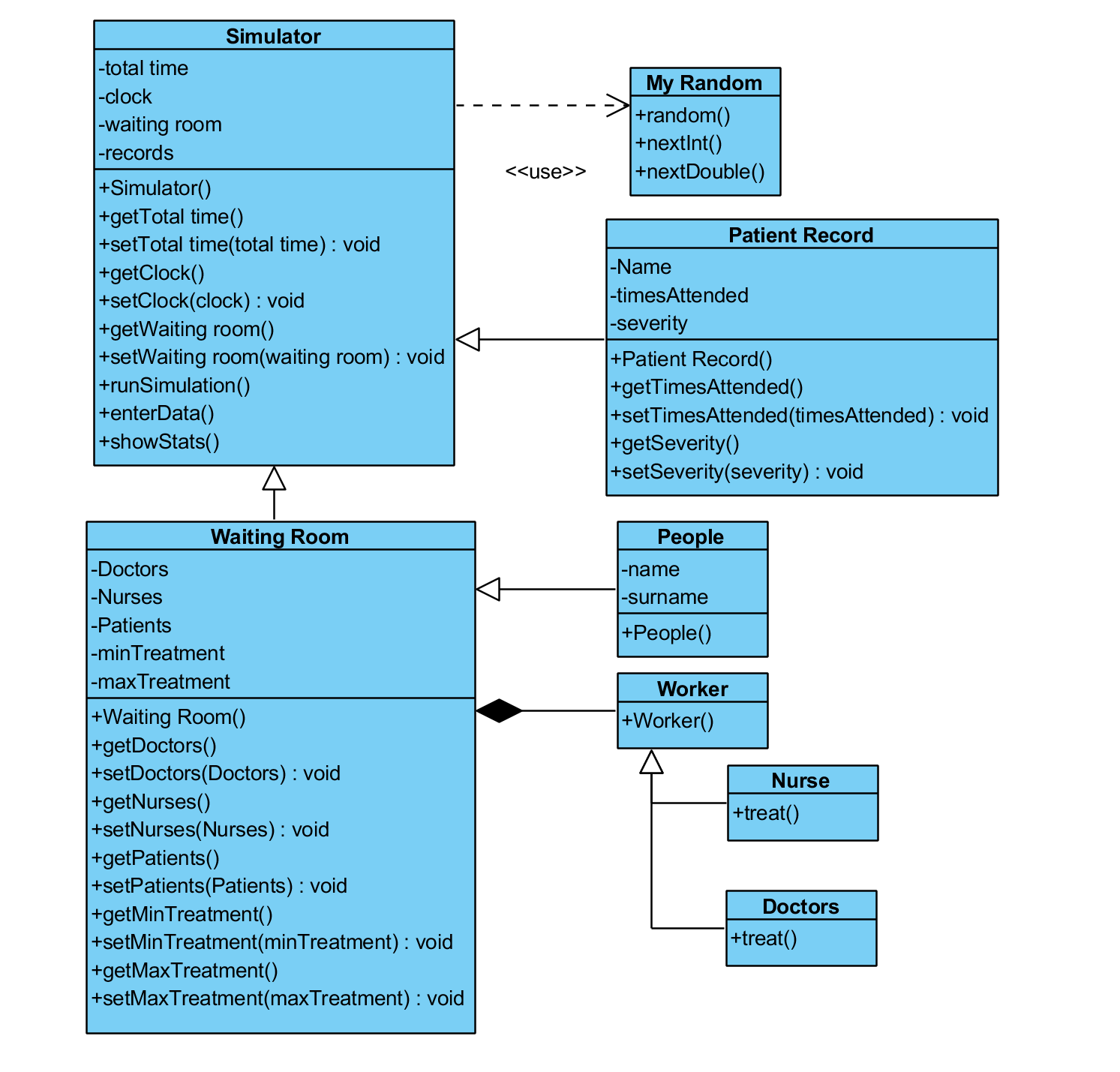
Mateo Reynoso  
Data Structures  
2018-07-18

**Simulation project UML and specifications**

The simulation selected is the hospital simulation. Basically, a person from 273 Ville gets randomly selected according to the rate of persons per minute. They get assigned an number according to the severity of the illness, then an array holding doctors and nurses will be in charge of their treatment, which will vary according to the user input.

**Requirements and specifications:**

The base class will be called simulator, which will be update according to the internal clock of the program. Here the waiting room class is located, which will be in charge of assigned ill people a doctor or nurse. Then, it will proceed to create a register object, which will hold all the patients treated. The requirements are that the user will be allowed to input the number of doctors and nurses, the time of the simulation and the arrival rate.  
 The main will run the function simulation, where the internal clock will be incremented in one by one. This way, some random number generators can be used to select the patients, the times and the kind of illness. The other class will serve the purpose of providing the simulator the information needed and containing the queue of patients. I was also possibly thinking of using a hash table for the patients so they can be ordered in terms of the severity of their problem.

**UML:**

**User Cases:**

The main operations the user will make are introducing the parameters for the simulation and requesting the data. The input for the simulation will be the duration of the simulation, the rate of patients and the number of personnel in the hospital. On the other hand, the user will be able to request the names of all the people treated in the hospital and request the history of any person.

**Unit tester cases:**

For now, the test cases I can think of, will be for the patient record class, to make sure that its registering the information correctly. On the other hand the tester could be used to check how the treatment works, to see if the queue of patients gets emptied.

**Pseudo-code:**

* **How the code will run in main**

Loop clock, until max time   
 - input  
 - update waiting room  
 - show stats

* **Pseudocode for the waiting room will clean the waiting queue**

Update waiting room (clock)

* If random < rate

make new patient

add to the queue

* If doctors or nurses free

Pop front queue

Add patient history to the records

* Then update doctors and nurses