Group 14

Author: Chen Fang

Software Requirements

Proj.1 Elevator

Table of Contents

[System Objective 3](#_Toc10411767)

[Domain Analysis 3](#_Toc10411768)

[System Architecture 4](#_Toc10411769)

[Use Cases 5](#_Toc10411770)

[Software Requirements 6](#_Toc10411771)

[R1: FloorUI 6](#_Toc10411772)

[R2: ElevatorUI 6](#_Toc10411773)

[R3: DB 6](#_Toc10411774)

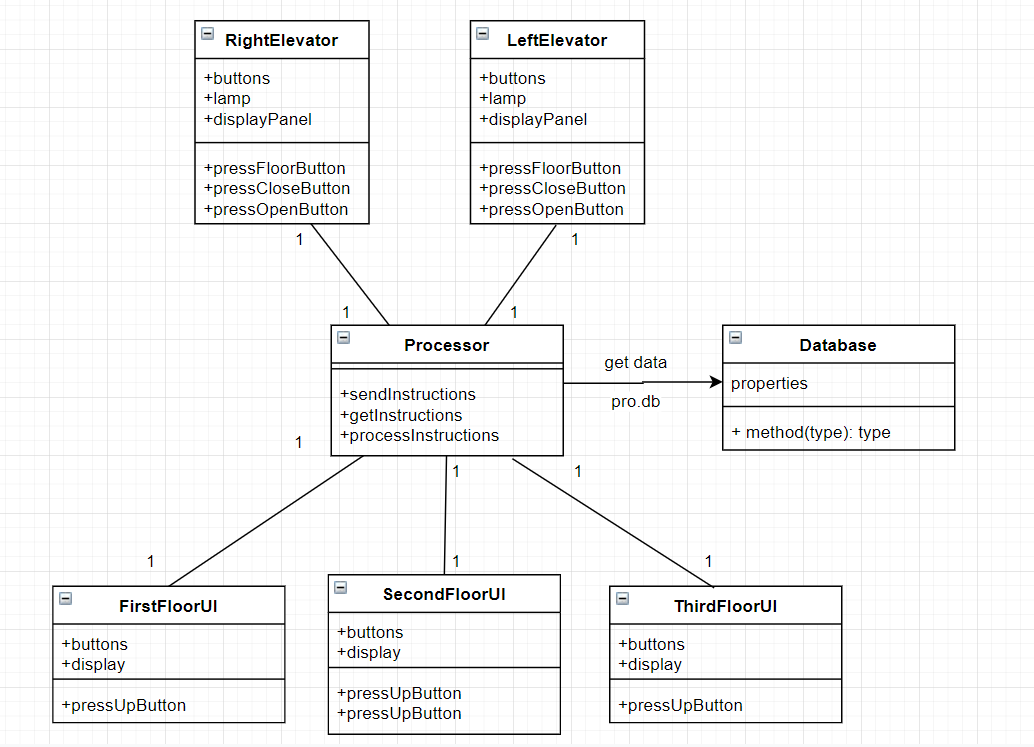
[R4: Processor 6](#_Toc10411775)

## System Objective

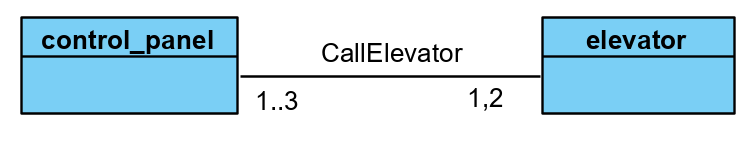
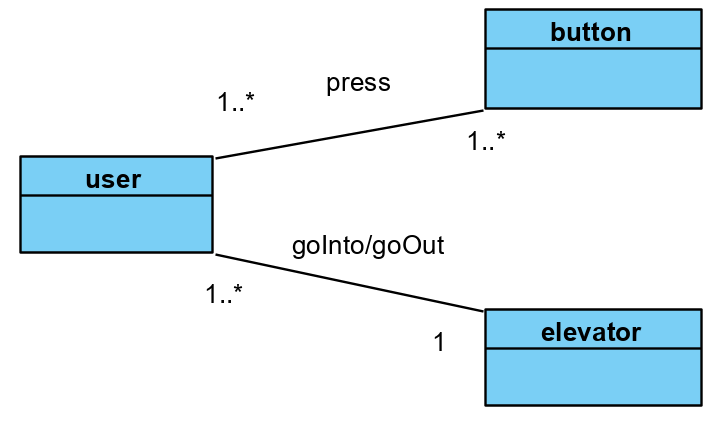
In this project, we are developing a system & design of elevator that can provide users high efficiency and safety when using elevator. By providing interconnected interfaces to users and elevator system, the system can reduce waiting time and probability of failure, which can improve customer satisfaction and make using elevator easy.

