

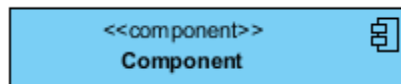
## **Inżynieria oprogramowania**

### **Część 7: UML – Diagramy komponentów i wdrożenia**

Komponent .....	3
Interfejs .....	3
Modelowanie plików wykonywalnych i bibliotek.....	3
zadanie 1.....	3
Modelowanie tabel, plików i dokumentów .....	3
Zadanie 2 .....	4
Modelowanie interfejsu programowego (API) .....	4
Zadanie 3 .....	4
Modelowanie kodu źródłowego .....	4
Zadanie 4 .....	5
Diagram Wdrożenia .....	5
Zadanie 5 .....	5
Połączenia .....	6
Zadanie 6 .....	6
Zadanie 7 .....	6
Zadanie 8 .....	7

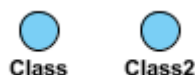
## KOMPONENT

Komponent to fizyczna, wymienna część systemu, która wykorzystuje i realizuje pewien zbiór interfejsów na diagramie przedstawiany jako prostokąt z bocami.



## INTERFEJS

Interfejs to zestaw operacji, które wyznaczają usługi oferowane przez klasę lub komponent.

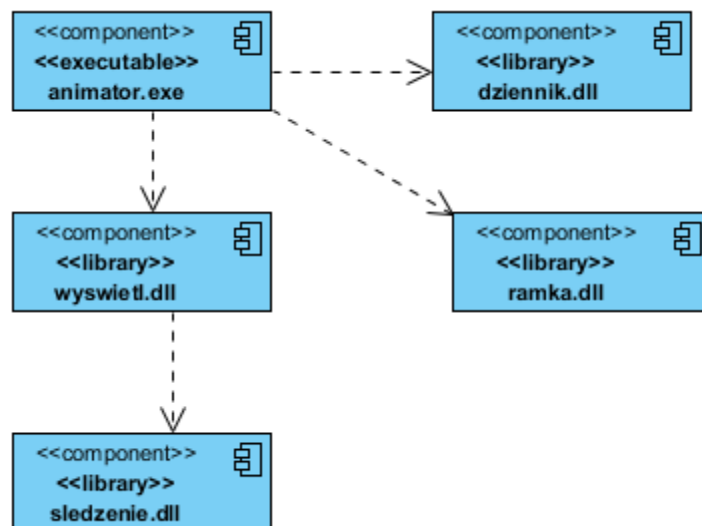


## MODELOWANIE PLIKÓW WYKONYWALNYCH I BIBLIOTEK

Jeżeli system składa się z wielu plików wykonywalnych i licznych bibliotek obiektowych to przy użyciu komponentów można zobrazować decyzje projektowe dotyczące systemu fizycznego.

### ZADANIE 1

Opracować następujący diagram komponentów:

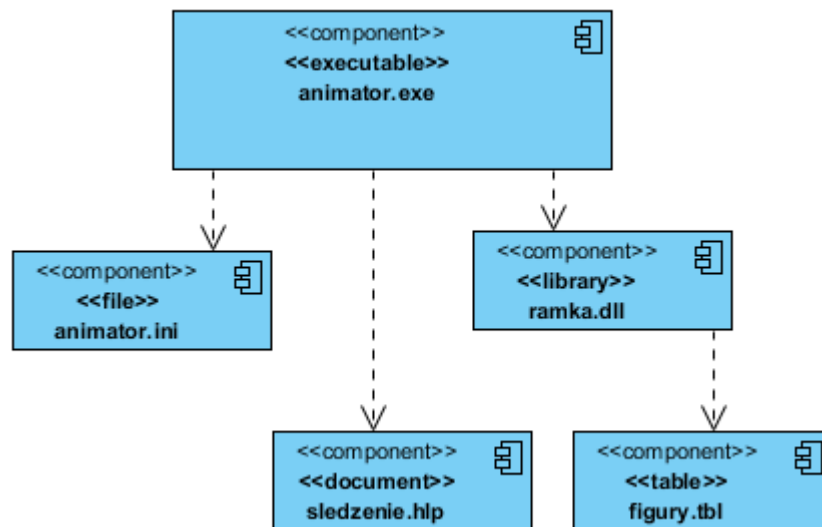


## MODELOWANIE TABEL, PLIKÓW I DOKUMENTÓW

Istnieją inne rodzaje (niż programy wykonywalne i biblioteki) pomocniczych komponentów, które są niezbędne w działającym systemie. Składnikami implementacji mogą być np. dokumenty pomocy, skrypty oraz pliki dzienników, inicjalizacyjne, z danymi, instalacyjne i z procedurami kasowania. Modelowanie takich komponentów jest istotną częścią procesu zarządzania konfiguracją systemu.

