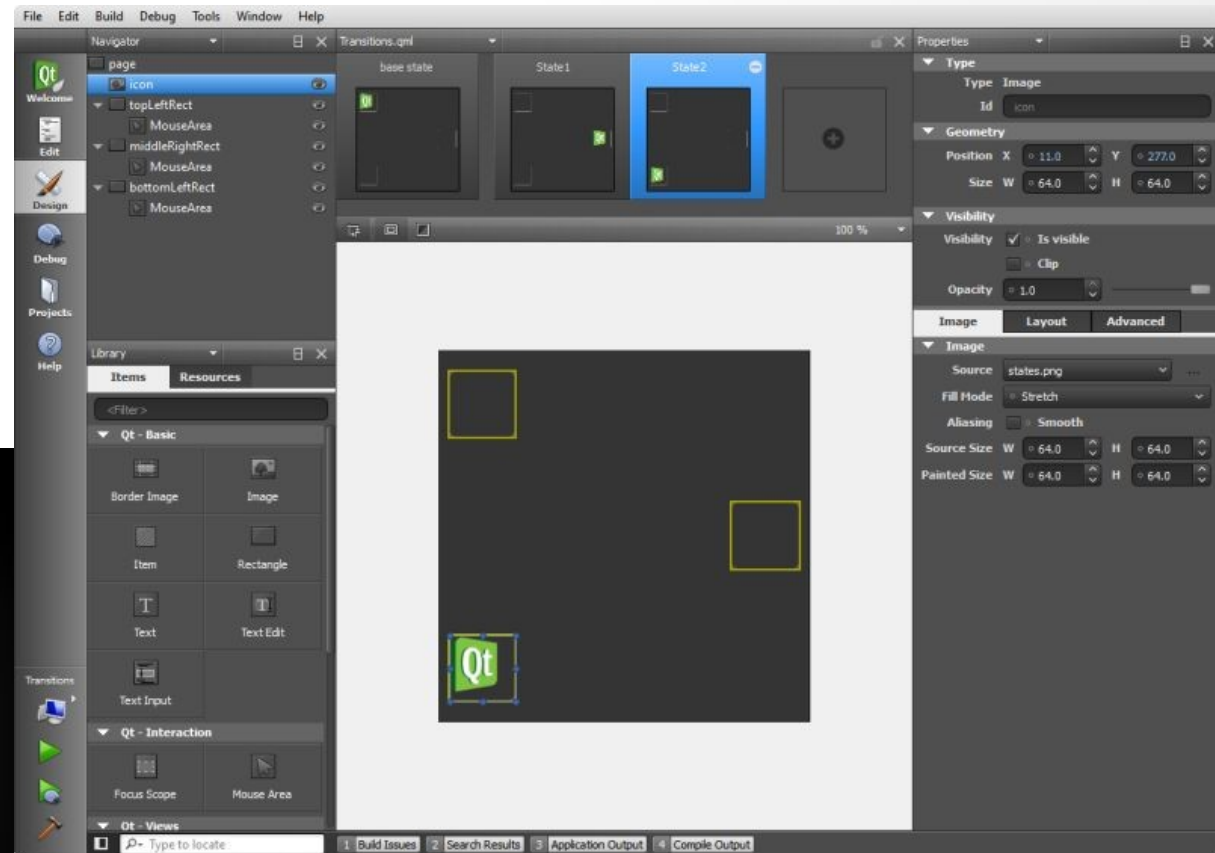
# III etap - implementacja

- Realizacja ustalonej architektury poprzez implementację składowych (modułów) i połączeń między nimi
- Wstępna generacja kodów za pomocą diagramów klas w programie Visual Paradigm



# III etap - implementacja

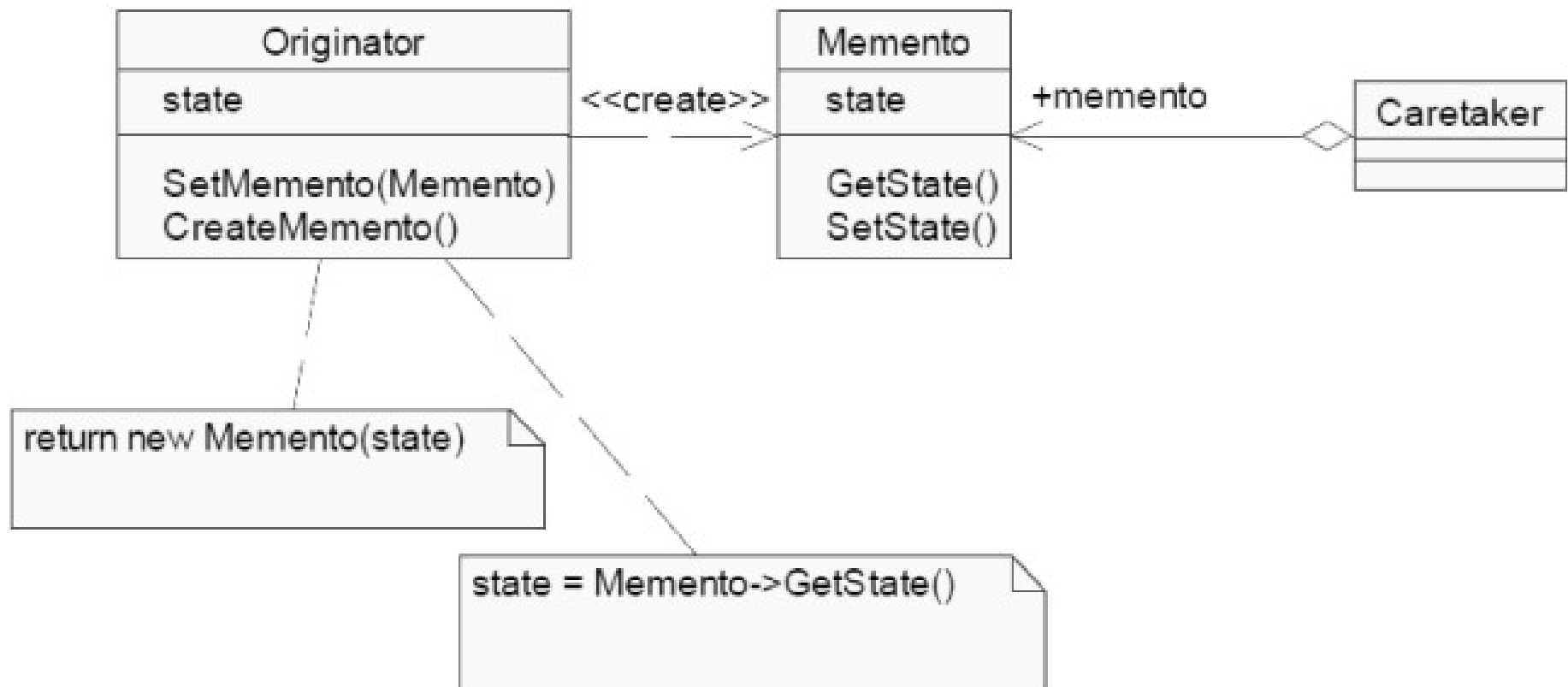
- Wybranie środowiska
- Wstępny zarys aplikacji w programie QT



Code less.  
Create more.  
Deploy everywhere.

