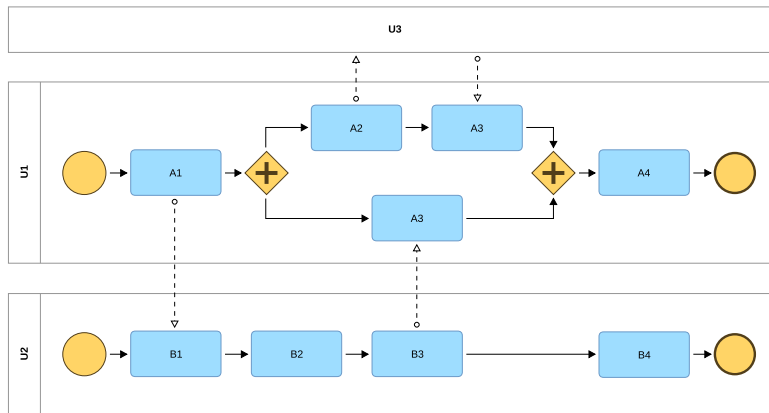


Modelowanie procesów biznesowych

Diagram współpracy
Przepływ komunikatu
Typy procesów

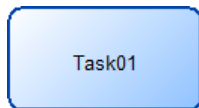


Wykład 01 – diagram procesów

Diagram procesów (*Process Diagram*) –opisuje sekwencję działań wykonywanych w ramach organizacji dla osiągnięcia ustalonego celu (pokazuje przebieg pojedynczego procesu biznesowego).



Zdarzenia (*events*) opisują wystąpienie pewnej sytuacji istotnej z punktu widzenia modelowanego procesu.



Aktywności (*activities*) reprezentują operacje realizowane w ramach procesu (kolejne kroki procesu). Mogą być abstrakcyjne lub określonego typu, który określa sposób wykonania.



Bramki (*gateways*) określają logikę przepływów w procesie biznesowym (rozdzielają i łączy przepływy).



Przeływ sekwencyjny (*sequence flow*) określa kolejność działań realizowanych w ramach pojedynczego procesu.

Token (żeton) – znacznik generowany dla każdej instancji procesu, który przemieszcza się od zdarzenia początkowego do zdarzenia końcowego, przechodząc przez węzły przepływu (czynności, bramki, zdarzenia), zgodnie z kierunkiem wyznaczonym przez przepływ sekwencyjny. Pozycja tokenu określa aktualnie wykonywany krok procesu.

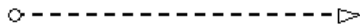
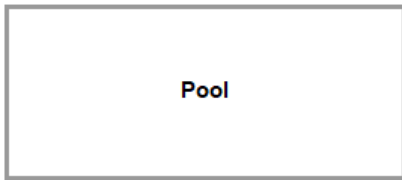
Diagram współpracy/kooperacji/kolaboracji (ang. Collaboration Diagram) – opisuje współpracę pomiędzy dwoma lub więcej uczestnikami procesu biznesowego, uwzględniając przepływ komunikatów (wymianę informacji) związanych z realizacją operacji wykonywanych w ramach współpracy.

Uczestnik (ang. Participant) – reprezentuje jednego aktora uczestniczącego we współpracy, może być opisany przez:

- **podmiot** (ang. entity) szczegółowo określający typ jednostki uczestnika (np. nazwa firmy lub instytucji, nazwisko konkretnej osoby, itp.),
- **rolę** (ang. role) określającą ogólnie funkcję pełnioną przez uczestnika (np. sprzedawca, kupujący, klient, dostawca, itp.).

Przepływ komunikatu (ang. Message flow) – reprezentuje wymianę informacji, która jest realizowana pomiędzy dwoma współpracującymi uczestnikami obejmującą przesyłanie danych, dokumentów, produktów, itp.

Podstawowe elementy diagram współpracy





Basen (ang. Pool) reprezentuje uczestnika procesu. Może zawierać model procesu realizowanego przez danego uczestnika lub pozostać pusty. Na rysunku basen w postaci rozwiniętej (stosowany gdy proces uczestnika jest pokazywany) oraz zwinięty (stosowany, gdy proces uczestnika nie jest widoczny).

Tor lub **linia** (ang. Lane) dzieli basen na części reprezentujące jednostki lub role realizowane w obrębie jednego uczestnika reprezentowanego przez basen.










Przepływ komunikatu (ang. Message flow) opisuje wymianę informacji realizowaną pomiędzy uczestnikami (basenami).

Komunikat (ang. Message) artefakt, reprezentuje treść wiadomości przekazywanej pomiędzy uczestnikami (basenami)

Zasady tworzenia diagramu współpracy

- ❑ Basen reprezentuje jednego uczestnika.
- ❑ Basen może zawierać proces realizowany przez uczestnika lub może pozostać pusty.
- ❑ Basen może być podzielony na kilka torów reprezentujących jednostki organizacyjne lub role realizowane w obrębie danego uczestnika.
- ❑ Jeden proces na diagramie, zazwyczaj reprezentujący operacje realizowane przez uczestnika z punktu widzenia którego prezentowana jest kooperacja, może być umieszczony poza basenem (znajduje się w tzw. basenie domyślnym).
- ❑ Procesy realizowane przez poszczególnych uczestników kooperacji mają własne tokeny i są realizowane niezależnie, kooperacja jest zakończona gdy wszystkie tokeny osiągną zdarzenia końcowe swoich procesów.
- ❑ Przepływ sekwencyjny (linia ciągła zakończona czarnym grotem) może występować tylko w ramach jednego basenu (nie może przekraczać jego granic), ale może przepływać pomiędzy torami. 
- ❑ Przepływ komunikatu (linia przerywana z białym grotem) może odbywać się tylko pomiędzy basenami, przepływ komunikatu nie przenosi tokenu. 

