

Dokumentacja wstępna – Szpital

Oliwier Szypczyn, Andrzej Tokajuk

1. Temat projektu

Tematem projektu jest szpital. Jest on podzielony na oddziały, w których znajdują się sale. Pracują w nim pracownicy, którzy pełnią różne role m. in lekarze, pielęgniarki i ratownicy medyczni. W szpitalu hospitalizowani mogą być pacjenci z różnymi schorzeniami, którzy wymagają zróżnicowanego leczenia.

2. Klasy wykorzystane w projekcie:

- ☐ Symulacja
- ☐ Szpital
- ☐ Miejsce
 - ☐ Oddział
 - ☐ SOR/Recepcja
 - ☒ Dyspozytornia karetek
 - ☐ Sala
 - ☐ Ogólna
 - ☐ Gabinet
- ☒ Człowiek
 - ☒ Pacjent
 - ☒ Lekarz
 - ☒ Pielęgniarka
 - ☒ Ratownik medyczny
- ☒ Karetka
- ☒ Usługa medyczna
 - ☒ konsultacja
 - ☒ operacja
- ☒ ~~Baza danych wszystkich usług medycznych~~
- ☒ Karta zdrowia
- ☐ Kolejka
- ☐ Generator liczb losowych

3. Opis klas:

→ Karta zdrowia:

◆ pola:

- pesel pacjenta - (identyfikator)
- choroby
- zaplanowane usługi medyczne

◆ metody:

- dodawanie i "uleczanie" chorób
- sprawdzanie czy jest dana choroba
- planowanie i kończenie usług
- sprawdzanie czy jest dana usługa

→ Usługa medyczna:

◆ pola:

- ID usługi
- całkowity planowany czas trwania
- aktualny czas wykonania
- lista lekarzy
- lista pielęgniarek
- pacjent
- stan usługi
- bool czy na NFZ?

◆ metody:

- gettery, settery
- operator inkrementacji czasu usługi
- dodanie pracownika
- liczenie kosztu
- sprawdzanie czy jest poprawny personel
- rozpoczęcie usługi
- kontynuacja usługi
- koniec usługi
- zwracanie pacjenta z usługi
- zwracanie personelu z usługi

→ Baza danych usług:

◆ pole:

- lista usług

◆ metody:

- planowanie usługi
- usuwanie usługi zrealizowanej
- wyszukiwanie usługi
- obliczanie kosztu wszystkich usług

→ Kolejka:

◆ pola:

- numerki
- pacjenci
- sala (identyfikator)

◆ metody:

- gettery, settery
- dodawanie pacjentów i usuwanie (queue)
- przydzielanie pacjentom numerków

→ Karetka:

◆ pola:

- ratownicy medyczni
- pacjent
- numer rejestracyjny
- całkowity planowany czas jazdy
- aktualny czas wykonania interwencji
- bool czy jest w akcji czy nie
- stan w którym się znajduje

◆ metody:

- gettery, settery
- jedź na interwencję
- kontynuuj interwencję
- wróć z interwencji
- dodaj personel
- zwróć personel
- zwróć pacjenta

→ Człowiek:

◆ pola:

- PESEL - (identyfikator)
- Imię
- Nazwisko
- Płeć
- Wiek
- czy stan uległ zmianie (globalnie dla symulacji)
 - Pacjent
 - ◆ Karta zdrowia
 - ◆ Stan w którym aktualnie jest
 - Lekarz
 - ◆ Specjalizacja (enum)
 - ◆ stan w którym jest
 - ◆ stawka godzinowa
 - Pielęgniarka
 - ◆ stan w którym jest
 - ◆ stawka godzinowa

- Ratownik Medyczny
 - ◆ stan w którym jest
 - ◆ stawka godzinowa
- ◆ metody:
 - gettery i settery
 - sprawdzenie czy stan uległ zmianie
 - zmiana stanu
 - operatory porównania

→ Szpital:

- ◆ Pola
 - Nazwa
 - Lista "miejsc"
 - Pacjenci
 - personel
 - usługi medyczne
- ◆ metody:
 - manipulacja pacjentami
 - manipulacja personelem
 - manipulacja usługami

→ Miejsca:

- ◆ pola:
 - Oddziały
 - Nazwa
 - Lekarze
 - Sale
 - ◆ Numer (identyfikator)
 - Ogólna:
 - Liczba łóżek
 - Pacjenci
 - Gabinet:
 - Pracownicy
 - Kolejka
 - pacjent
 - Rodzaj badania
 - SOR/Recepcja
 - kolejka
 - pielęgniarki
 - Dyspozytornia Karet
 - Karetki
 - Ratownicy medyczni

→ Symulacja:

- ◆ lista pacjentów szpitala
- ◆ data i czas
- ◆ metody
 - zarządzanie stanem ludzi ze względu na ich stan
 - czytanie statusu człowieka

[Link do murala z reprezentacją UML](#)

4. Opis założeń:

- szpital jest tworzony na podstawie pliku w formacie json i jego stan (liczba sal, personelu itd.) nie będzie zmieniany w trakcie działania programu
- w argumencie wywołania programu będzie podany okres działania symulacji, plik do odczytu szpitala i do zapisu przebiegu symulacji
- symulacja będzie iterować co 15 minut
- symulacja będzie losowała czy i ile w danym takcie ma przyjść do recepcji/przyjechać karetką pacjentów
- symulacja będzie posiadać dostęp do listy wszystkich pacjentów/pracowników i za każdym wykonaniem pętli będzie czytać odpowiednie pola i przemieszczać pacjentów/pracowników po szpitalu oraz generować opisy (tylko osoby których stan uległ zmianie)
- pacjenci którzy przychodzą do recepcji to pacjenci, którzy mają już kartę zdrowia, są umówieni na konkretne badania, operacje
- pacjenci którzy przyjeżdżają karetką nie wiedzą na co chorują najpierw muszą iść na konsultację, żeby dowiedzieć się na co chorują
- Symulacja i klasy w których jakaś czynność będzie korzystać z liczb losowych będzie korzystać z klasy -> generator liczb losowych

5. Podział obowiązków:

1. Oliwier

- Człowiek -> Pacjent/Lekarz/Pielęgniarka/Ratownik
- Dyspozytornia karetek
- Karetka
- Karta zdrowia
- Usługa medyczna -> Konsultacja/Operacja

2. Andrzej

- Sala
- Oddział
- Szpital
- Recepcja
- Kolejka

Symulacja i obsługa plików zostaje na koniec i jeszcze nie wiemy, kto konkretnie zajmie się czym aby całość mogła działać sprawnie. Najpierw chcemy skupić się na fizycznych elementach szpitala.