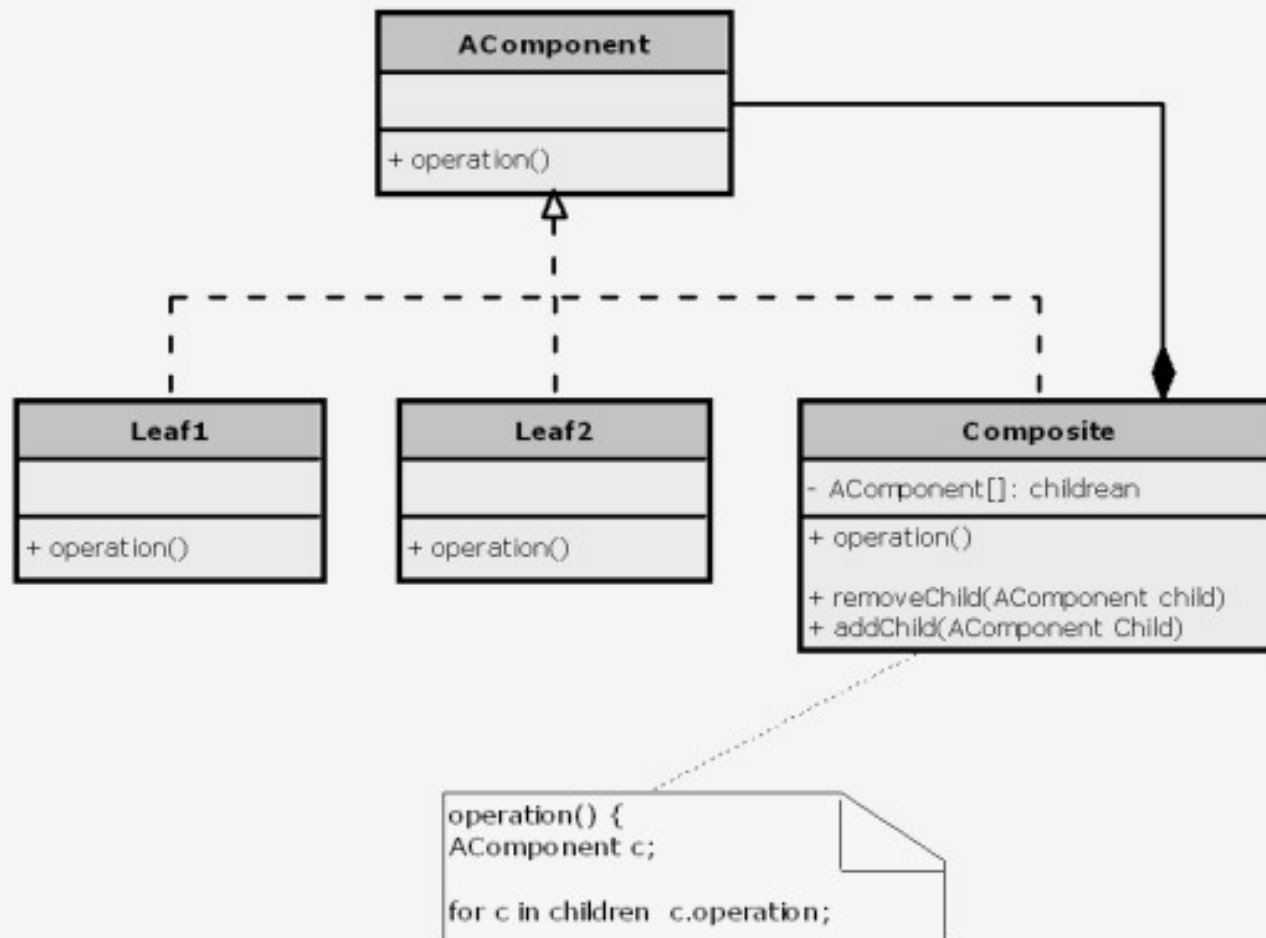
# III etap - implementacja

- Czynnościowe wzorce projektowe
- Pamiętka



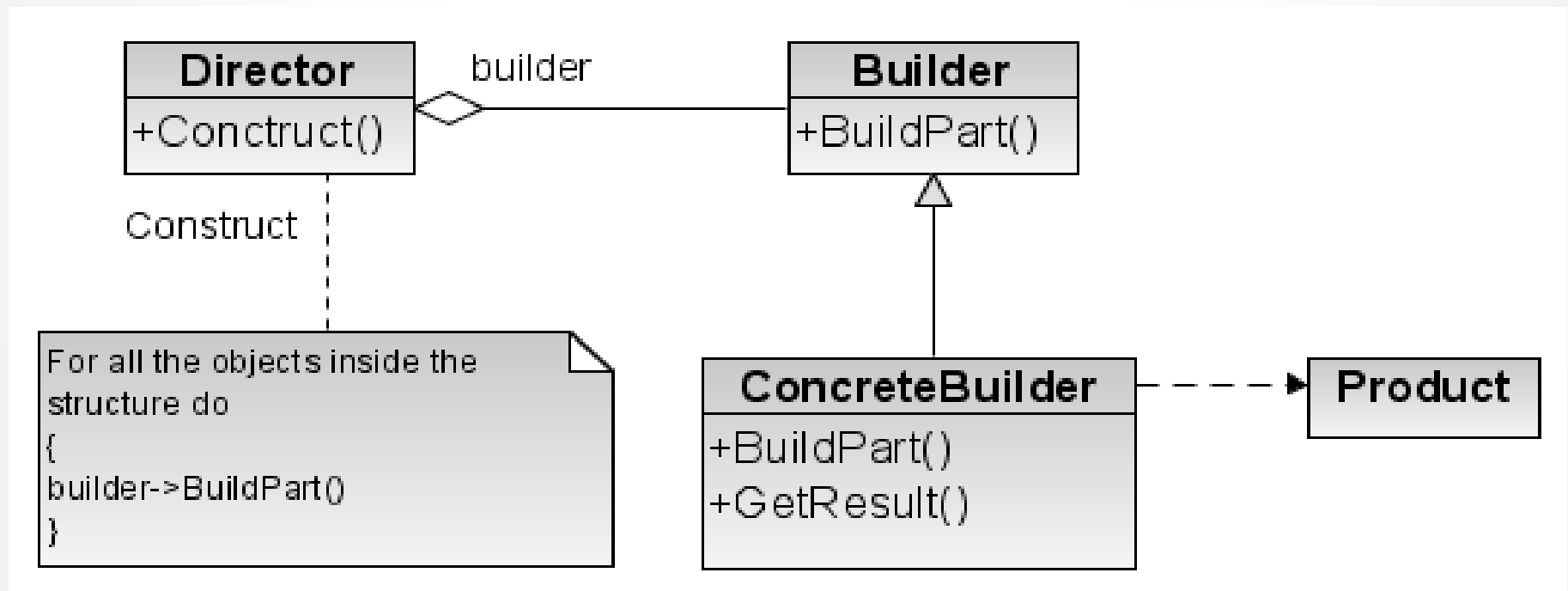
# III etap - implementacja

- Strukturalne wzorce projektowe
- Kompozyt



# III etap - implementacja

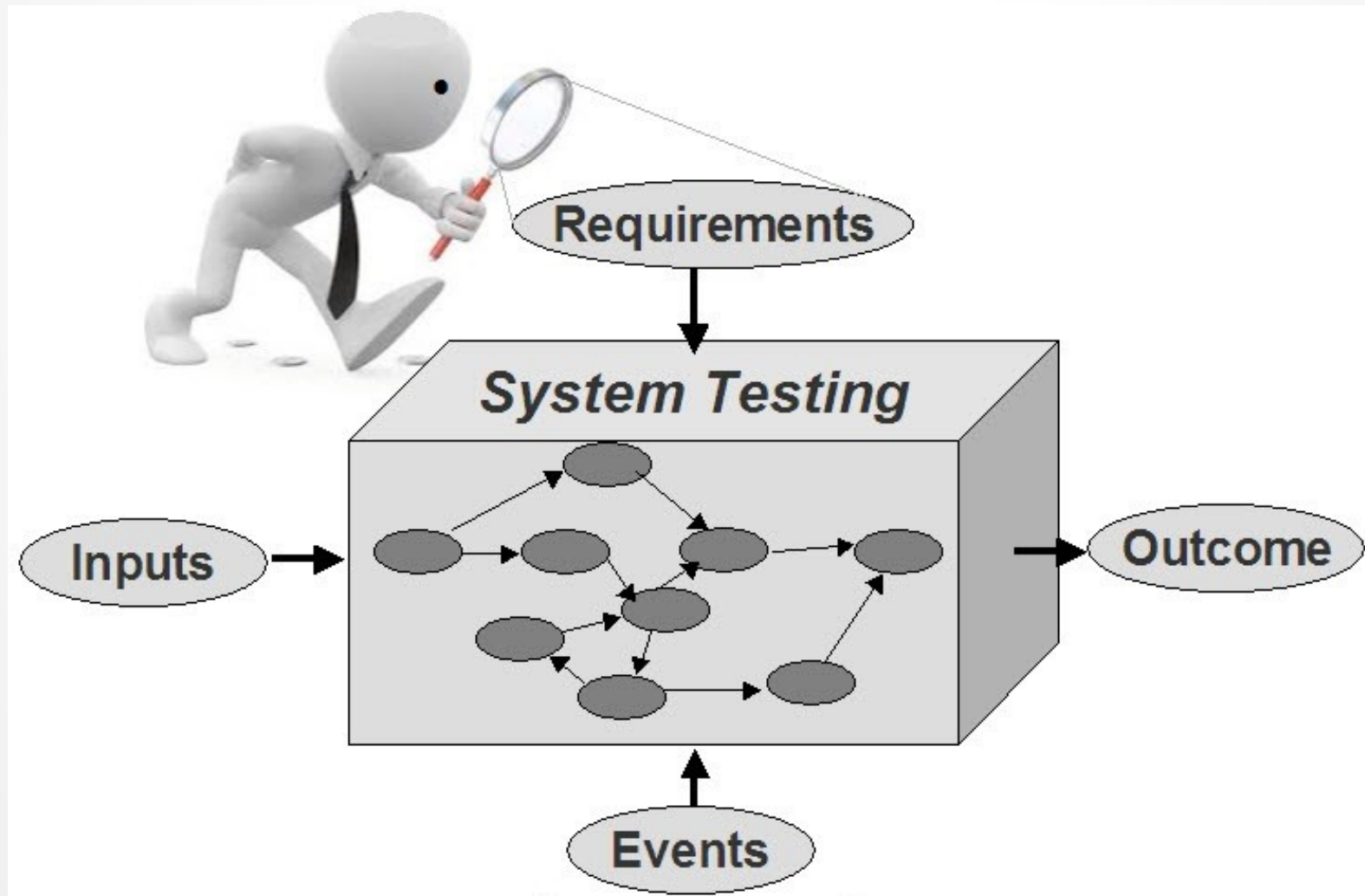
- Kreacyjne wzorce projektowe
- Wzorzec Budowniczy (ang. Builder)





# IV etap - integracja

- Zintegrowanie poszczególnych składowych w jeden system, testowanie całego systemu



# Reengineering



# Inżynieria Oprogramowania

Dziękujemy za uwagę.