Reguły połączeń przepływu komunikatu

Źródło \ Cel						
	—	—	—	—	—	—
	—					—
	—					—
	—					—
	—					—
	—					—

Generowanie komunikatu

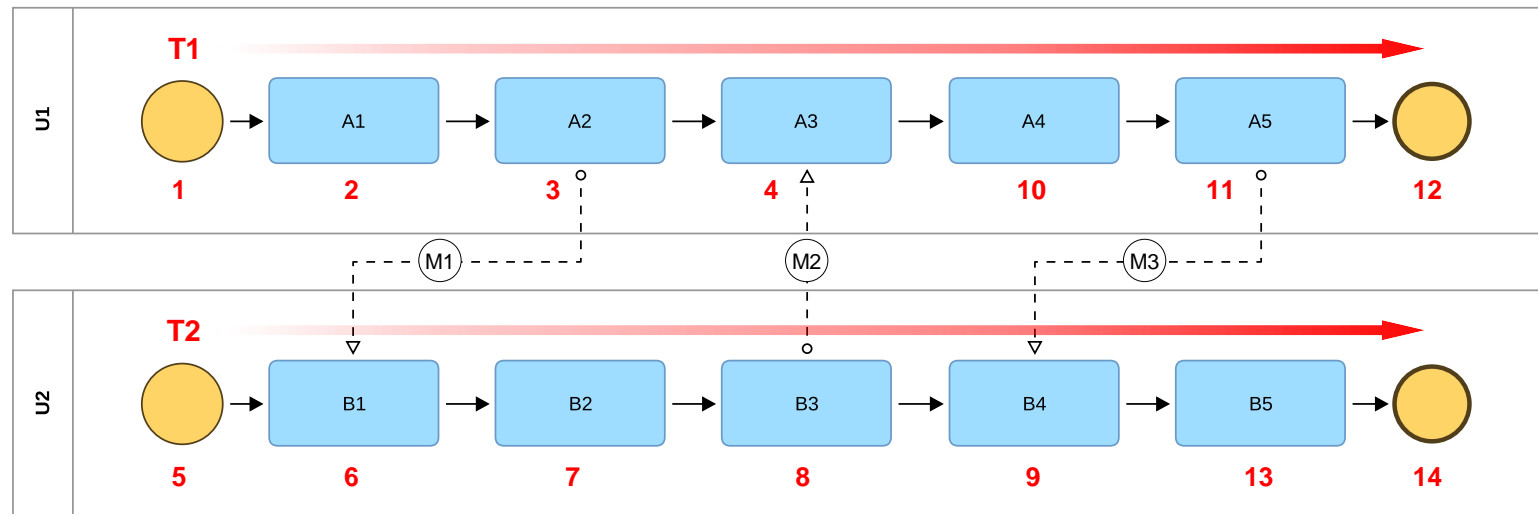
- Komunikat jest wysyłany w chwili gdy token dociera do elementu procesu, który jest jego źródłem.
- Po wygenerowaniu komunikatu token jest przenoszony do kolejnego elementu połączonego przepływem sekwencyjnym (kolejny element jest wykonywany).
- Treść komunikatu jest przenoszona przez przepływ komunikatu do elementu docelowego, czas przejścia komunikatu nie jest określony.

Odbiór komunikatu

- Gdy token procesu osiąga element, który jest celem komunikatu czeka na jego odbiór, tzn. token jest przenoszony do kolejnego elementu, dopiero po odbiorze komunikatu.
- Komunikat może być odebrany tylko gdy element docelowy jest aktywny, tzn. token procesu dotarł do tego elementu.
- Jeżeli komunikat dociera do elementu zanim zostanie on osiągnięty przez token nie zostanie odebrany, tzn. treść komunikatu zostanie utracona.

Korzystając z zasad generowania i odbierania komunikatów (oczekiwanie na odbiór) możliwe jest synchronizowanie procesów realizowanych w ramach kooperacji.

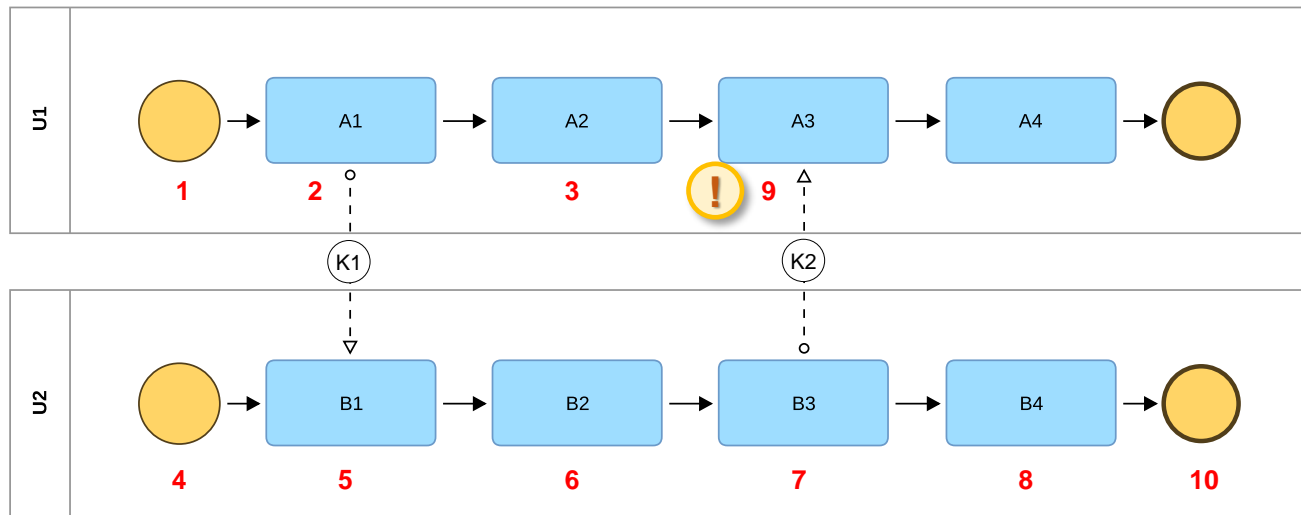
Przepływ komunikatów – przykład 1



Kooperacja uczestników U1 (zadania A1..A5) i U2 (zadania B1..B5):

1. Wystąpienie zdarzenia początkowego w procesie uczestnika U1 (powstaje token T1).
2. Wykonanie zadań A1 i A2 (przeływ komunikatu K1 do U2), przeływ tokenu T1 do zadania A3 i oczekiwanie na komunikat K2 (proces zostaje zawieszony do chwili odbioru komunikatu).
3. Odbiór komunikatu K1 uruchamia zdarzenie początkowe procesu uczestnika U2 (powstaje token T2), następuje wykonanie zadań B1 (odbiór komunikatu K1), B2 i B3 (przeływ komunikatu K2 do U1), przeływ tokenu T2 do B4 i oczekiwanie na komunikat K3.
4. Odbiór komunikatu K2 kończy zadanie A3, następuje wykonanie zadań A4 i A5 (przeływ komunikatu K3 do U2), token T1 dociera do zdarzenia końcowego, proces U1 jest zakończony.
5. Odbiór komunikatu K3 kończy zadanie B4, wykonanie zadania B5, token T2 osiąga zdarzenie końcowe, proces U2 jest zakończony, wszystkie tokeny osiągnęły zdarzenia końcowe kooperacja jest zakończona.

