Group 18

Author: 王斌

Software Specifications

Elevator

Table of Contents

[System Architecture 2](#_Toc44020256)

[Software Specifications 2](#_Toc44020257)

[S1: ElevatorUI implementation 2](#_Toc44020258)

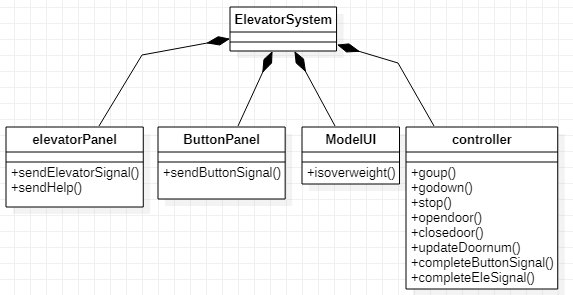
[S2: buttonUI implementation 5](#_Toc44020259)

[S3: modelUI implementation 7](#_Toc44020260)

[S4: model checking 8](#_Toc44020261)

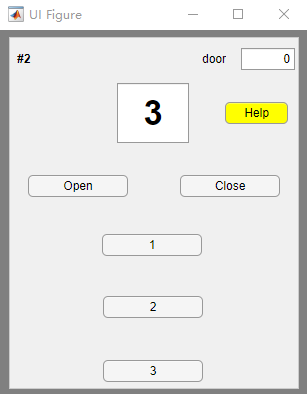
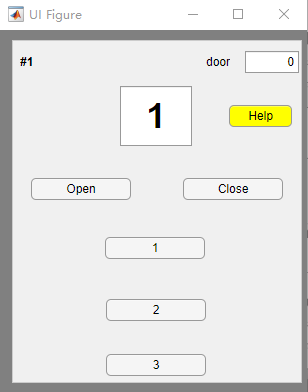
## System Architecture

The system architecture is shown below:

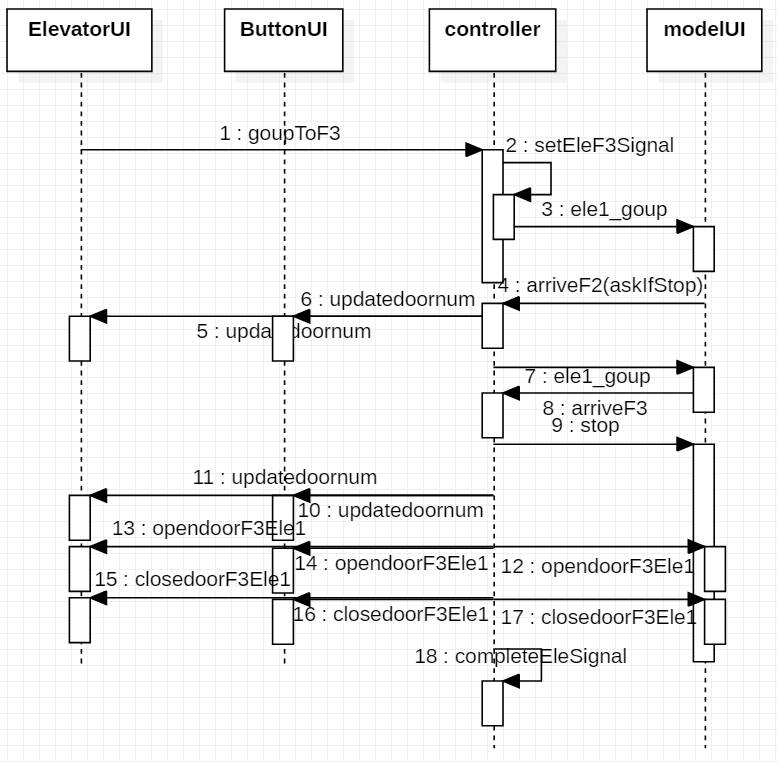


## Software Specifications

### S1: ElevatorUI implementation



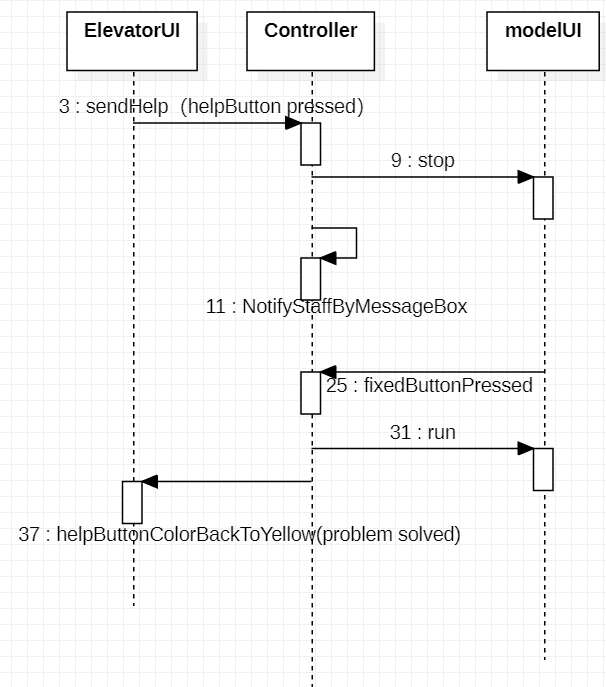
#### S1.1: Elevator call go up



* S1.1.1: elevator go up to Floor3

1. Floor3 button in elevator1 is pressed
2. Send go up to F3 elevator signal to controller
3. Controller set one F3 elevator signal to 1
4. Call modelUI elevator1 to go up
5. Elevator1 arrive at F2 ask controller if stop
6. Controller updatedoornum(the number that represents elevator are at which floor) to buttonUI and ElevatorUI
7. Control check there is no F2Ele Signal, tell model go up, do not stop
8. Elevator2 arrive at F3 tell controller
9. Controller send stop to modelUI
10. Controller check if there is F3Ele Signal
11. Controller updatedoornum to buttonUI and ElevatorUI
12. Controller send opendoor to all UI
13. Controller send closedoor to all UI
14. Controller complete F3Ele Signal, set it to 0

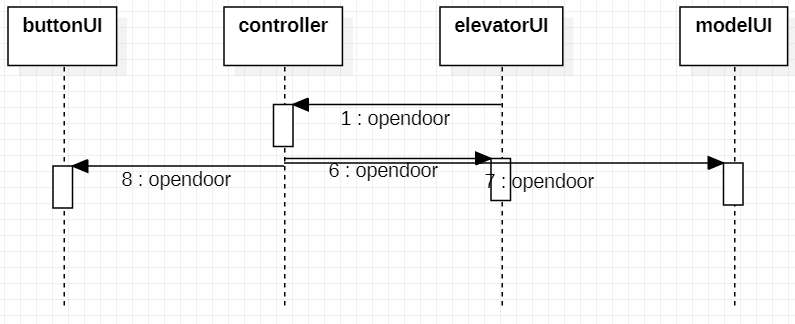
#### S1.2: Help



* S1.2.1: Elevator call help

1. Elevator send Help to controller
2. Controller send stop to ModelUI
3. Controller notify staff by coming out one message box
4. ProbelmFixed Button is pressed in ModelUI, sending signal to controller
5. Controller tell modelUI run again
6. Controller tell elevatorUI problem solved by changing the help Button color back to yellow

