

# **Painkiller Injection System**

## **Requirements**

student xxxx

# Table of Contents

Introduction & Rules

System composition

Sequence analysis

Use cases

Software Requirements

Detailed explanation of properties

## Introduction & Rules

In this project, we simulate a program, taking care of patients who need painkillers after surgery.

Very importantly, we should provide robustness, as minor carelessness could result in serious consequences in hospitals.

To reduce pain, we automatically inject painkillers slowly for patients.

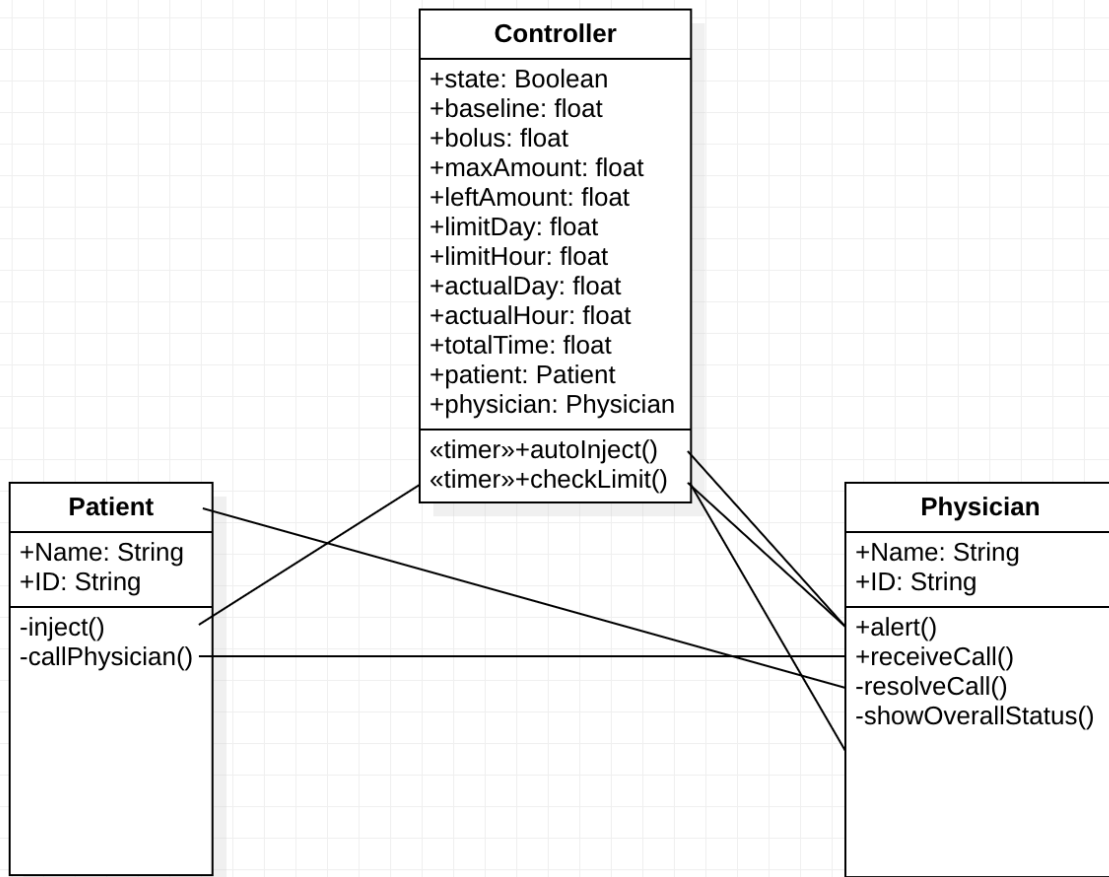
For fear of sudden incidents, patients have buttons for a single dose of painkillers or an alarm to physicians.

If patients push the injection button too often, physicians should receive an alarm to check patients out.

Professional physicians need to replace the type/amount of painkillers to achieve the best postoperative rehabilitation.