Przeptyw komunikatów – przykład 2 (błąd)

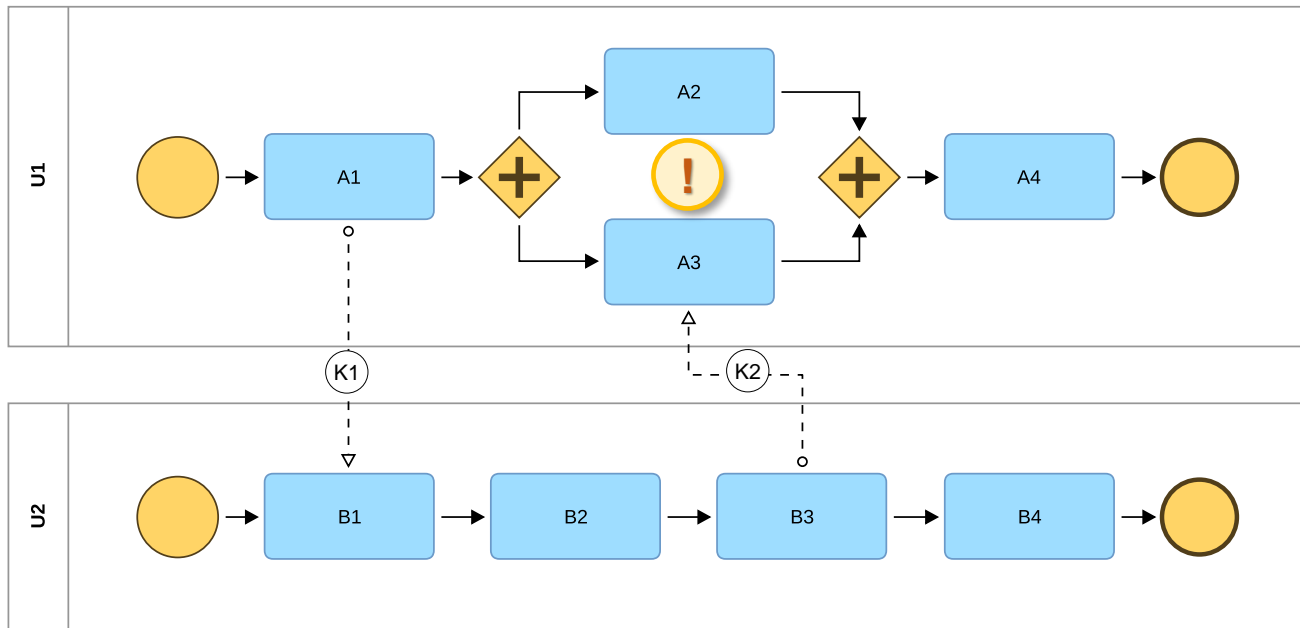


Kooperacja uczestników U1 (zadania A1..A4) i U2 (zadania B1..B4):

1. Proces uczestnika U1 wykonuje zadanie A1 (przeptyw komunikatu K1 do U2), przechodzi do zadań A2, A3 i oczekuje na komunikat K2 (token pozostaje na A3 do chwili odbioru komunikatu).
2. Proces uczestnika U2, zainicjowany komunikatem K1, wykonuje zadania B1, B2, B3 (wysyłka komunikatu K2) i B4. Token dociera do zdarzenia końcowego i proces zostaje zakończony.

Czasy wykonania A2 i B2 nie są znane. Jeżeli czas wykonania B2 jest większy od czasu A2, komunikat K2 zostanie wysłany zanim token osiągnie A3. W takim wypadku komunikat K2 nigdy nie zostanie odebrany (patrz s. 7), token nie będzie mógł opuścić zadania A3 i osiągnąć zdarzenia końcowego. Proces uczestnika U1 utknie, a kooperacja nigdy nie zostanie zakończona. Model jest nieprawidłowy, ponieważ nie gwarantuje zakończenia.

Przeptyw komunikatów – przykład 3 (korekta)



Kooperacja uczestników U1 (zadania A1..A4) i U2 (zadania B1..B4):

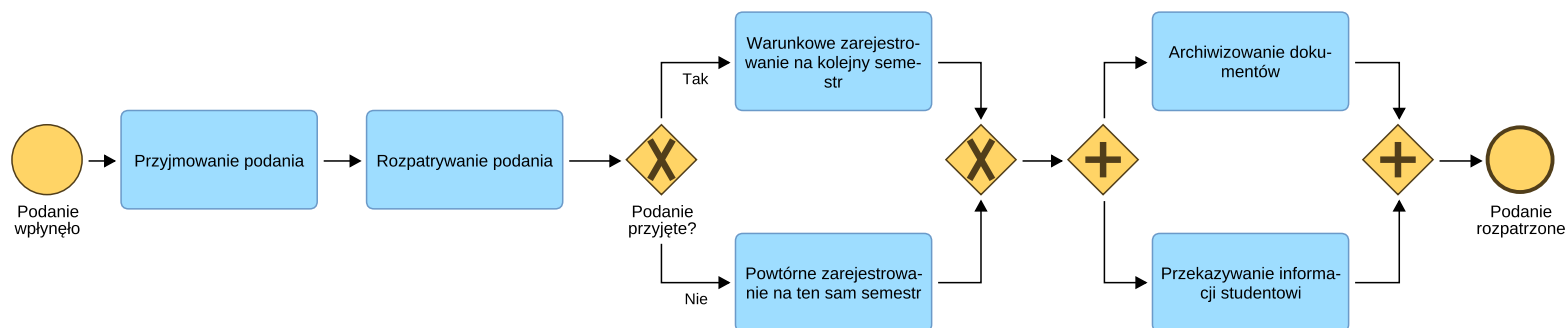
Zadania A2 i A3 (oczekiwanie na komunikat K2) są wykonywane równolegle, token przejdzie do zadania A4 po zakończeniu A2 i A3.

Niezależnie od czasów wykonania zadań A2 i B2 komunikat K2 zostanie odebrany. Token procesu uczestnika U1 przechodzi do zadania A3 (oczekiwanie na komunikat K2) natychmiast po wysłaniu komunikatu K1, niezależnie realizując inne operacje (w tym przypadku zadanie A2). Proces nigdy nie utknie, więc kooperacja zawsze zostanie zakończona. Model jest prawidłowy.

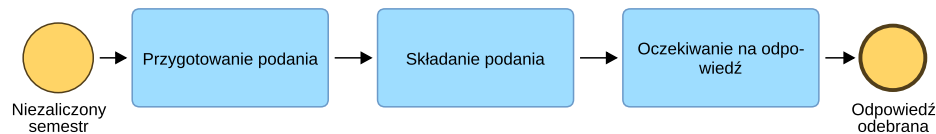
Podanie o wpis warunkowy – diagram współpracy

Uczestnicy procesu: Uniwersytet, Student

Proces Uniwersytetu (przyjmuje wniosek, zajmuje się jego rozpatrzeniem, udziela odpowiedzi i archiwizuje dokumenty).

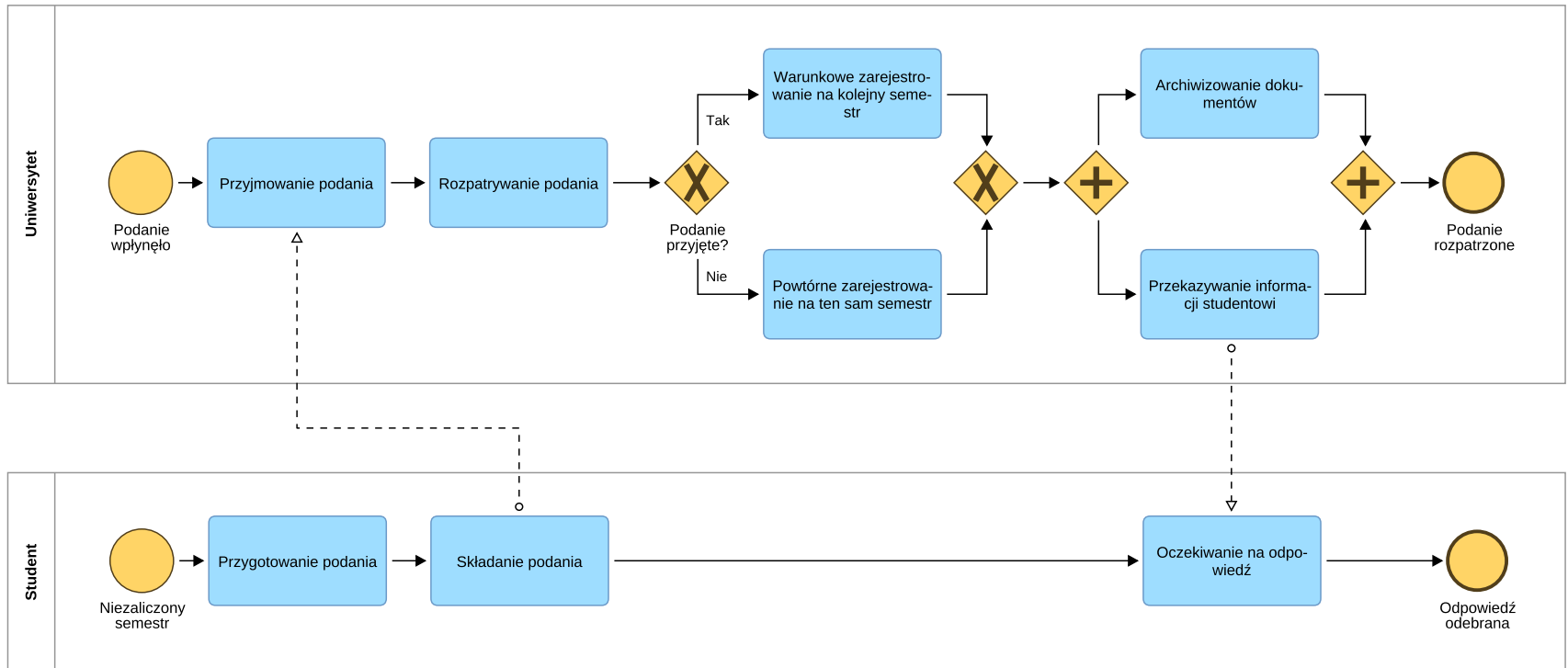


Proces Studenta (przygotowuje wniosek, składa w BOS, odbiera odpowiedź).



Przepływ komunikatu: Składanie podania, Przekazywanie informacji studentowi.

Podanie o wpis warunkowy – diagram współpracy, v.1

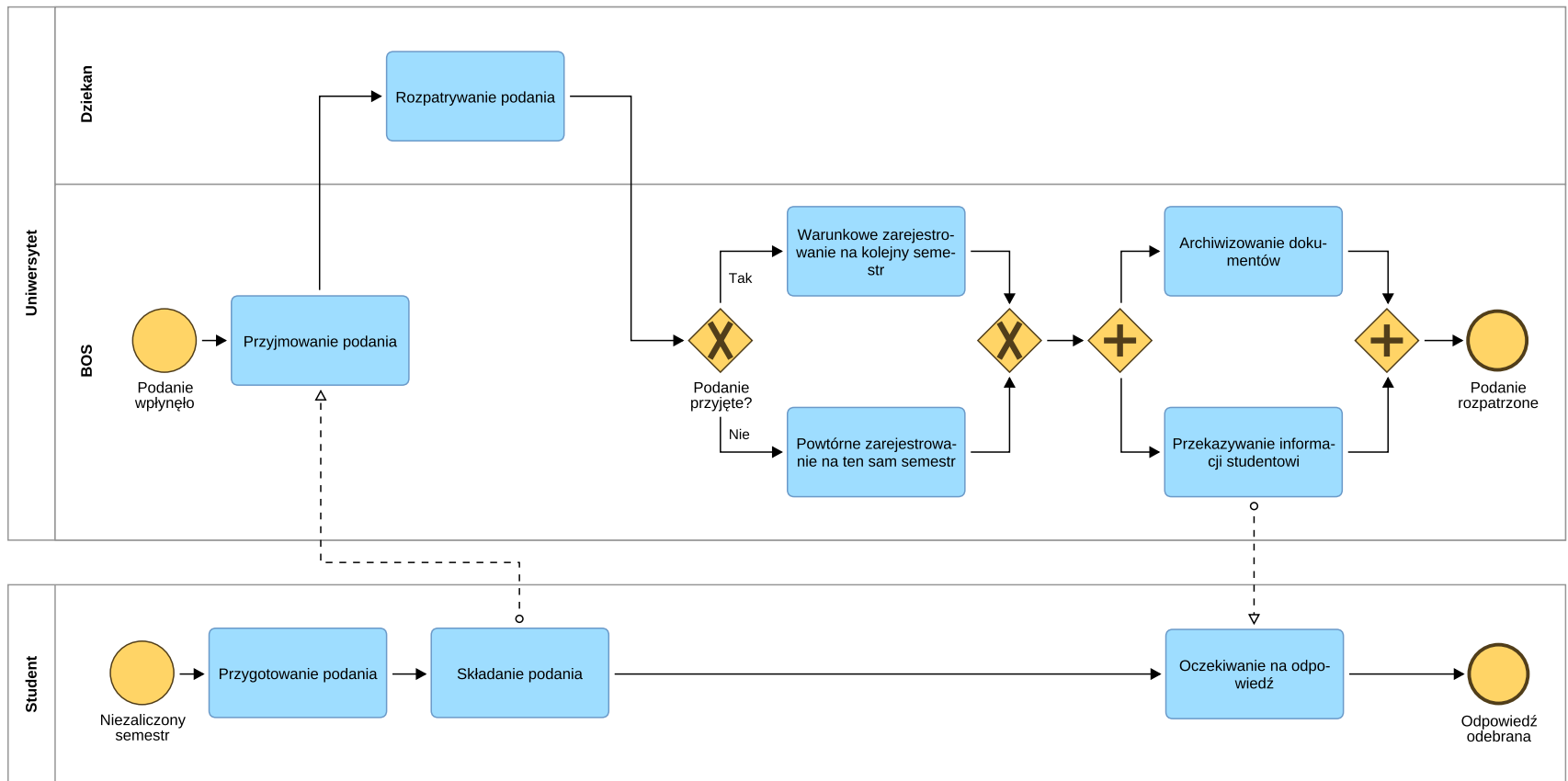


1. Składając podanie do BOS Student generuje przepływ komunikatu, który inicjuje uruchomienie procesu Uniwersytetu.
2. BOS generuje przepływ komunikatu przekazując decyzję Studentowi.
3. Proces Studenta zostanie zakończony dopiero po odbiorze informacji (komunikatu) z BOS-u (token nie może opuścić zadania przed odbiorem komunikatu).

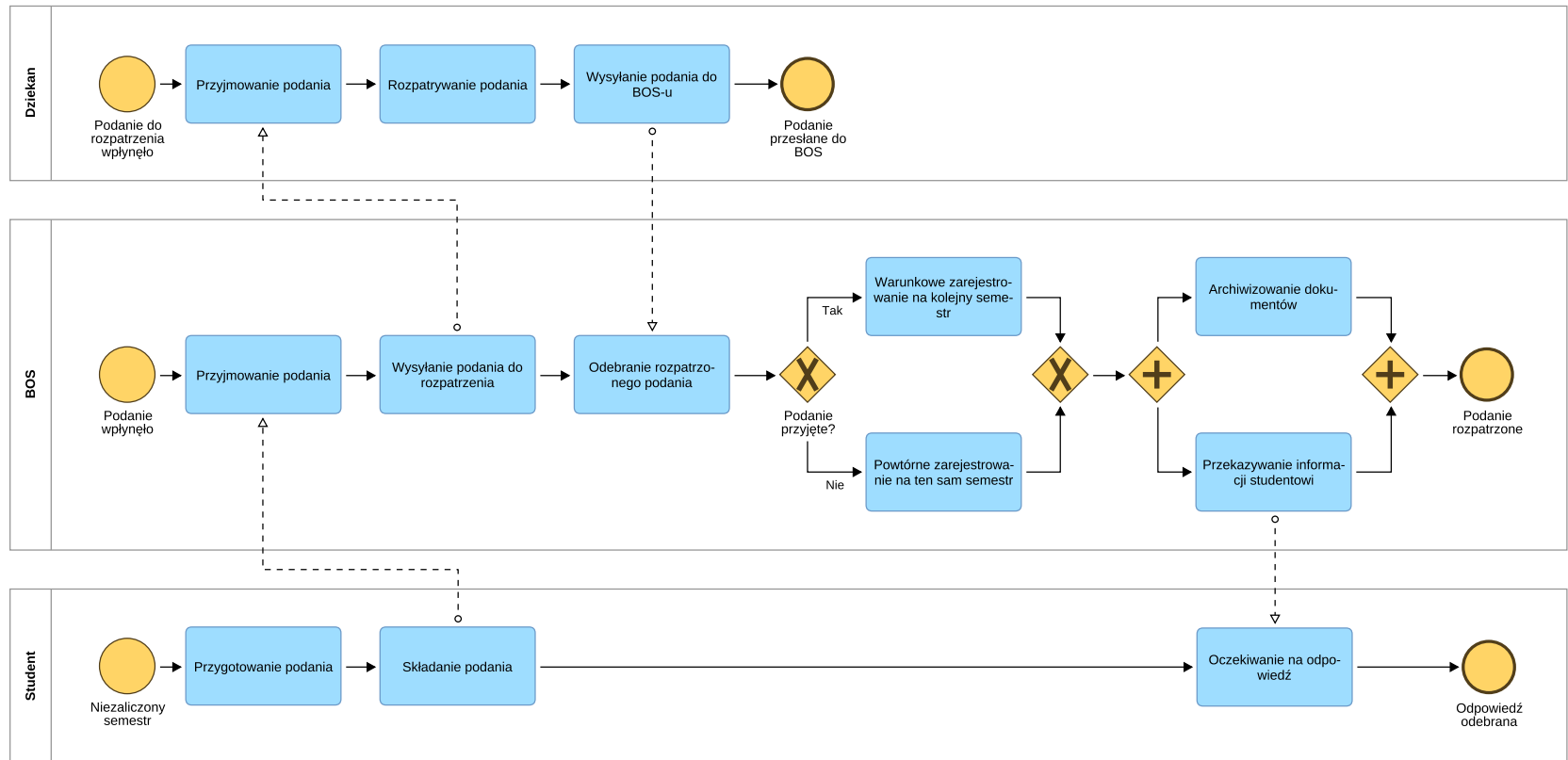
Podanie o wpis warunkowy – diagramy współpracy v.2

Jednostki organizacyjne uniwersytetu realizujące proces (tory)

1. Biuro Obsługi Studenta (BOS).
2. Dziekan Wydziału.



Podanie o wpis warunkowy – diagramy współpracy v.3



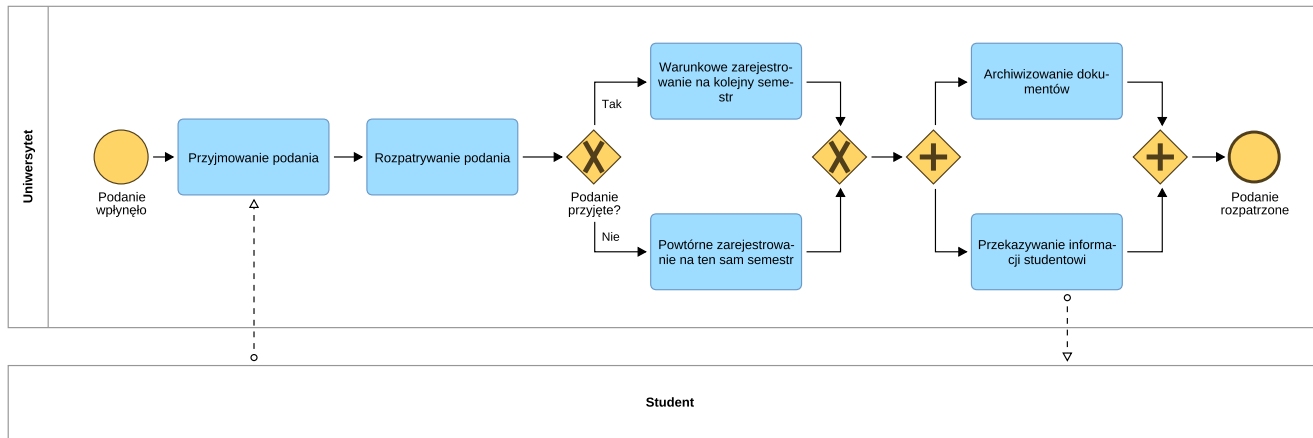
Jednostki organizacyjne Uniwersytetu (BOS, Dziekan) zamodelowane jako baseny (3 uczestników)

1. BOS i Dziekan wykonują odrębne procesy.
2. Komunikacja pomiędzy BOS-em i Dziekanem realizowana przepływem komunikatu (przepływ sekwencyjny nie może przekraczać granic basenu, patrz s. 5).
3. Dodatkowe zadania związane z generowaniem i odbieraniem komunikatów.

