Group 7

Author: Jintong Luo

SOFTWARE SPECIFICATIONS

Painkiller Injection System

Table of Contents

[System Architecture 2](#_Toc169091444)

[Software Specifications 3](#_Toc169091445)

[S1: Injection Processor UI Implementation 3](#_Toc169091446)

[S2: CheckPIN UI Implementation 8](#_Toc169091447)

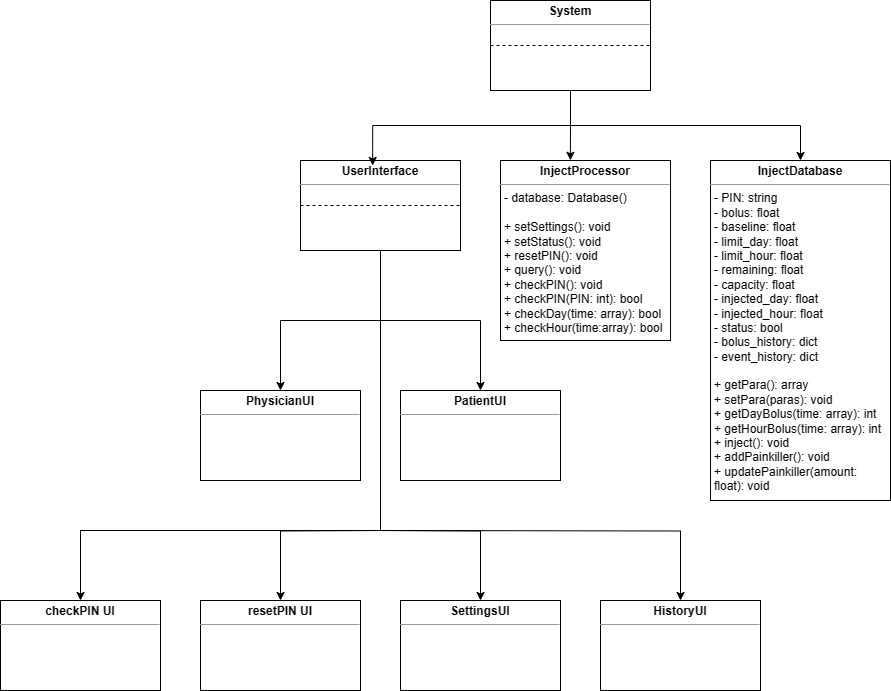
[S3: ResetPIN UI Implementation 9](#_Toc169091448)

[S4: SettingsUI Implementation 10](#_Toc169091449)

[S5: PatientUI Implementation 12](#_Toc169091450)

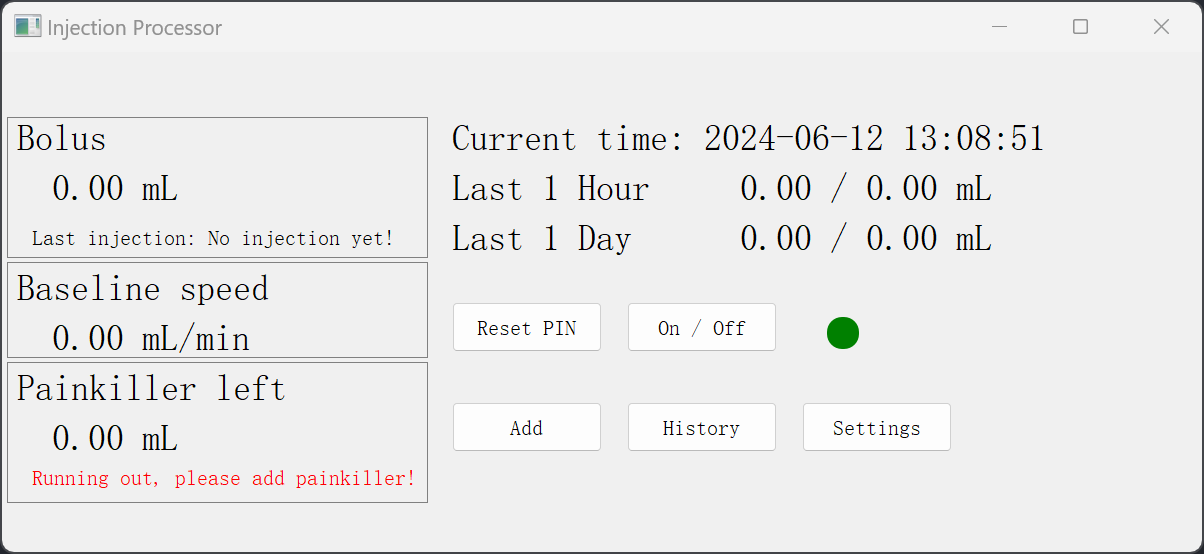
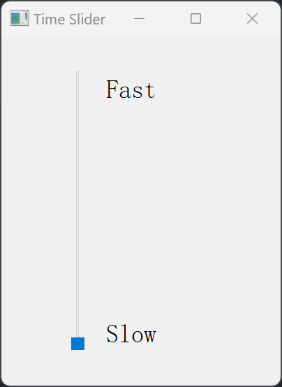
## System Architecture