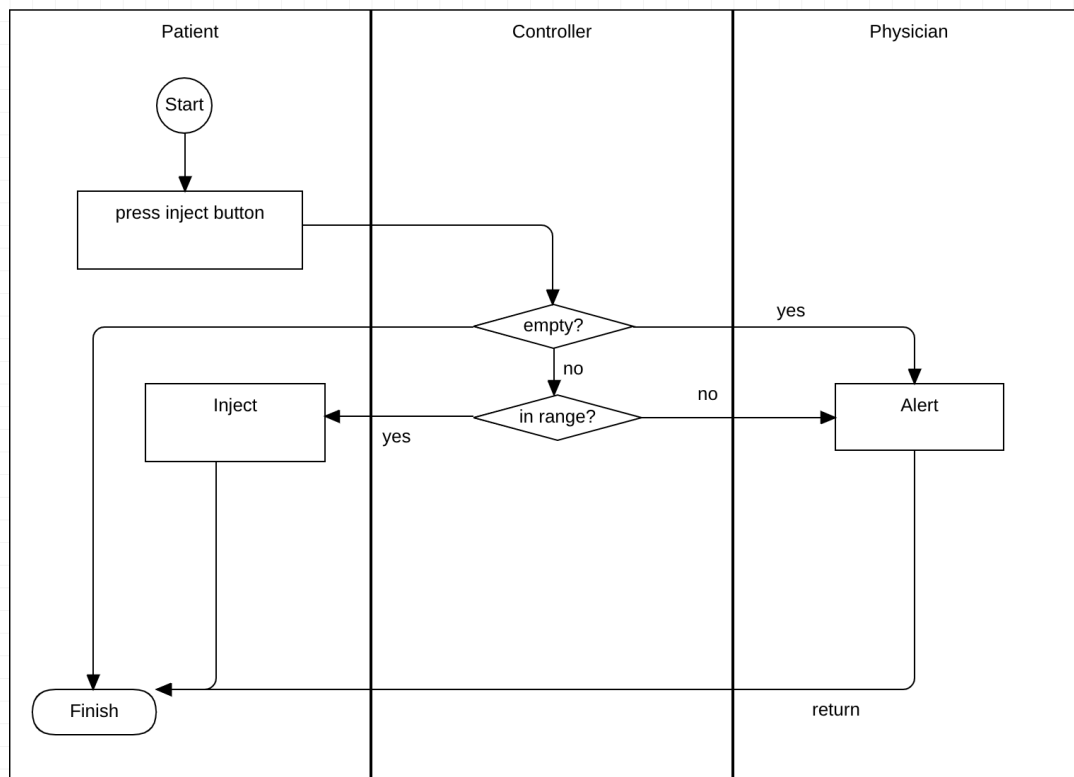
However, we shall notice that people make mistakes. For physicians, if they set the amount to an obviously inappropriate (high) level, the system should reject the changes to keep patients safe.