Perspektywy diagramu współpracy

Diagram współpracy zazwyczaj jest rysowany z perspektywy jednego użytkownika (zazwyczaj organizacji, która tworzy model).

Perspektywa Uniwersytetu



Perspektywa Studenta

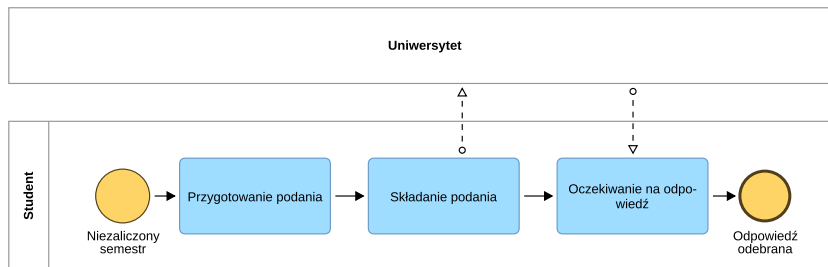
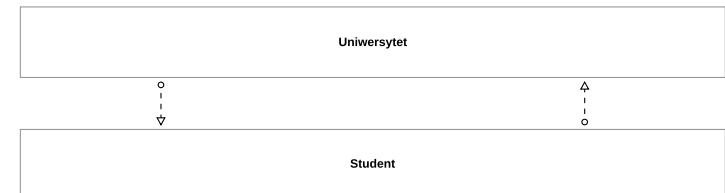
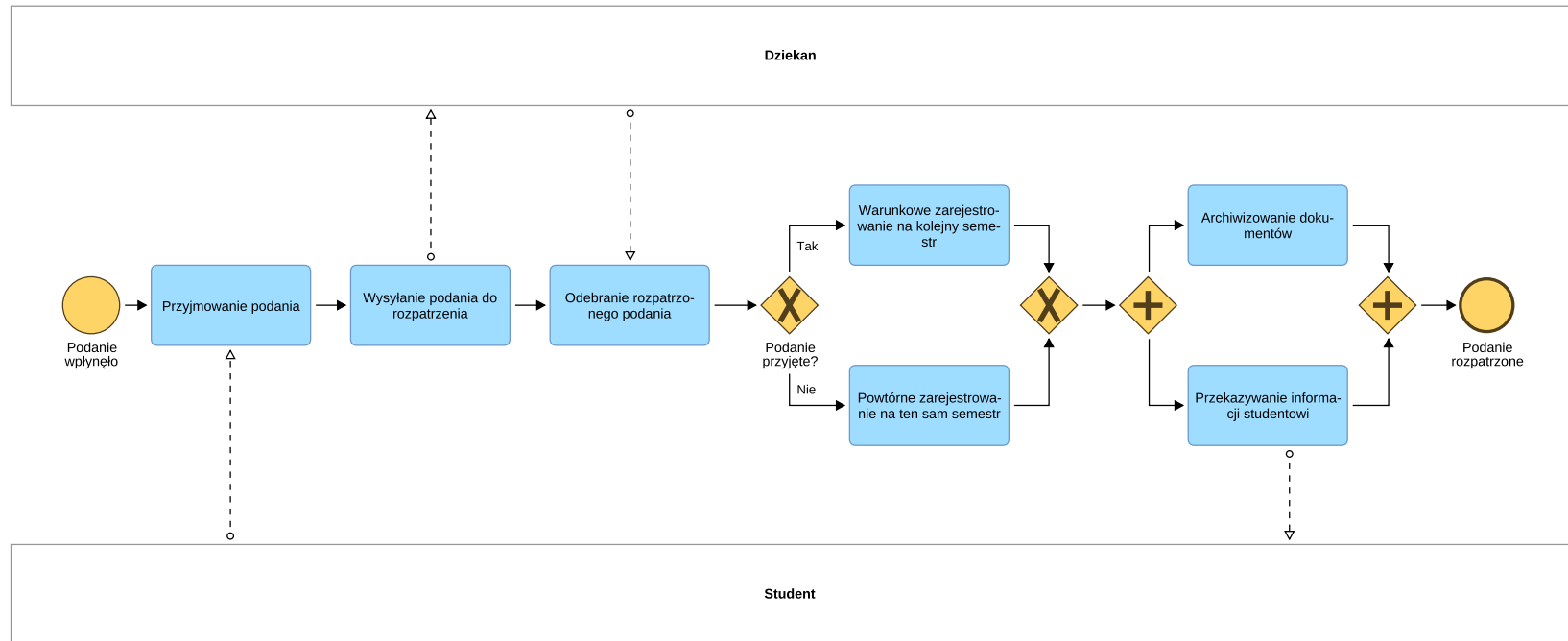


Diagram współpracy bez procesów



Perspektywy diagramu współpracy

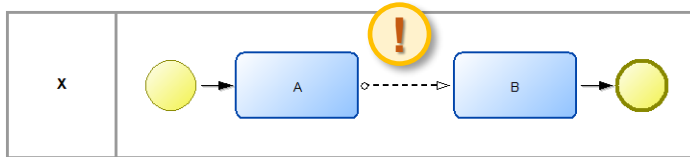
Perspektywa BOS (na podstawie s.14)



Jednostki organizacyjne Uniwersytetu (BOS, Dziekan) zamodelowane jako baseny (3 uczestników). Proces realizowany przez BOS bez jawnego basenu (znajduje się w basenie domyślnym, patrz s.5)

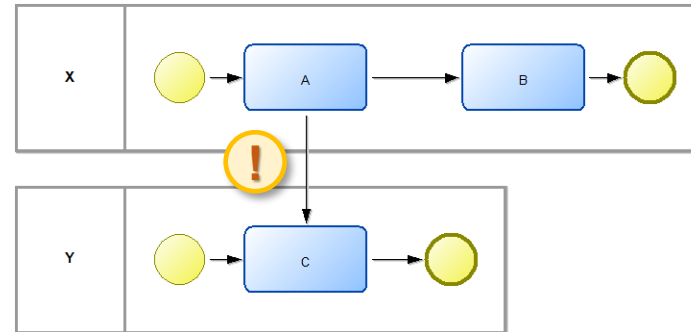
Częste błędy i dobre praktyki modelowania 3

Przepływ komunikatu w basenie

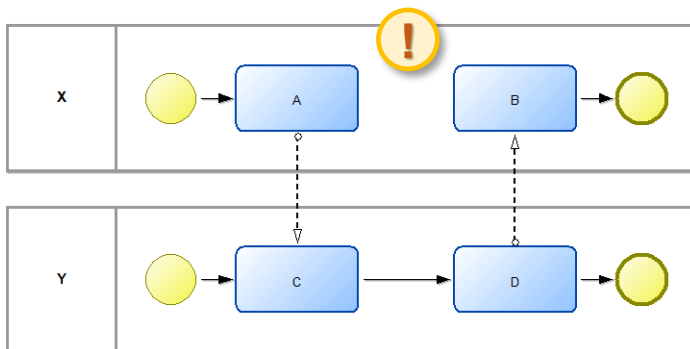


Komunikat może przepływać tylko pomiędzy basenami, przepływ sekwencyjny nie może przekraczać granic basenu (proces musi być realizowany w jednym basenie).