The system architecture is shown below:

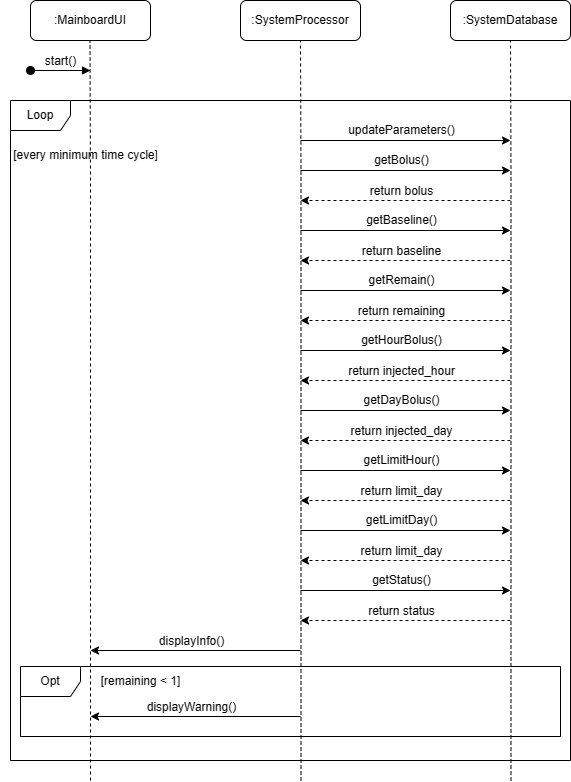


Software Specifications

S1: Injection Processor UI Implementation

*S1.1: Display Basic Information*



* S1.1.1: Run the System, Initialize and Update the Data

1. Run the system processor
2. Initialize the time as the system time and the speed of time current as the reality, then the timer would update the data every second.
   1. If controlling the time slider, the update interval would change.
   2. The time slider is only used for testing, so it would not be included in the physical machine, i.e. main board.
3. All the parameters should be updated based on the database during the update operation.
4. The main board should operate correctly and deal with possible hazards.
   1. If the main board calls out a Dialog, the buttons on main board except “Add” would be disabled.

* S1.1.2: Display Some Basic Data on the Main Board

1. Display bolus, baseline speed, remaining painkiller, the painkiller injected in the last one hour, the painkiller injected in the last one day, the limit of one hour, the limit of one day and inject status on the main board.
2. All the parameters should be obtained from the system database via the processor.
3. The parameters should be updated in every set time interval.
4. Display the current time and the last time to inject painkiller.
   1. If having not injected yet, just display “No injection yet!”.
   2. If there exists injection before, the accurate time should be displayed.