# System Composition

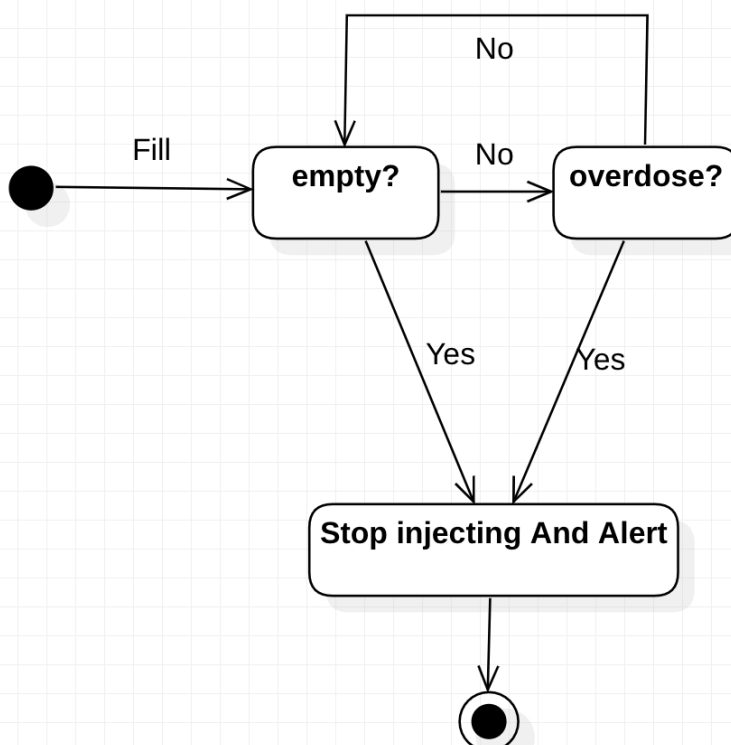


# Sequence analysis

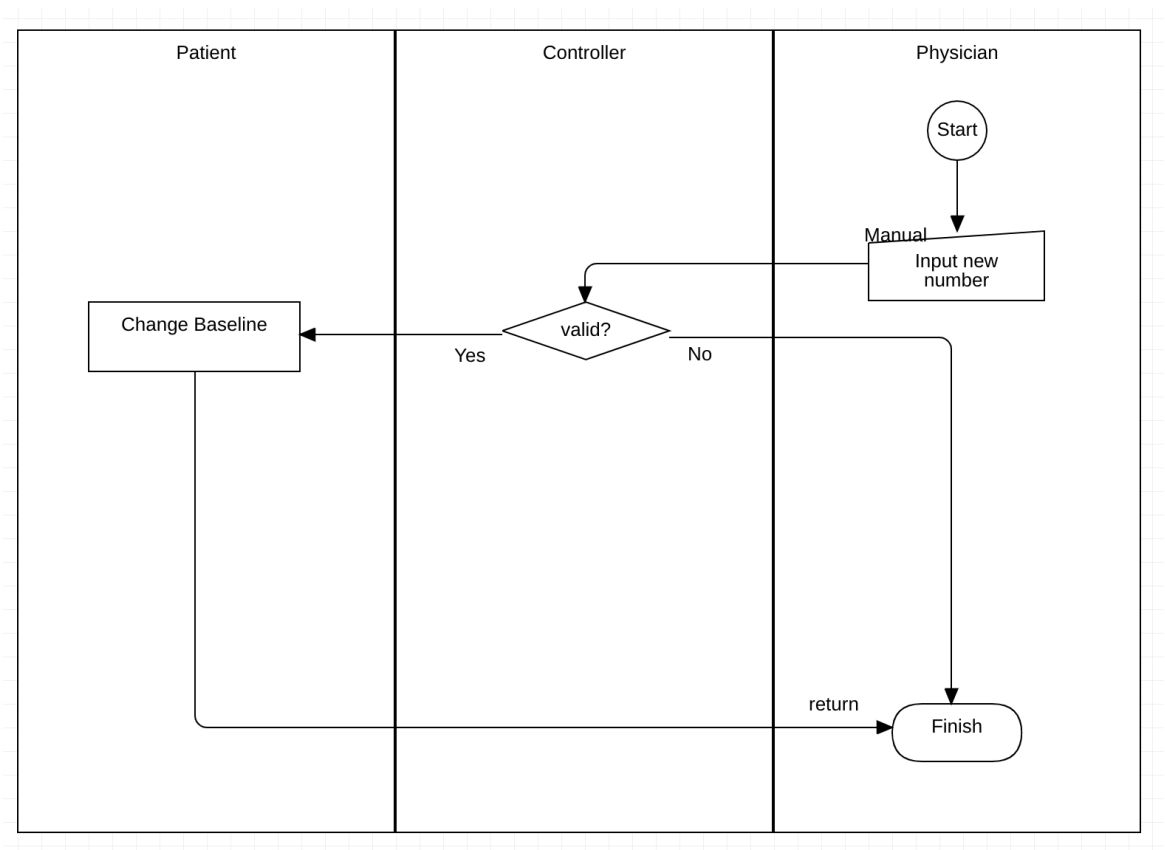
Patients giving themselves a single bolus:



Check whether the drug amount is valid:

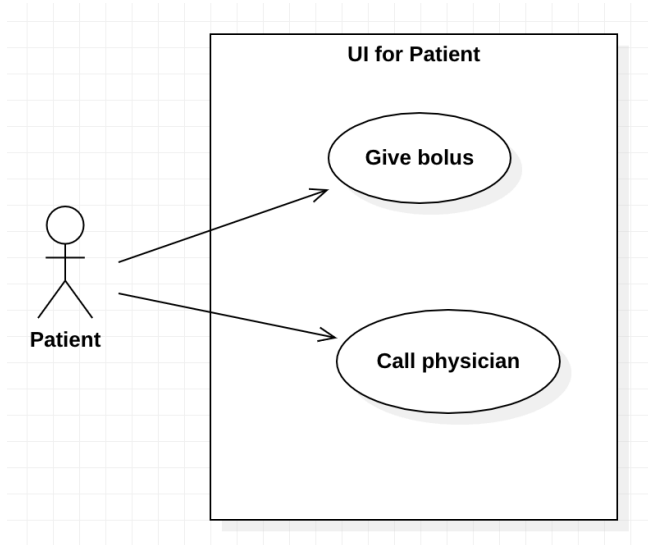


Physicians setting a new value (as baseline):