## Domain Analysis

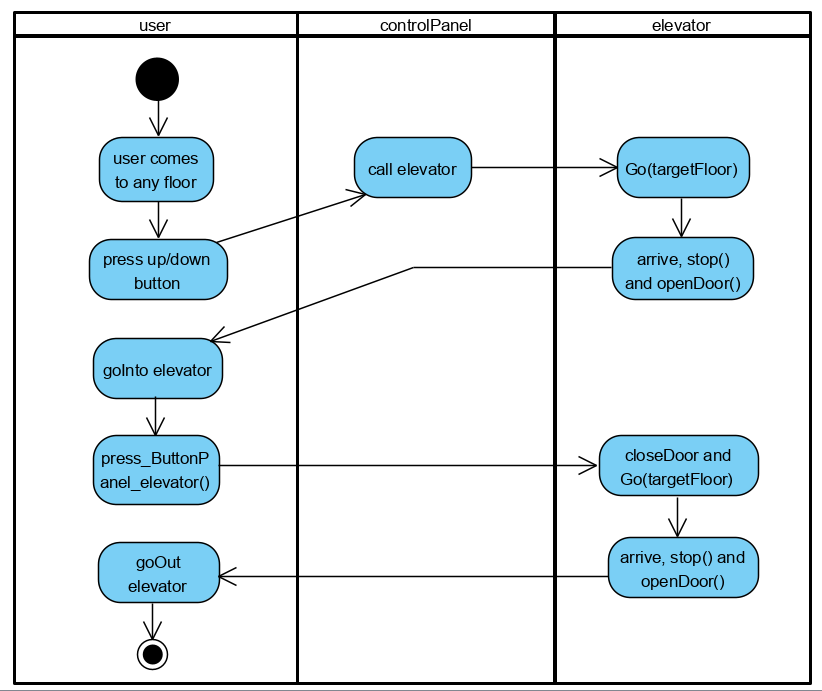
The participants of activities in using elevator are users, 2 elevators (each with UI in it) and control panels on each 3 floor.



The relationships among different participants are shown as follows:

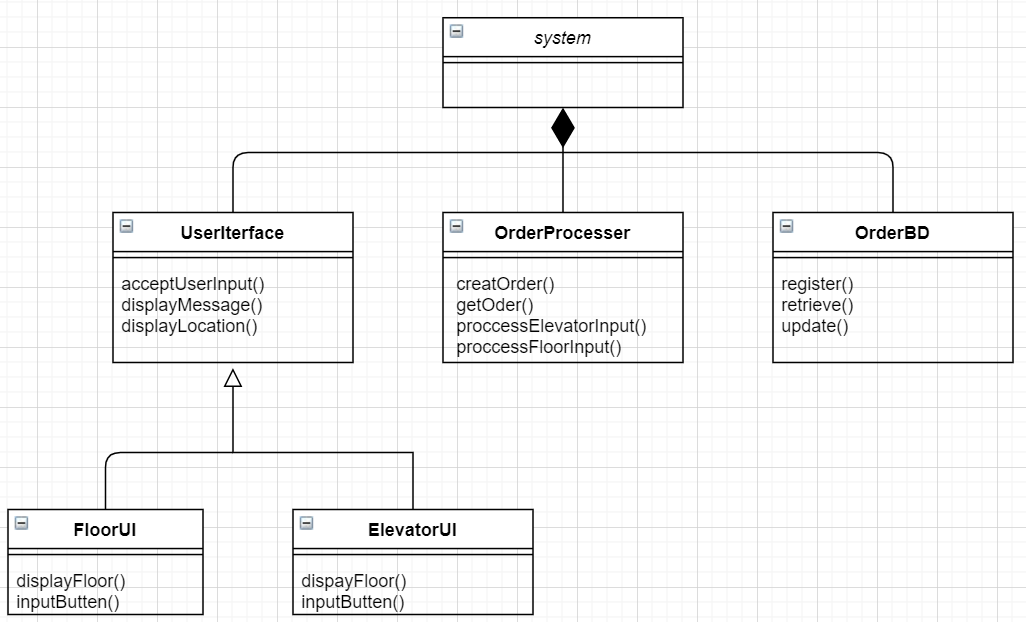


Here is the sequence of events for user use elevator:



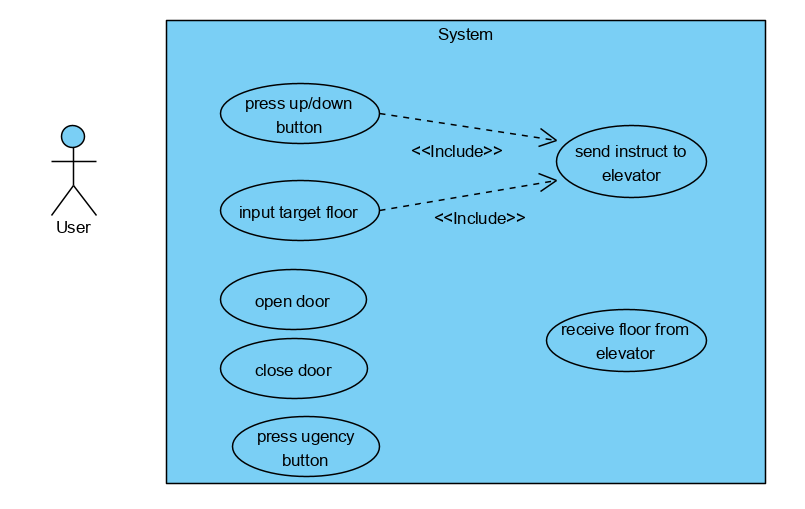
## System Architecture

From the information above, we will design a software system that connect the user, control panel on each floor and elevators. The system architecture is shown below:



## Use Cases

The system can achieve the following use cases from user’s perspectives:



## Software Requirements

### R1:FloorUI

* R1.1: The user on any floor should get the information of elevator.
  + R1.1.1: Which floor the elevator is.
  + R1.1.2: If the elevator is going up or going down or stop.
  + R1.1.3: The elevators’ floor info should be displayed on each floor.
* R1.2: The user can press up/down button and can also repeat pressing them.

### R2: ElevatorUI

* R2.1: User should get information on screen.
  + R2.1.1: The passengers should be able to press all the buttons and access the floors.
  + R2.1.2: Whether the elevator is going up or going down or stop.
  + R2.1.3: If there is fault with elevator.
* R2.2: User can input command to control elevator.
  + R2.2.1: Input target floor.
  + R2.2.2: Input open/close door while elevator is stop at any floor.
  + R2.2.3: Input urgency button when something happen.

### R3: DB

* R3.1: DB should have solutions for all case the and operations.
  + R3.1.1: Different elevators positions and their different action
  + R3.1.2: Command from each elevator and each floor.

### R4: Processor

* R4.1: Processor should get the input from user:
  + R4.1.1: Able to process legal command.
  + R4.1.2: Ignore illegal command (e.g. 2🡪3,press 1, ignore 1)
* R4.2: Processor should give correct and efficient command to 2 elevators:
  + R4.2.1: Each command should be valid.
  + R4.2.2: Command should make user wait less.
  + R4.2.3:The nearest free elevator should come to save time.