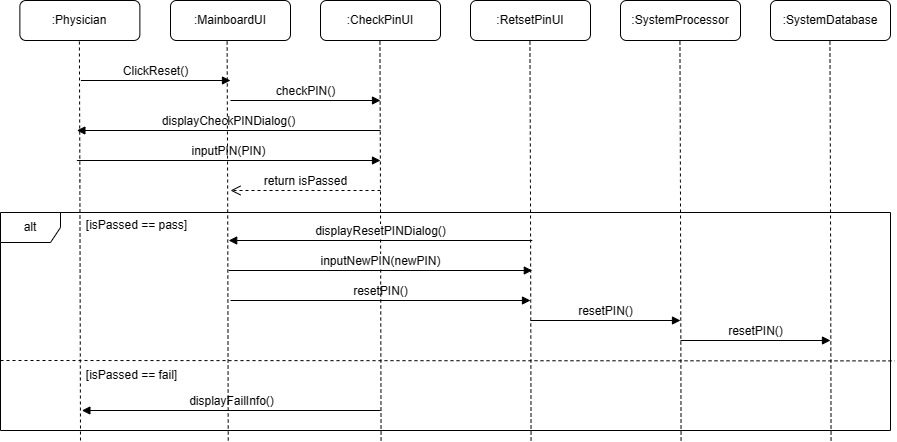
* S1.1.3: Display the baseline status and Control the Status Light

1. Display the baseline status on the main board.
   1. If the baseline is on, the light should be green.
   2. If the baseline is off, the light should be grey.

* S1.1.4: Display the Remaining Painkiller

1. Display the amount of remaining painkiller on the main board.
   1. If the amount is too low, i.e. less than 1 mL, the main board would display a warning, “Running out, please add painkiller!”.
   2. Otherwise, there would not be any indicator.

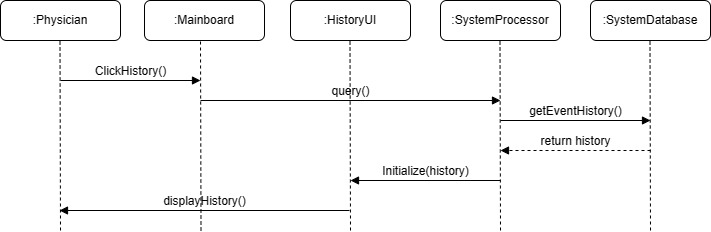
*S1.2: Reset the PIN*

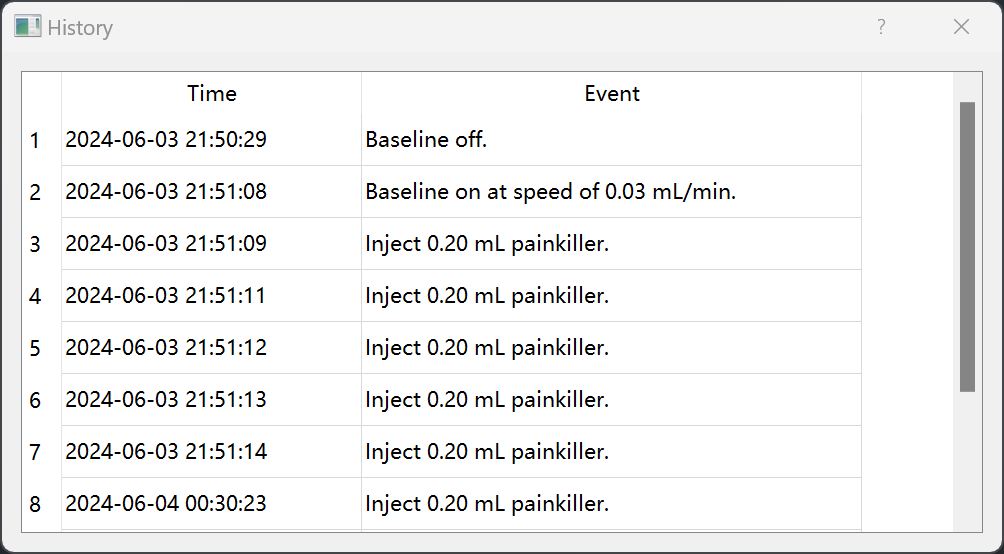


1. Click the “Reset PIN” button.
2. Call out a CheckPIN Dialog to check the identification. The physician could enter the PIN.
   1. If pass the check, a ResetPIN Dialog would be displayed then the processor would reset the PIN in database.
   2. If fail the check, it would display a warning and the physician could not reset the PIN.

