

# SAG - Dokumentacja Projektu

## 1. Opis ogólny projektu

### 1.1. Temat projektu

Wykorzystanie agentów aktywnych w negocjacjach

### 1.2. Implementacja

System wspomagający proces wyboru aukcji i licytacji artykułów na portalu aukcyjnym

### 1.3. Platforma implementacji

JADE

### 1.4. Poruszany problem

System automatyzuje proces zakupu artykułu na portalu aukcyjnym odciążając tym samym klienta od czynnego udziału w aukcji. Automatyzacja zakupu odbywa się na dwóch etapach:

- a. Etapie wyboru aukcji dotyczących pożądanego produktu.
- b. Licytacji ceny (negocjacji cenowej) artykułu na wybranych w etapie a.

### 1.5. Cel

Celem systemu jest wylicytowanie (w imieniu klienta) produktu na portalu aukcyjnym, zgodnego z zapotrzebowaniem klienta, po najniższej możliwej cenie nie wyższej od zdefiniowanej przez klienta.

### 1.6. Opis

Środowisko portalu aukcyjnego jest środowiskiem wieloagentowym (MAS) o cechach heterogenicznych. W środowisku może istnieć wiele różnych agentów stworzonych przez wielu twórców. Jednakże każdy z tych agentów musi umieć posługiwać się standardowym dla tego środowiska protokołem komunikacyjnym (odszukiwanie aukcji, licytacja aukcji). Agenci zmuszeni są do konkutowania między sobą o wygrywanie aukcji.

Środowisko z punktu widzenia agentów ma charakter dynamiczny (środowisko zmienia się niezależnie od działań agenta), asynchroniczny (agent w dowolnym momencie może podjąć działanie) i deterministyczny (każda akcja agenta zmienia środowisko w przewidywalny sposób).

### 1.7. Użytkownik docelowy

- Klient, który chce wylicytować produkt o konkretnych parametrach
- Osoba analizująca zachowania rynkowe

## 2. Opis funkcjonalności projektu

- Możliwość złożenia preferencji dotyczących produktów jakie klient by chciał zakupić.

- Preferencje powinny mogą mieć granice stałe lub elastyczne
- Możliwość wystawienia ofert przez sprzedających na wielu serwisach
- Możliwość wyszukania oraz podjęcia licytacji wszystkich przedmiotów zgodnych z preferencjami klienta

### 3. Analiza wymagań niefunkcjonalnych

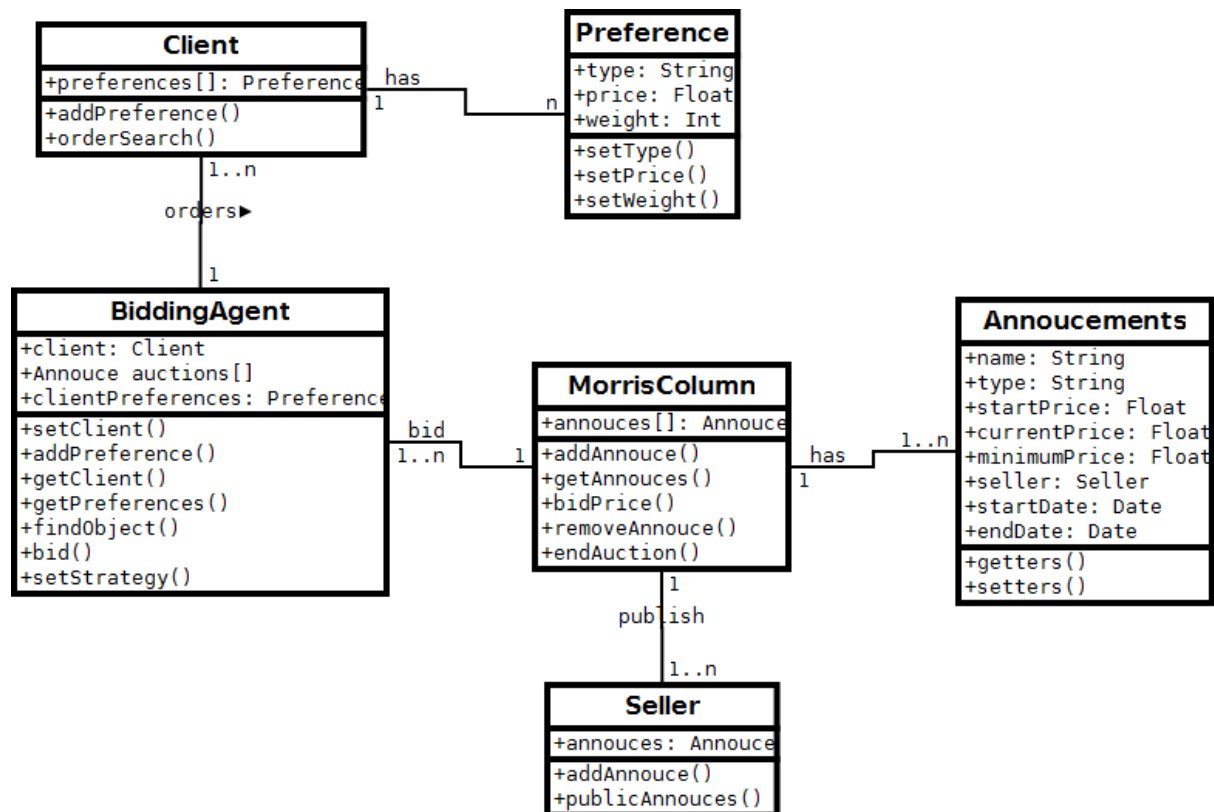
- Powinna być umożliwiona licytacja kilku produktów naraz
- Powinna być umożliwiona licytacja tego samego produktu na różnych aukcjach
- Powinno być umożliwione dobranie strategii licytacji

