Group 14

Author: Zhang Sihan

Software Requirements

Proj.3 Painkiller Injection System

Table of Contents

[System Objective 2](#_Toc44183870)

[Domain Analysis 2](#_Toc44183871)

[System Architecture 3](#_Toc44183872)

[Use Cases 4](#_Toc44183873)

[Software Requirements 5](#_Toc44183874)

[R1: PatientUI 5](#_Toc44183875)

[R2: PhysicianUI 5](#_Toc44183876)

[R3: InjectionDB 5](#_Toc44183877)

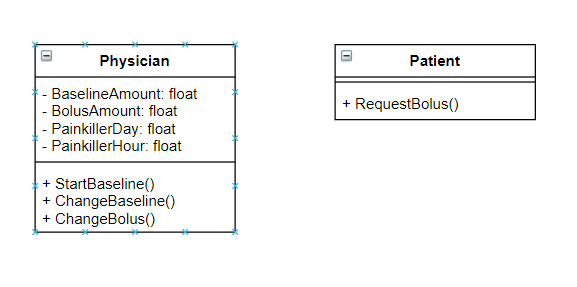
[R4: InjectionProcessor 5](#_Toc44183878)

## System Objective

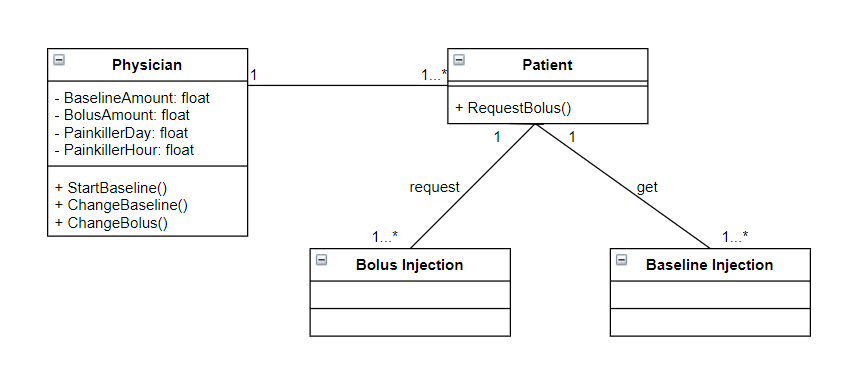
In this project, we are going to develop and design a painkiller injection system that gives patient painkiller after surgery with a limit in amount during some certain period. By providing interconnected interfaces to patients and physicians in the hospital, the system can reduce delays and the number of errors during injections, which can improve patients’ satisfaction and comfort.

## Domain Analysis

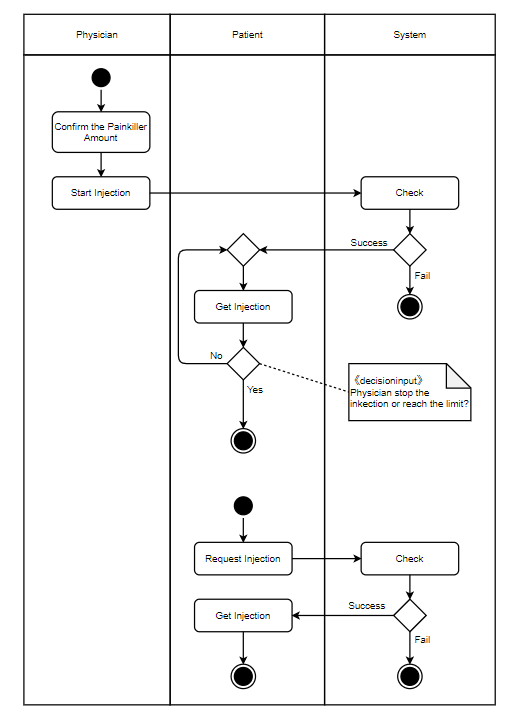
The participants of the project are physician and patient.



The relationships among different participants in the system are shown as follows:

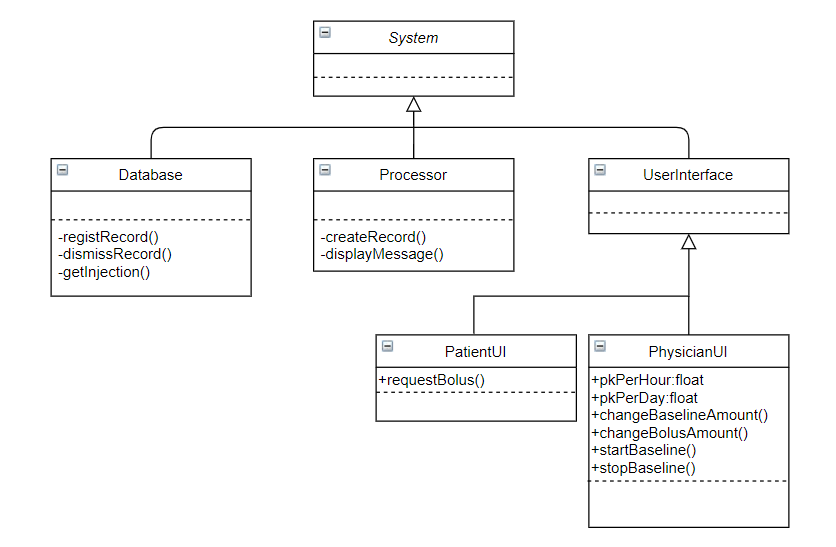


Here is the sequence of getting a baseline injection and getting a bolus injection:



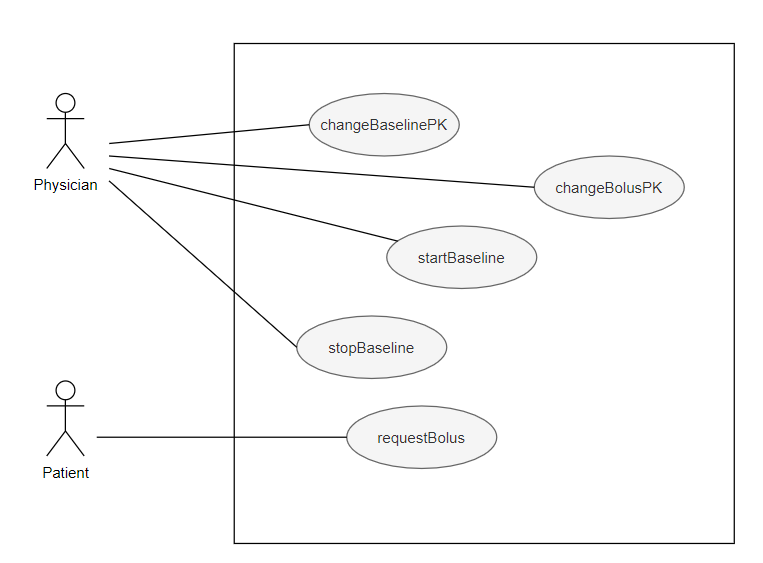
## System Architecture

From the information above, we will design a software system that allows the patient to request bolus injection on a hand-held device and transmitted to the system. The physician can adjust the painkiller amount of bolus injection and baseline injection, and whether to give a baseline injection. Physician can also see how much painkiller patient has got in one hour and during this day. The system architecture is shown below:



## Use Cases

The system can achieve the following use cases from the patient’s and the physician’s perspectives:



## Software Requirements

### R1: PatientUI

* R1.1: The patient should be able to request a bolus injection on his UI.
  + R1.1.1: The patient should be able to know whether request successfully.
  + R1.1.2: The patient should be able to see the reason why request fail.

### R2: PhysicianUI

* R2.1: The physician should be able to give a baseline injection on his UI.
  + R2.1.1: The physician should be able to know whether inject successfully.
  + R1.1.2: The physician should be able to see the reason why injection fail.
* R2.2: The physician should be able to adjust the painkiller amount for both ways of injection.
  + R2.2.1: The physician should be able to see the painkiller amount for both ways of injection in time.
  + R2.2.2: The physician should not be able to set the painkiller amount exceeding the limit.
* R2.3: The physician should be able to see the painkiller amount that the patient has got in one hour and during one day.
  + R2.3.1: The painkiller amount should be updated instantly.
  + R2.3.2: The painkiller amount should be reset when after an hour and a day.
* R2.4: The physician should be able to see the injection record.

### R3: InjectionDB

* R3.1: InjectionDB should store the injection record of the patient.
  + R3.1.1: InjectionDB should be able to register a new injection record.
  + R3.1.2: InjectionDB should be able to dismiss all the record when one day passed.
* R3.2: InjectionDB should be able to start an injection according to the record.
  + R3.2.1: InjectionDB should be able to correctly stop injection according to the injection ways and limit.
  + R3.2.2: InjectionDB should be able to stop baseline injection when physician stop the process.

### R4: InjectionProcessor

* R4.1: InjectionProcessor should be able to deal with the input from physician and patient.
  + R4.1.1: InjectionProcessor should be able to deal with the illegal input and show the message.
  + R4.1.2: InjectionProcessor should be able to create a new injection record according to the input.