P.S. The details of ResetPIN UI are included in S3.

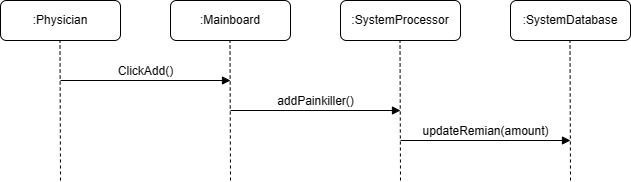
*S1.3: Query the Events History*





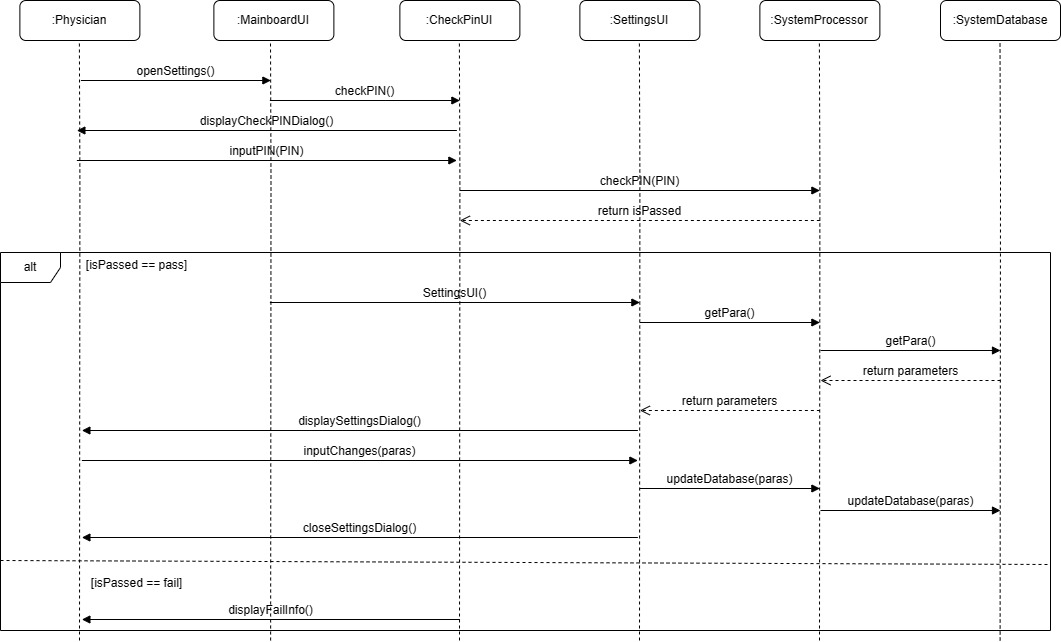
1. Click the “History” button.
2. The processor would obtain history dictionary from the database.
3. There would be a History Dialog to display the events and corresponding time.

*S1.4: Add Painkiller*



1. Click the “Add” button.
2. The processor would update the amount of remaining painkiller in the system database and add painkiller in the physical container.