## 4. Opis architektury systemu

### 4.1. Moduł mechanizmu negocjacyjnych

- Ze względu na funkcjonalność można wyodrębnić 3 części systemu:
  1. Część kliencka – agent o pewnych preferencjach wraz ze swoim przedstawicielem w negocjacjach
  2. Część sprzedaży – agent wystawiający oferty na aukcję internetową
  3. Część informacyjna – agent reprezentujący kolumnę do zamieszczania oraz przeglądania informacji na temat aktywnych ofert. W zasadzie kolumna informacyjna może pełnić dwojaką funkcję:
    - a. Jest serwisem aukcyjnym (implementowany system pełni właściwie taką rolę)
    - b. Pełni rolę pośrednika między systemem agentowym a serwisem aukcyjnym – pobiera (synchronizuje się) systematycznie wszystkie aukcje z serwisu i udostępnia je licytującym agentom.

## 5. Szczegółowy podział komponentów



## 6. Scenariusze działania aplikacji obrazujące współpracę między komponentami

### 6.1. Lista aktorów

- **Client** – profil użytkownika dodającego swoje preferencje (implementowany jako agent)
- **BidderAgent** – agent wyszukiujący oferty i licytujący wg strategii
- **Seller** – agent wystawiający oferty
- **MorrisColumn** – kolumna informacyjna ze wszystkimi ofertami w serwisie aukcyjnym

### 6.2. Proces wystawienia ofert

1. *Sprzedawca tworzy ofertę o określonych parametrach*
2. *Oferta jest przekazywana do MorrisColumn*
3. *MorrisColumn udostępnia informacje o aktualnych ofertach i odmierza czas ich aktywności*

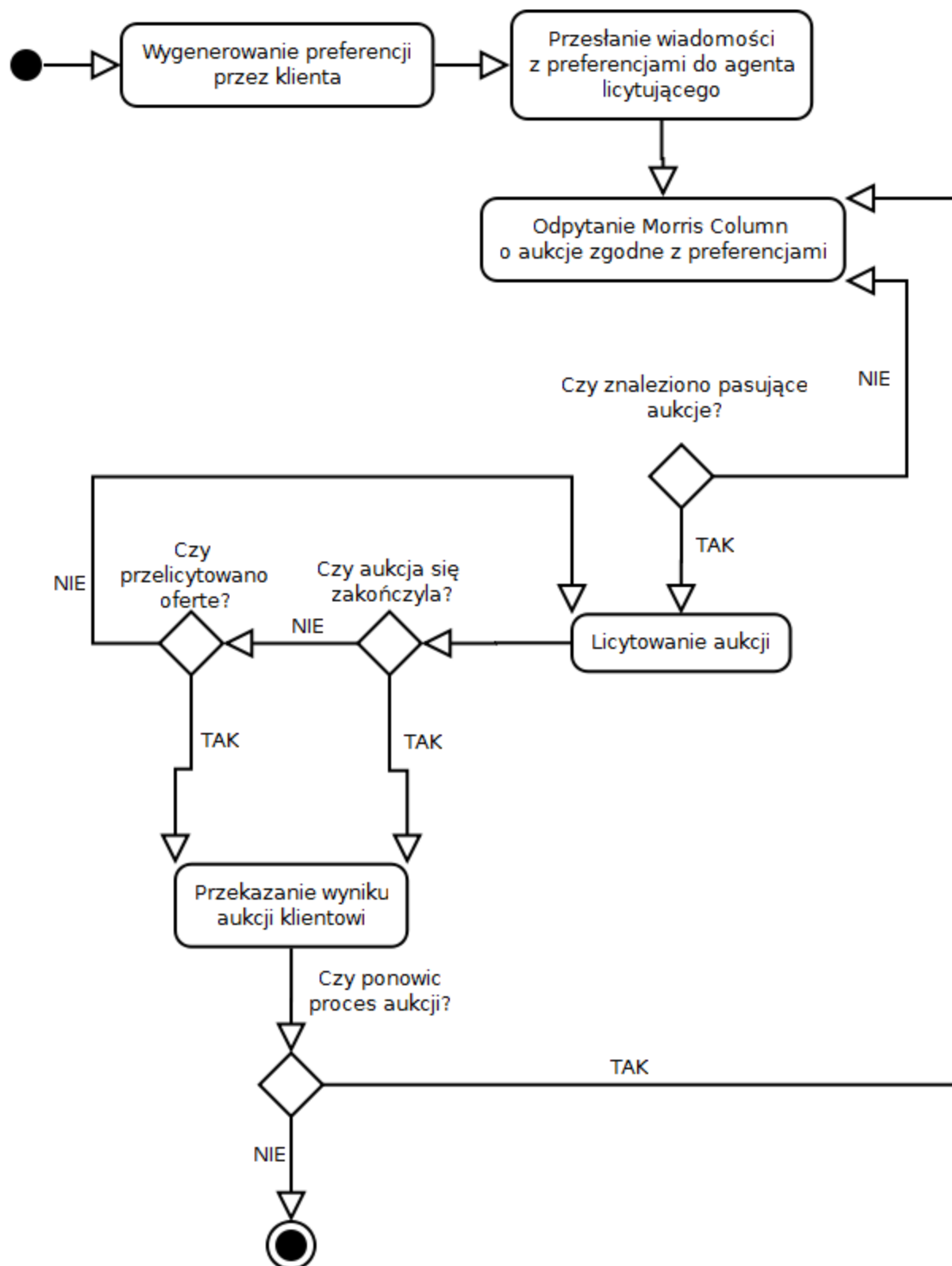
### 6.3. Proces wyszukiwania ofert

- 1. Klient tworzy profil preferowanego zakupu – definiuje jego atrybuty, maksymalną cenę za jaką jest w stanie kupić przedmiot.**
- 2. Przekazuje preferencje swojemu agentowi licytującemu**
- 3. Klient definiuje strategię jaką ma się kierować agent podczas licytacji**
- 4. Agent licytujący wyszukuje oferty na MorrisCoulumn (odpytuje) zgodne z przekazanymi mu parametrami**

#### **6.4.      Proces licytacji**

- 1. Agent licytujący rozpoczyna proces licytacji na każdej z aukcji z wyszukanych**
- 2. Licytowanie przebiega wg zdefiniowanej strategii**
- 3. Składana oferta nie może być mniejsza niż 1zł**
- 4. Zakończenie licytacji**
  - a.   Optymistycznie: czas aukcji dobiega końca, a oferta agenta jest najwyższą z proponowanych
  - b.   Pesymistycznie: czas aukcji dobiega końca, a oferta agenta nie jest najwyższą z proponowanych
  - c.   Pesymistycznie: cena minimalna sprzedającego nie jest osiągnięta

## 6.5. Diagram czynności opisujący przebieg aukcji



## 7. Opis strategii

- 7.1. **Od razu po przyłączeniu się do aukcji licytuje cenę narzuconą przez klienta**
- 7.2. **Licytuje co każdą zmianę wartości oferty zwiększając o 1zł**
- 7.3. **Odczekuje do końcowego pułapu czasu trwania aukcji i wtedy rozpoczyna licytację**

## 8. Opis komunikacji

- 8.1. **Klient przekazuje agentowi licytującemu wiadomość o performatywie informującej o jego preferencjach**
  - Preferencje muszą zostać przesłane jako wiadomość dla agenta.
  - Zawartość wiadomości jest Stringiem.
  - Aby licytujący agent mógł zrozumieć wiadomość, posiada swój parser preferencji, aby odczytać jaki produkt ma wyszukać a następnie do jakiej ceny może się przybliżyć i z jakiej strategii korzystać.
- 8.2. **BiddingAgent wraz z odebraniem wiadomości podejmuje wymianę informacji z kolumną informacyjną w celu ustalenia które aukcje będzie licytować**
  - W tym wypadku odebrane preferencje muszą zostać sparsowane na komunikaty implementowanego środowiska aukcyjnego.
- 8.3. **SellerAgent wysyła informacje o sprzedawanych produktach do MorissColumn. Wiadomości te również są parsowane i przechowywane jako kolekcja obiektów ogłoszeń (annoucements).**
- 8.4. **BiddingAgent po zakończeniu licytacji kieruje zapytanie do klienta czy ponowić cały proces wyszukiwania i licytacji.**

## 9. Sytuacje krytyczne

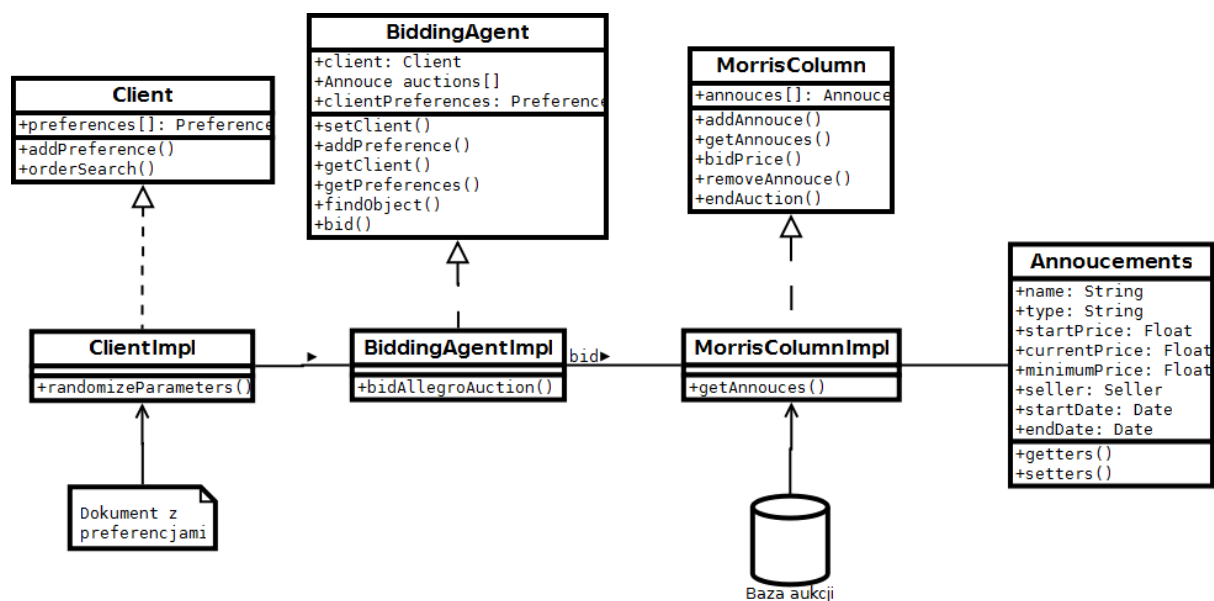
- 9.1. **Zarządzanie czasem – system działa asynchronicznie, zgłoszenia podbicia ceny aukcji przychodzą w różnym czasie. Gdy jednocześnie nadejdą 2 zgłoszenia są kolejgowane przez kolejkę wiadomości (zgodnie ze standardem ACL w JADE). Ważność zgłoszenia jest związana z czasem obsługi wiadomości (gdy nadejdzie czas obsługi zakolejkowanego zgłoszenia, a czas aukcji już się skończy, to zostanie ono odrzucone).**

## 10. Środowisko testowe

Środowisko testowe będzie miało za zadanie zasymulować przebieg licytacji wzorowanej na mechanizmie portalu allegro.pl.

Klient w celu symulacji jest agentem losującym swoje preferencje.

Informacje na temat aktywnych aukcji są dostarczane w postaci bazy danych Postgres.



### 10.1. Opis przykładów testowych

- Wygenerowanie takich preferencji klienta aby były wykorzystane różne strategie (w celu sprawdzenia efektywności)
- Wygenerowanie takich preferencji klienta z licytacjami o tej samej strategii
- Wystawienie przez sprzedającego kilku identycznych aukcji w celu sprawdzenia czy ceny końcowe będą różne

### 10.2. Struktury danych wykorzystywane w programie

#### 1. Dane wejściowe

- a. Baza danych z ofertami z serwisu aukcyjnego
- b. Plik z preferencjami dla agentów (plik jest w celu stworzenia takiego profilu preferencji agentów, aby istniała możliwość zaobserwowania interakcji między agentami).

#### 2. Dane wyjściowe

- a. Informacja o przebiegu aukcji oraz zwrot informacji dla klienta czy udało się wylicytować przedmiot o preferencji
  - 1. Jeżeli tak to generowane jest zapytanie czy kupować.
  - 2. Jeżeli nie to generowane jest zapytanie czy ponowić wyszukiwanie i licytacje.