Przepływ sekwencyjny pomiędzy basenami



Brak ciągłości procesu



Przepływ procesu musi zachować ciągłość. Token jest przenoszony tylko przez przepływ sekwencyjny (nie jest przenoszony przez komunikat).

W tym przypadku token nigdy nie osiągnie zadania B i zdarzenia końcowego, komunikat nie będzie odebrany, proces utknie.

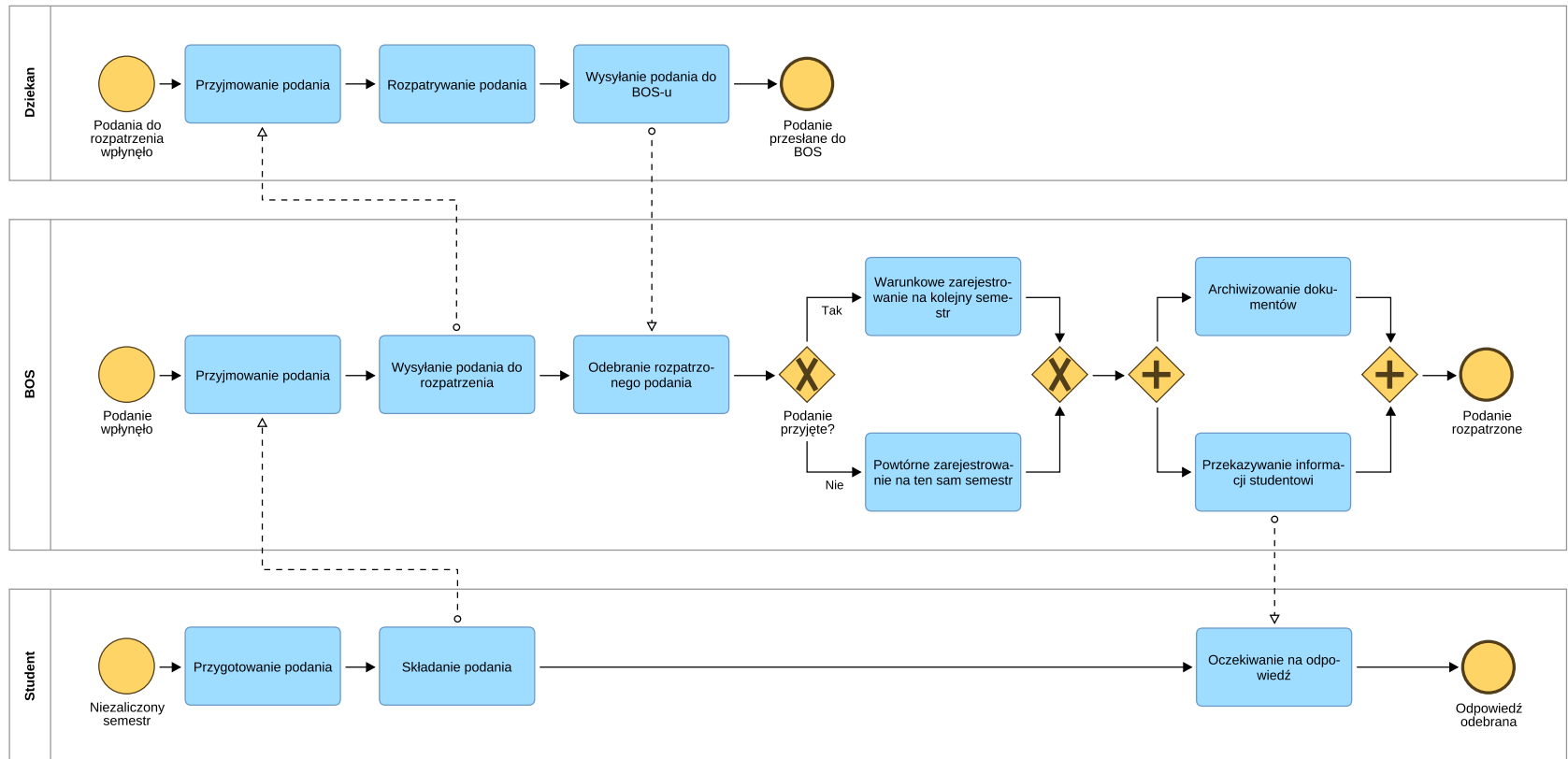
Typy procesów w BPMN 2.0

- Nieokreślony (ang. *None*) – wartość domyślna, proces o nieokreślonym typie;
- Prywatny/wewnętrzny (ang. *Private*) – prezentuje cykl pracy (*workflow*) realizowany wewnątrz organizacji, zazwyczaj nie jest udostępniany partnerom zewnętrznym;
- Publiczny (ang. *Public*) – prezentuje interakcje pomiędzy procesem prywatnym i innym uczestnikiem procesu biznesowego, uwzględnia tylko elementy, które służą do komunikacji z partnerami zewnętrznymi.