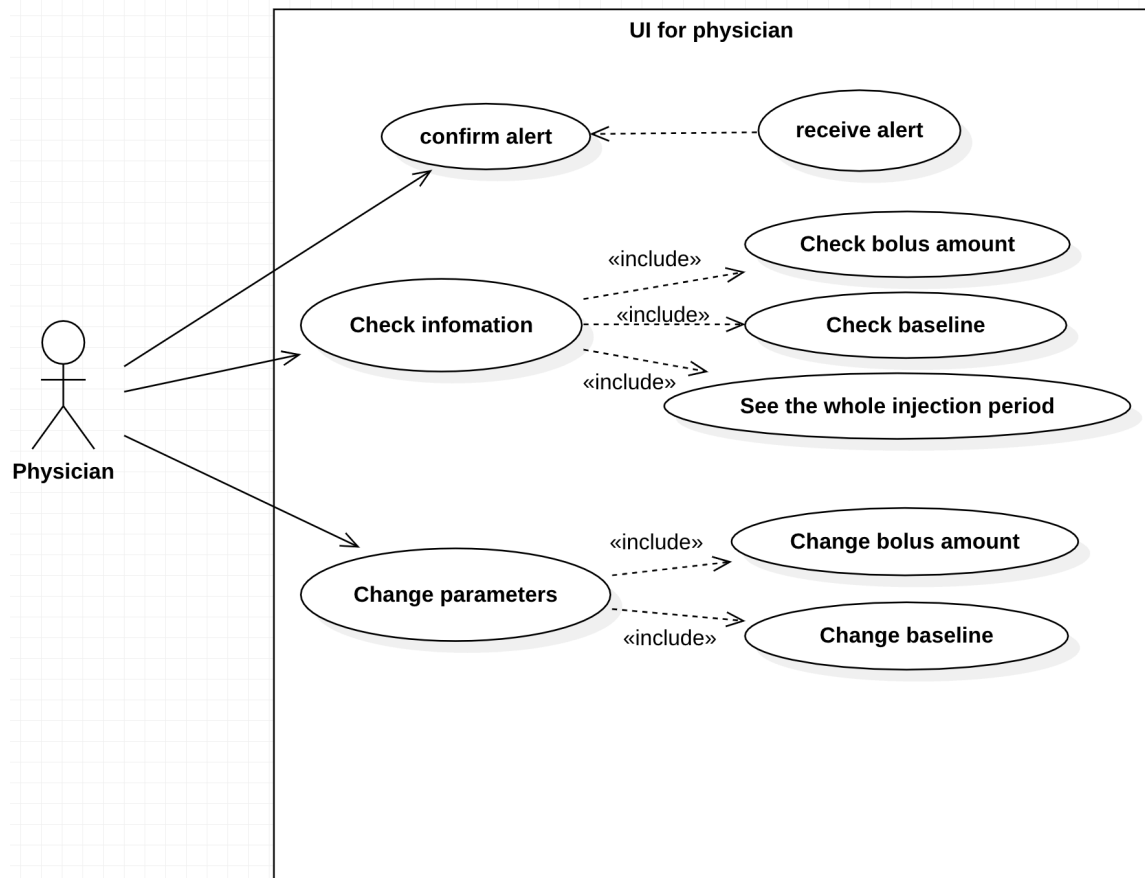


# Use cases

For patients:



For physicians:



# Software Requirements

## R1: Patient UI

### R1.1: Patients can inject a bolus by themselves

R1.1.1: Bolus should not be given if drug overdose in the past 24h.

R1.1.2: Bolus should not be given if drug overdose in the past hour.

R1.1.3: Bolus should not be given if no drug is left in the machine.

R1.1.4: Bolus should be given if nothing unusual happens.

### R1.2: Patients can call physicians

## R2: Physician UI

### R2.1: Physicians can check the status of patients

R2.1.1: Physicians can check bolus amount at any time.

R2.1.2: Physicians can check baseline at any time.

R2.1.3: Physicians can check the time of events in the whole injection process.

### R2.2: Physicians can modify the status of patients

R2.2.1: Physicians can modify the bolus amount to an appropriate value.

R2.2.2: Physicians can modify the baseline to an appropriate value.

R2.2.3: Physicians cannot modify the bolus amount to an inappropriate value.

R2.2.4: Physicians cannot modify the baseline to an inappropriate value.

## R3: Controller

### R3.1: The machine can do auto works correctly

R3.1.1: Auto inject is stable.

R3.1.2: Alert will be sent to a physician if the machine goes empty.

R3.1.3: Alert will be sent to a physician if the patient is overdosing.

### R3.2: The machine updates its information to UI automatically.



# Detailed explanation of properties

## 1. Patient

### 1.1. Name

### 1.2. ID

IDs are unique for patients.

## 2. Physician

### 2.1. Name

### 2.2. ID

IDs are unique for physicians.

## 3. Controller

### 3.1. State

0 indicates that the machine is off, and 1 means on.

### 3.2. baseline

The amount of drug automatically injected in 0.1 minutes.

### 3.3. bolus

The amount of drug injected in if a bolus is given.

### 3.4. maxAmount

The max amount of drug that the machine can store.

### 3.5. leftAmount

The amount left in the machine.

### 3.6. limitDay

The maximum amount of drug that the patient should get in 24h.

### 3.7. limitHour

The maximum amount of drug that the patient should get in 1h.

### 3.8. actualDay

The amount of drug that the patient has actually gets injected in 24h.

### 3.9. actualHour

The amount of drug that the patient has actually gets injected in 1h.

### 3.10. totalTime

The total length of time since injection started.

### 3.11. patient

A related Patient object.

### 3.12. physician

A related Physician object.

## Detailed explanation of methods

### 1. Patient

#### 1.1. Inject()

This function is called automatically by a timer.

#### 1.2. callPhysician()

This function is called manually by patients, it will set a signal to the physician.

### 2. Physician

#### 2.1. Alert()

This function is called by the controller, lighting up a red lamp on the UI.

#### 2.2. receiveCall()

This function is called by the patient when the patient is calling the physician.

#### 2.3. resolveCall()

This function is called by physician manually.

#### 2.4. showOverallStatus()

This function is called automatically to update information on the UI.

### 3. Controller

#### 3.1 autoInject()

The function is called by a timer, calculating real-time drug consumption.

#### 3.2 checkLimit()

The function is called by a timer, detecting drug overdose.