#### S1.3: Open door

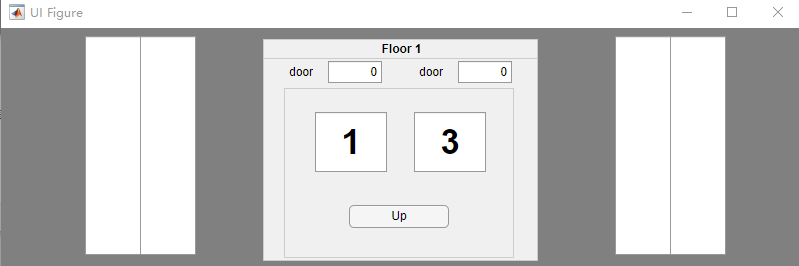


* S1.3.1: Elevator call open door from inside which is elevatorUI

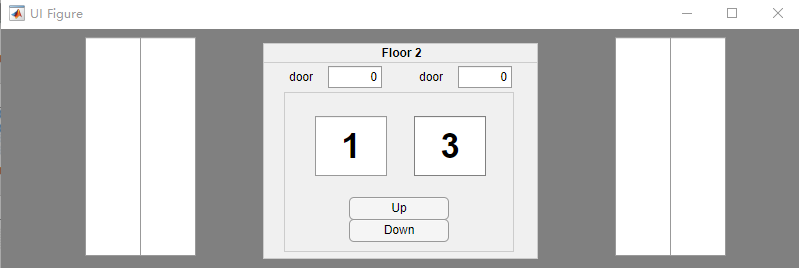
1. elevatorUI call opendoor to controller
2. controller call opendoor to all UI

### S2: buttonUI implementation

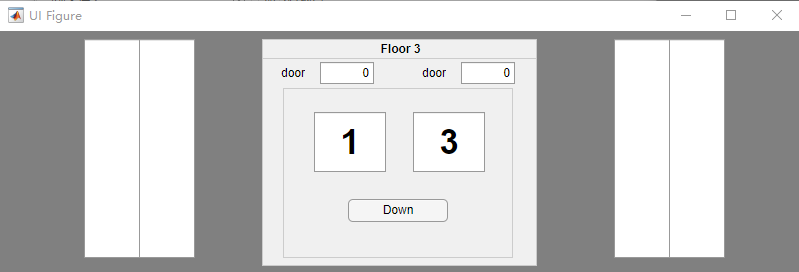
The buttonPanel has 3 UI(floor1,2,3)



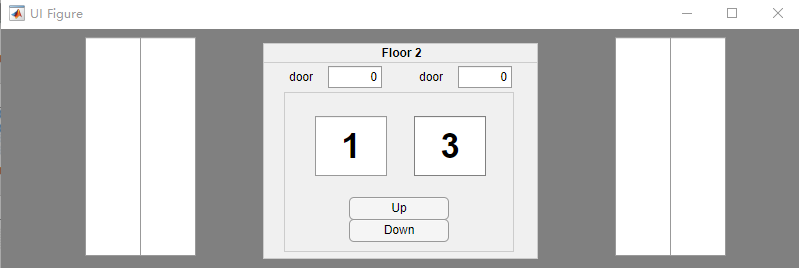
Floor1



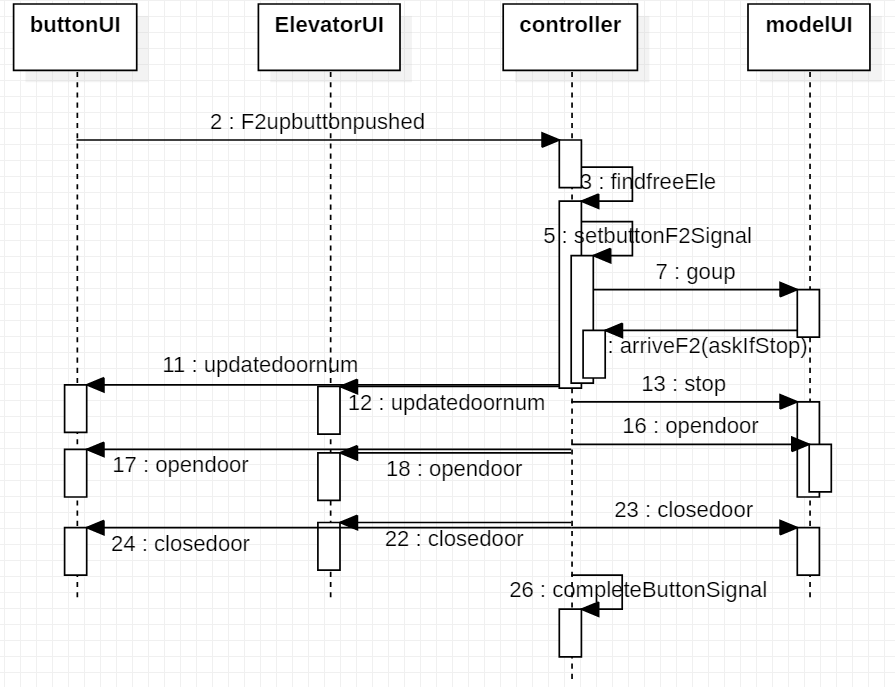
Floor2



Floor3



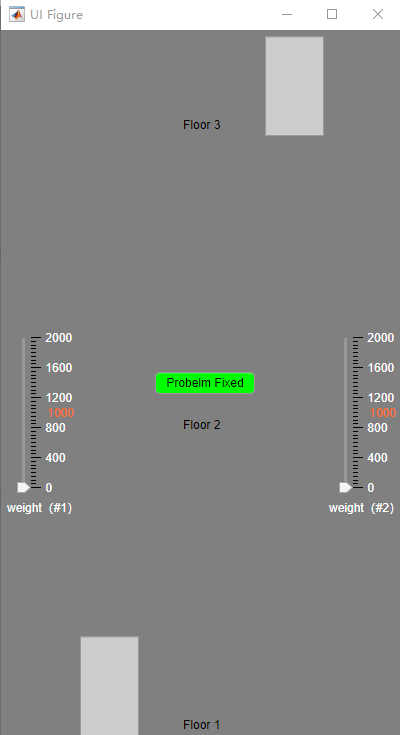
#### S2.1: button call go up



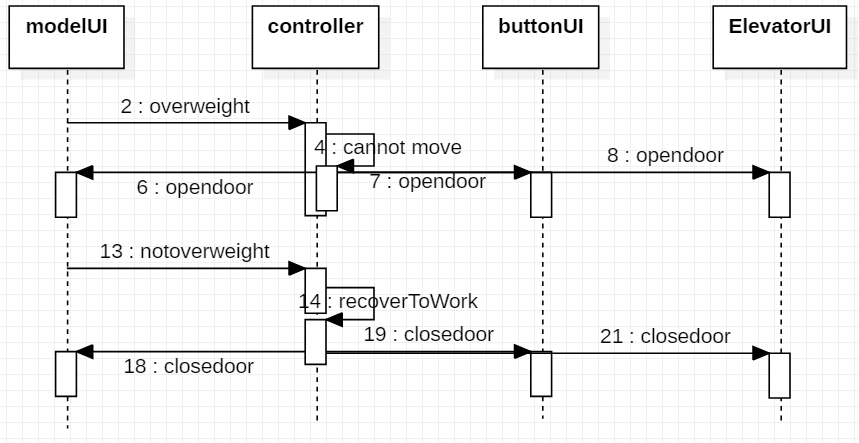
* S2.1.1: button go up to F2 from F1(suppose both elevator are free)

1. F2 up button is pushed
2. Controller find which elevator is free, suppose two elevators are free
3. Controller set the button F2 signal to 1
4. Controller send go up to modelUI
5. Elevator1 in modelUI arrives at F2 ask if stop
6. Controller send stop to modelUI
7. Controller updatedoornum to buttonUI and elevatorUI
8. Controller send opendoor to all UI
9. Controller send closedoor to all UI
10. Controller complete F2button Signal, set it to 0

### S3: modelUI implementation



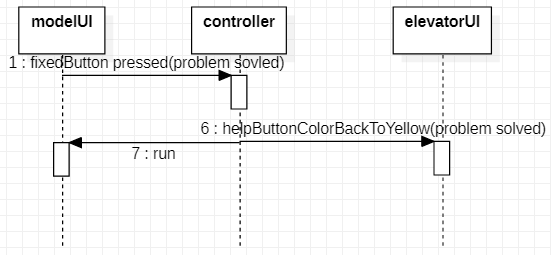
#### S3.1: modelUI call overweight



* S3.1.1: modelUI call overweight(only when elevator is at one certain floor can adjust weight)

1. modelUI send overweight to controller
2. controller cannot move if accept signal
3. controller send opendoor to all UI
4. modelUI send not overweight to controller
5. controller recover and accept signal
6. controller send closedoor to all UI

#### S3.2: modelUI fixed problem called from elevatorUI



* S3.2.1: modelUI fixed problem

1. modelUI Problemfixed Button is pressed and sending signal to controller
2. controller send run signal to modelUI and tell elevatorUI go back to working by setting its help button color back to yellow