Rodzaje procesów prywatnych

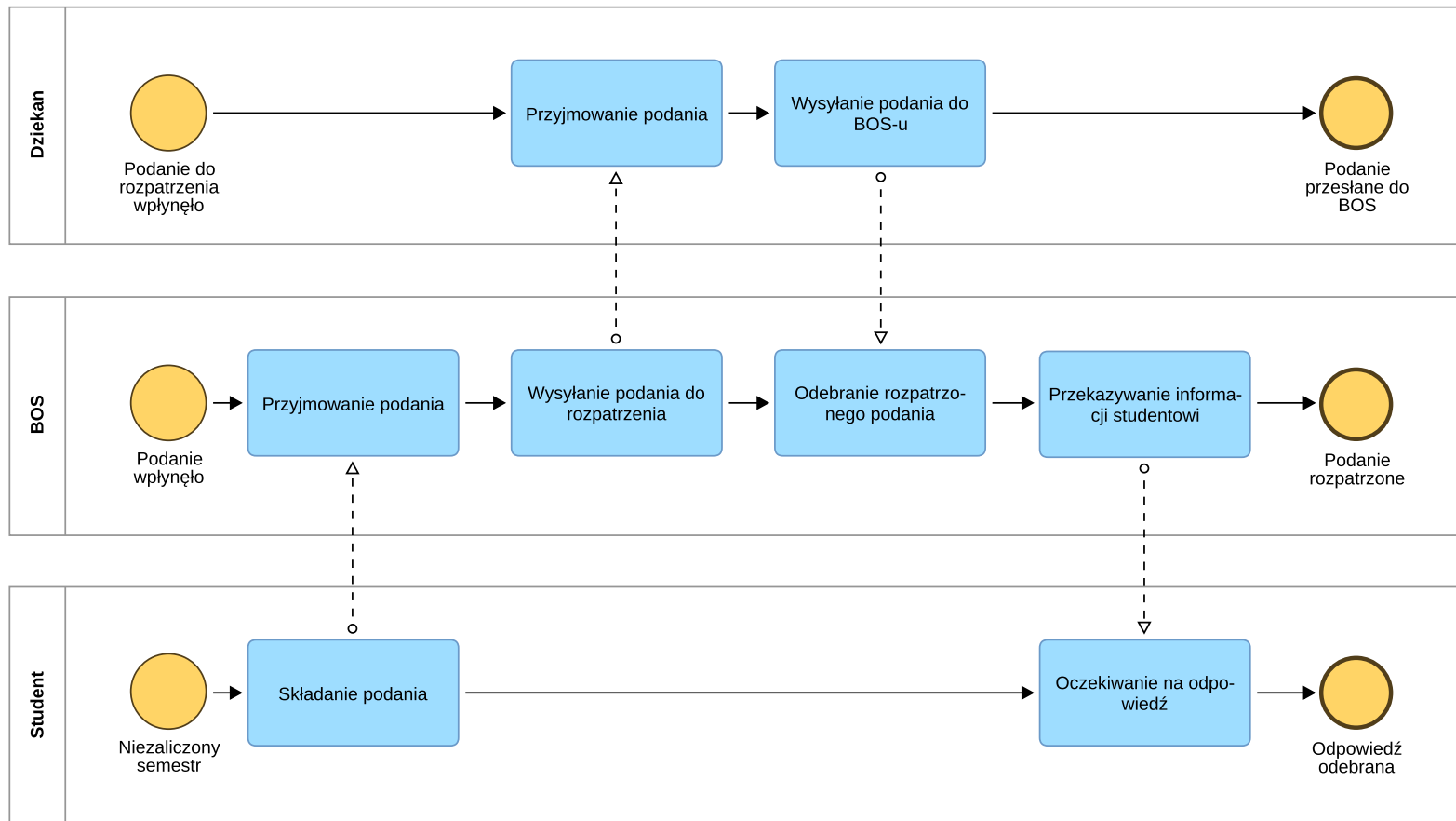
- Niewykonywalne (ang. *private non-executable*) – zdefiniowane na niższym poziomie szczegółowości, wyłącznie w celu dokumentacji przebiegu procesu;
- Wykonywalne (ang. *private executable*) – opisują szczegóły procesu, zawierają wszystkie informacje niezbędne do uruchomienia modelu na silniku procesów biznesowych.

Procesy prywatne – przykład



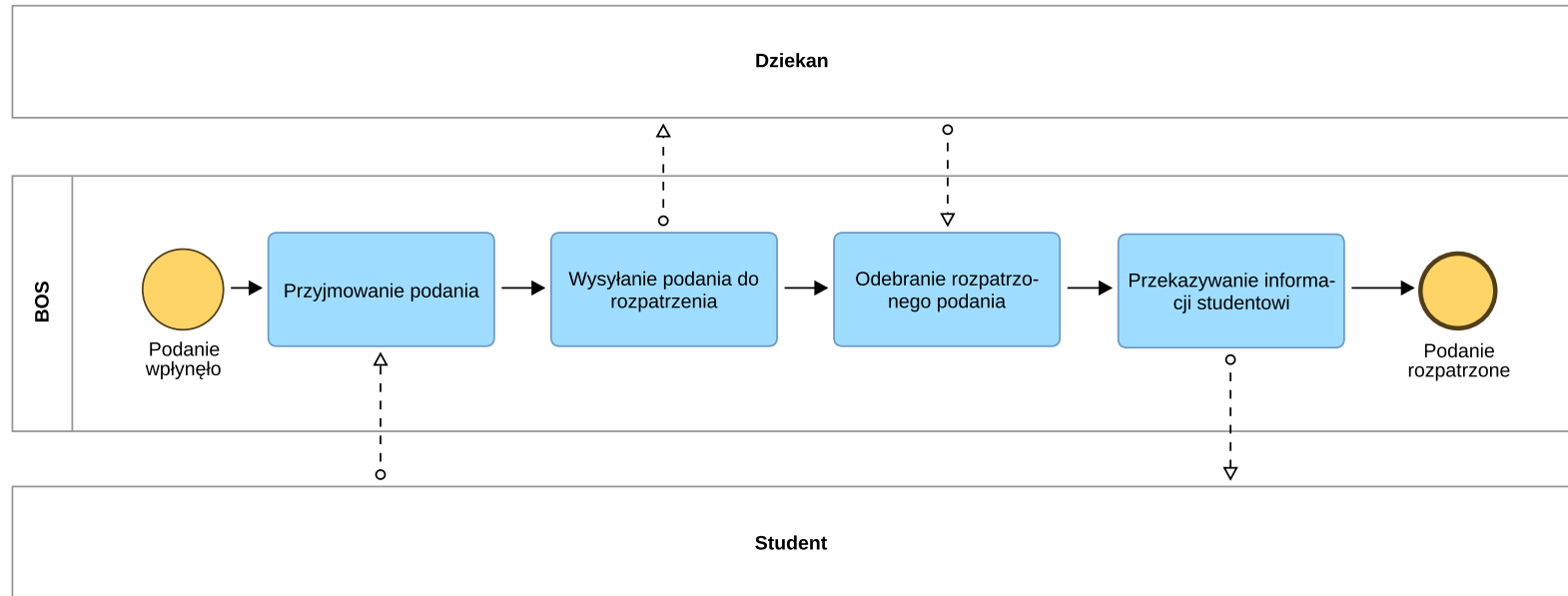
Jednostki organizacyjne Uniwersytetu (BOS, Dziekan) zamodelowane jako baseny (patrz s.14). Procesy wszystkich uczestników kooperacji prywatne, tzn. pokazują cykl pracy realizowany wewnątrz organizacji.

Procesy publiczne – przykład



Współpraca pomiędzy uczestnikami kooperacji (s. 19) pokazana jako procesy publiczne, widoczne tylko elementy odpowiedzialne za komunikację pomiędzy uczestnikami.

Procesy publiczne z perspektywy BOS



Współpraca pomiędzy uczestnikami kooperacji (s. 19) pokazana jako procesy publiczne z perspektywy BOS. Widoczne elementy procesu realizowanego przez BOS, które odpowiadają za komunikację, procesy pozostałych uczestników są ukryte.