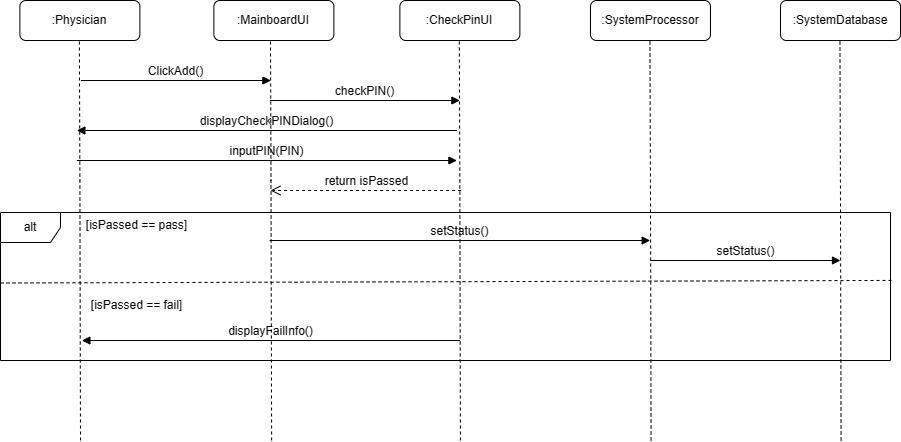
*S1.5: Get & Change the Settings*



1. Click the “Settings” button.
2. Call out a CheckPIN Dialog to check the identification. The physician could enter the PIN.
   1. If pass the check, the processor would obtain corresponding parameters from database and display them in the Settings Dialog. In the Settings Dialog, the physician could set new parameters and the processor would update them in the database.
      1. If the new parameters are valid, the parameters would update.
      2. If there exists invalid parameter, the Dialog would not close and the physician should reset the parameters.
   2. If fail the check, it would display a warning.

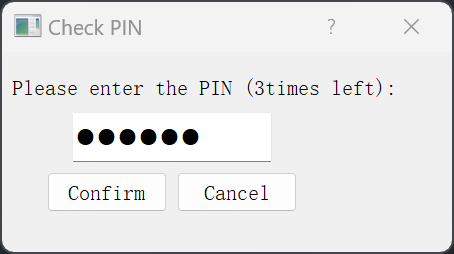
P.S. The details of Settings UI are included in S4.

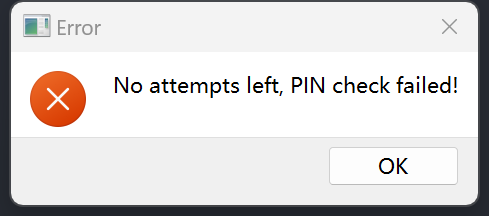
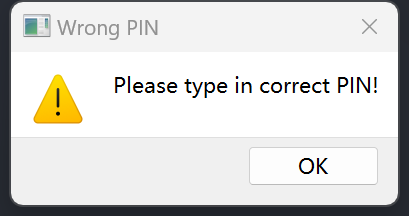
*S1.6: Control the Baseline Status*

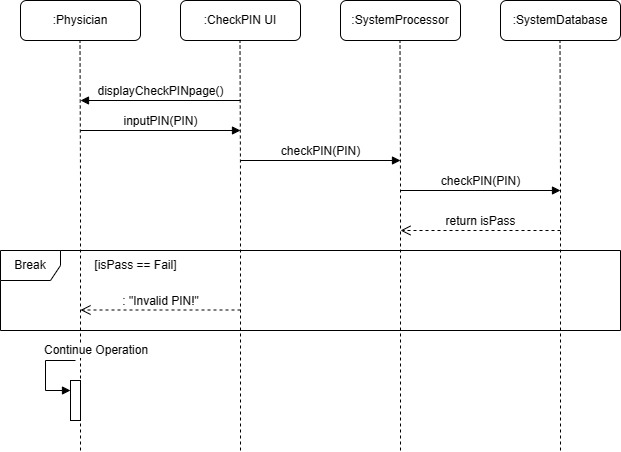


1. Click the “On / Off” button.
2. Call out a CheckPIN Dialog to check the identification.
   1. If pass the check, the processor would set the baseline status and update it in the database.
      1. If the current status is “on”, the new status will be set as “off”.
      2. If the current status is “off”, the new status will be set as “on”.
   2. If fail the check, it would display a warning.

S2: CheckPIN UI Implementation

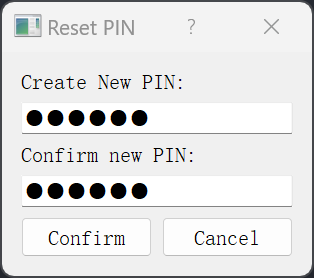


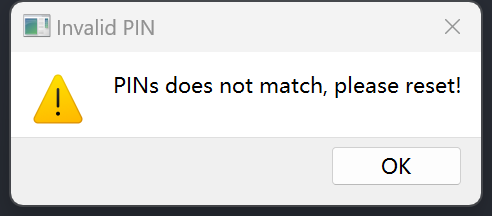
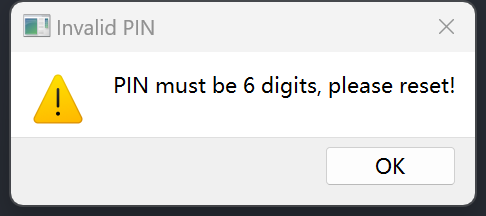


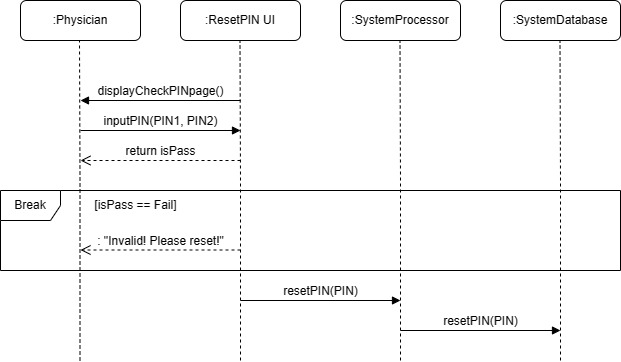


1. CheckPIN UI would let physician to type in PIN to the processor to check identification.
   1. If the physician confirms, the processor would call the database to check whether the PIN passes.
      1. If pass the check, the operation would continue.
      2. If fail the check for less than 3 times, there would be a warning and the physician could type in PIN again to pass the check.
      3. If fail the check for 3 times, the permission would be cancelled.
   2. If the physician cancels, the Dialog would close and nothing would happen.

S3: ResetPIN UI Implementation

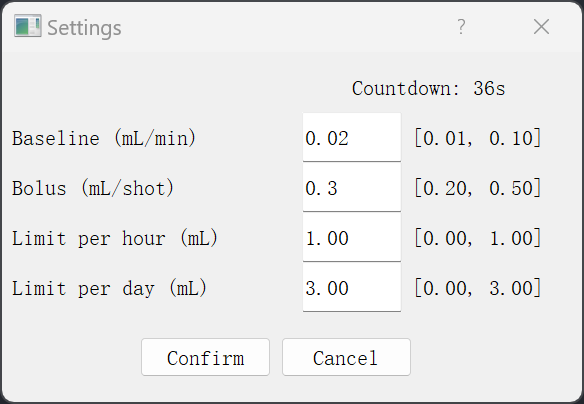


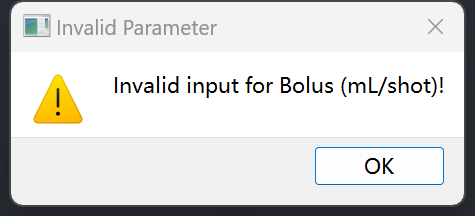
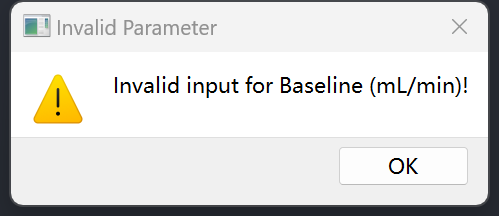


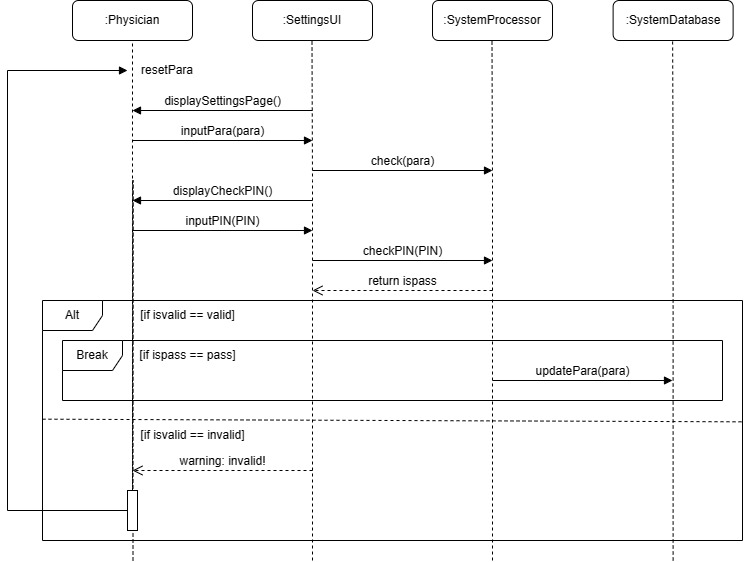


1. ResetPIN UI would let physician to type in new PIN.
2. If the physician confirms, the processor would check whether the PIN valid.
   * + 1. If the new PIN is valid and the two inputs are the same, the processor would call the database to reset the PIN.
       2. Otherwise, there would be a warning and the physician should reset the PIN.
3. If the physician cancels, the Dialog would close and nothing would happen. The PIN would not change.

S4: SettingsUI Implementation

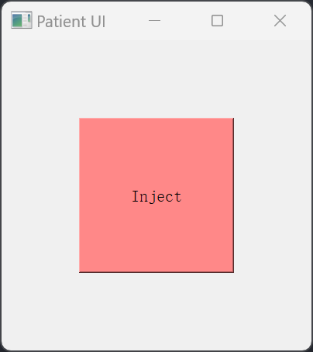


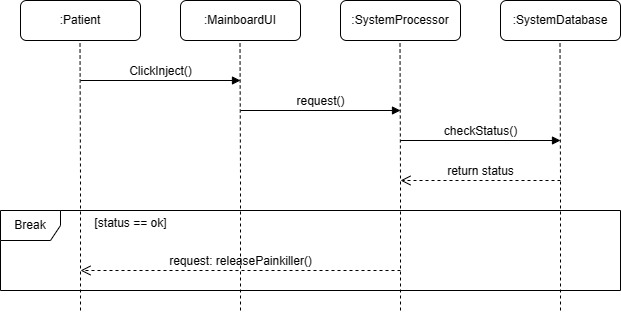




1. Settings UI would show the current parameters and their valid interval at first when it is called.
2. Settings UI allows physician to set new parameters.
3. If the physician confirms, the processor would call out a checkPIN Dialog to check the identification.
   * + 1. If pass the check, the processor would check whether the PIN valid.
     1. If all the new parameters are valid, the processor would call the database to reset the parameters.
     2. If any of the parameters is invalid, the Settings Dialog would hold on and there would be a warning to inform physician the invalid parameter in his / her new settings.
        1. If fail the check, there would be a warning and the processor would not update the parameters.
4. If the physician cancels, the Dialog would close and nothing would happen. All parameters would not change.
5. Settings Dialog has a countdown to control the Dialog. It will start counting down from 60s.
6. If the countdown is positive, the setting operation and the countdown continue.
7. If the countdown reaches zero, close the Dialog automatically.
8. If the physician confirms invalid parameters, the countdown would be reset to 60s and the operation continues.

S5: PatientUI Implementation





1. Click the “Inject” button.
2. The processor would ask the database for injection status.
   1. If the status is ok, the processor would release painkiller.