## ZADANIE 2

Opracować następujący diagram komponentów:

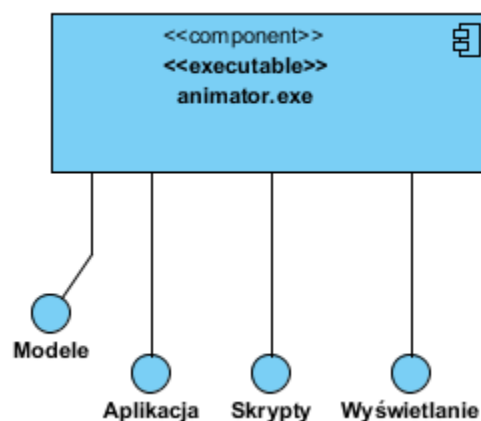


## MODELOWANIE INTERFEJSU PROGRAMOWEGO (API)

Programista budujący system z gotowych komponentów musi znać ich interfejsy programowe, umożliwiające łączenie ich ze sobą. Operacje wchodzące w skład niebanalnego API, są zwykle bardzo liczne. Należy obrazować jedynie te elementy interfejsu, które są istotne w danym otoczeniu.

## ZADANIE 3

Opracować następujący diagram komponentów:

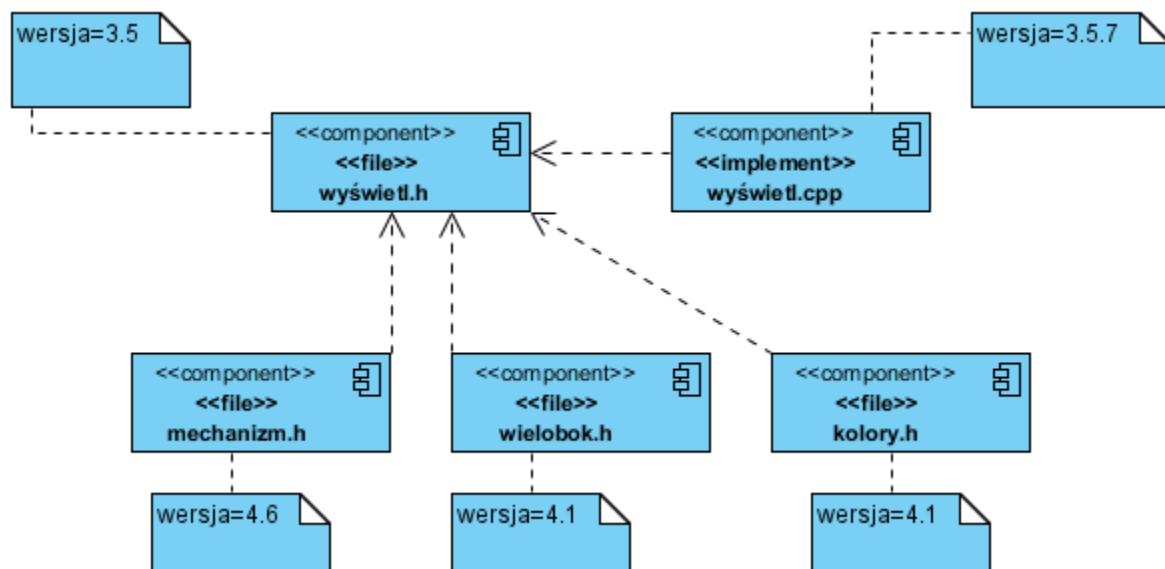


## MODELOWANIE KODU ŹRÓDŁOWEGO

Graficzne modelowanie kodu źródłowego jest szczególnie użyteczne do obrazowania zależności kompilacyjnych między plikami.

#### ZADANIE 4

Opracować następujący diagram komponentów:



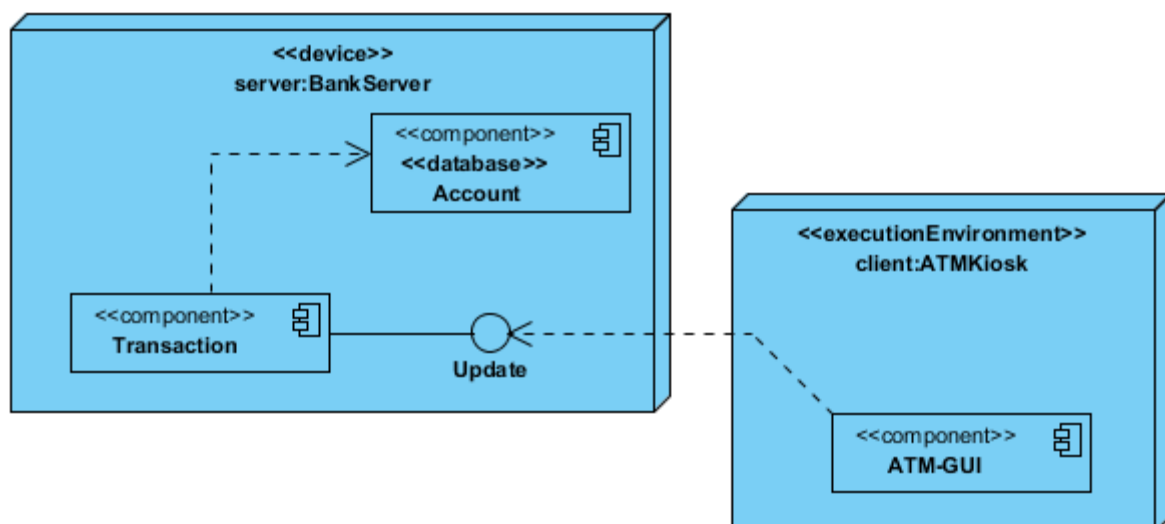
#### DIAGRAM WDROŻENIA

Wszystkie komponenty systemu informatycznego (SI) są wdrażane na sprzęcie komputerowym. Opracowując architekturę SI należy rozważać zarówno wymiar logiczny (klasy, interfejsy) jak i fizyczny (komponenty reprezentujące fizyczne opakowanie bytów logicznych oraz węzły reprezentujące sprzęt na którym te komponenty są posadowione).

**Węzeł** to fizyczny składnik działającego systemu; reprezentuje zasoby obliczeniowe; ma zwykle pewną ilość pamięci i zdolność przetwarzania. Węzłów używa się do modelowania układu sprzętu komputerowego, na którym działa system; zwykle reprezentują procesory lub urządzenia, na których wdrażane są komponenty.

#### ZADANIE 5

Opracować następujący diagram wdrożenia:



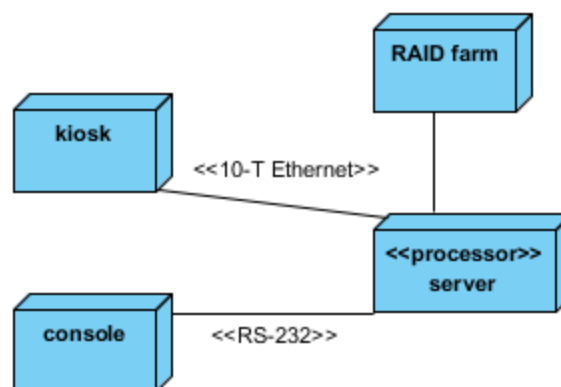
## POŁĄCZENIA

Najczęściej występującym związkiem pomiędzy węzłami jest powiązanie, które w tym przypadku oznacza połączenie fizyczne (np. sieć Ethernet, łącze szeregowe lub wspólna szyna). Powiązań można użyć też do modelowania połączeń pośrednich (np. komunikacja satelitarna między odległymi maszynami).

W przypadku powiązania węzłów mogą być wykorzystane role, liczebność i ograniczenia. Aby rozróżniać rodzaje połączeń warto używać stereotypów dotyczących powiązań.

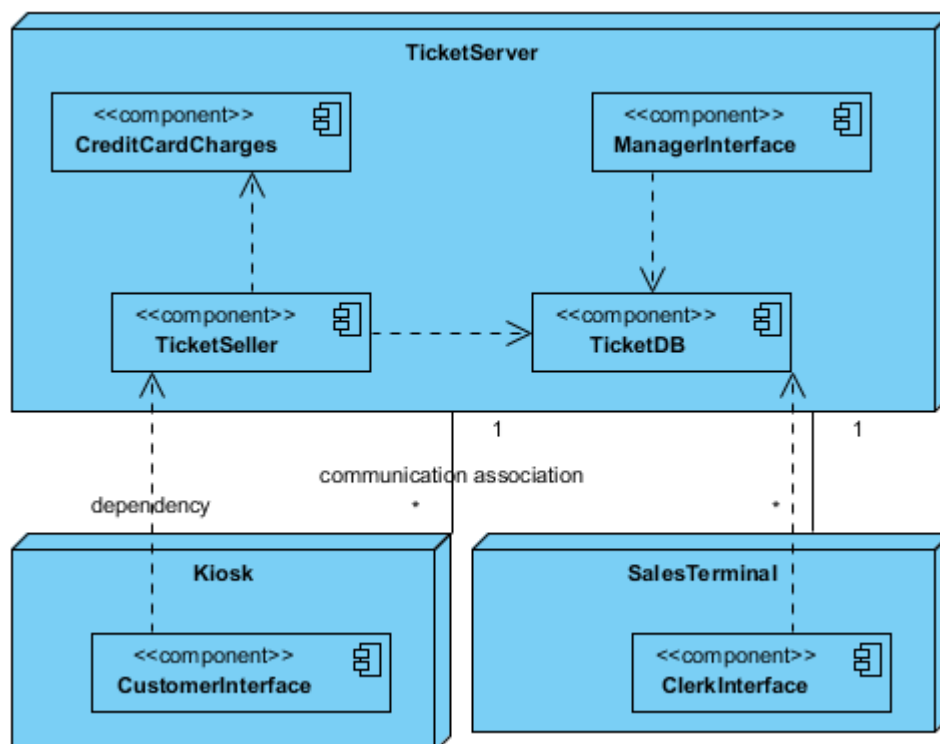
### ZADANIE 6

Wykorzystując odpowiednie stereotypy opracować następujący diagram wdrożenia:



### ZADANIE 7

Opracować następujący diagram wdrożenia:



## ZADANIE 8

Opracować następujący diagram wdrożenia:

