

Apteka - Dokumentacja Wstępna

Michał Pióro Damian Baraniak

Projekt z PROI sem. 23L

Założenia projektu:

- Stan początkowy symulacji jest opisany w pliku konfiguracyjnym JSON, do którego ścieżka jest podawana jako argument wywołania
- W ciągu jednostki czasu farmaceuta jest w stanie wydać do 3 leków
- Każdy klient ma współczynnik prawdopodobieństwa chęci zamiany leków i przy każdym leku na jego podstawie jest losowane czy szukamy zamiennika.
- Jeśli zamienniki będą szukane to długość całej transakcji jest wydłużona o jednostkę czasu. Poszukiwanie zamienników odbywa się na początku transakcji.
- Pod koniec jednostki czasu losowane jest czy najdłużej pracujący farmaceuta który jest wolny idzie na przerwę i zamyka okienko, o ile pozostanie jedno otwarte okienko. Priorytet ma okienko o niższym indeksie
- Jeżeli w aptece jest zamknięte okienko ma ono szansę zostać otworzone
- Pod koniec jednostki czasu losowane jest czy do apteki przychodzi nowy klient.
- Jeden lek może być zmieniony jedynie na inny jeden lek, zamiana 1→2 czy 2→1 jest zakazana
- Przy zamianie leki są porównywane poprzez cenę i substancję aktywną (działanie)
- Prawdopodobieństwo przyjścia nowego klienta powinno być uzależnione od długości kolejki i liczby otwartych okienek
- Prawdopodobieństwo zamknięcia okienka jest uzależnione od czasu jego otwarcia
- Prawdopodobieństwo otwarcia okienka jest zależne od liczby klientów i liczby otwartych już okienek
- Apteka ma nielimitowaną ilość danego leku

Planowany przebieg symulacji:

1. Utworzenie apteki wraz z okienkami, początkową kolejką klientów oraz dostępnymi lekami
2. Transakcje określoną ilość razy
 - 2.1. Przypisywanie po kolei klientów do wolnych okienek
 - 2.1.1. Dobieranie ewentualnych zamienników
 - 2.1.2. Odczekanie wyliczoną ilość jednostek czasu
 - 2.1.3. Przekazujemy opis transakcji do **Logger**-a i zwalniamy okienko
 - 2.2. Losowanie zdarzeń:
 - 2.2.1. Otwarcie/zamknięcie okienek
 - 2.2.2. Przyjście nowych klientów
3. Zapisanie aktualnego stanu apteki i podsumowanie jej działalności

Podstawowe klasy:

- **Pharmacy** - klasa reprezentująca aptekę, w niej znajduje się kolejka klientów(**Queue**), kontener z lekami(**Inventory**), oraz okienka(**Counters**)
- **CountersList** - klasa przechowująca okienka pozwala na zarządzanie każdym z okienek(**Counter**)
- **Counter** - klasa reprezentująca dane okienko
- **ClientsQueue** - przechowuje referencje do klientów, zarządza kolejnością obsługi
- **Client** - klasa reprezentująca klienta, który posiada id, imię, nazwisko, listę leków(**ShoppingList**), to czy chce zakupy zrobić na fakturę czy paragon oraz prawdopodobieństwo chęci zmiany leku
- **ShoppingList** - klasa przechowująca referencje do leków znajdujących się w aptece(**Inventory**)
- **Inventory** - klasa odpowiadająca za grupowanie i przechowywanie dostępnych leków(**Medicine**)
- **Transaction** - Klasa odpowiedzialna za obsłużenie klienta, poprzez połączenie kolejki (**ClientsQueue**) z okienkami (**CountersList**)
- **Simulation** - Klasa odpowiadająca za przeprowadzanie symulacji
- **Medicine** - klasa odpowiedzialna za przechowywanie leku, czyli jego id, nazwy i tego czy jest na receptę. Przy wykorzystaniu poliformizmu powstanie kilka klas reprezentujących różne rodzaje leków
- **File Manager** - klasa odpowiedzialna za pobieranie pliku konfiguracyjnego, plików danych do losowania leków, nazw klientów, itp.
- **Logger** - klasa zapisująca stan symulacji w terminalu oraz w pliku podsumującym.

Wstępny podział zadań:

Michał	Damian
Medicine	Inventory
Transaction	Logger
ShoppingList	FileManager
Client	Counter
ClientsQueue	CountersList
Simulation	
Pharmacy	