Group 7

Author: Xinyue Hu

Software Requirements

Elevator

Table of Contents

[System Objective 3](#_Toc1835931763)

[Domain Analysis 3](#_Toc1460100645)

[System Architecture 5](#_Toc615906246)

[Use Cases 5](#_Toc665097982)

[R1: SystemUI 7](#_Toc649400839)

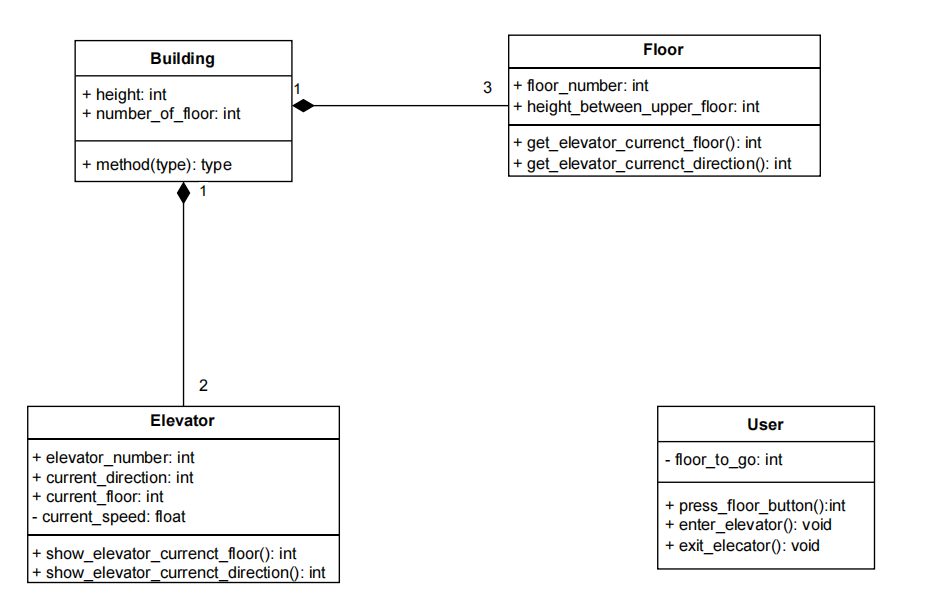
[R2: ElevatorProcessor 7](#_Toc968007019)

[R3: SystemProcessor 8](#_Toc2105342308)

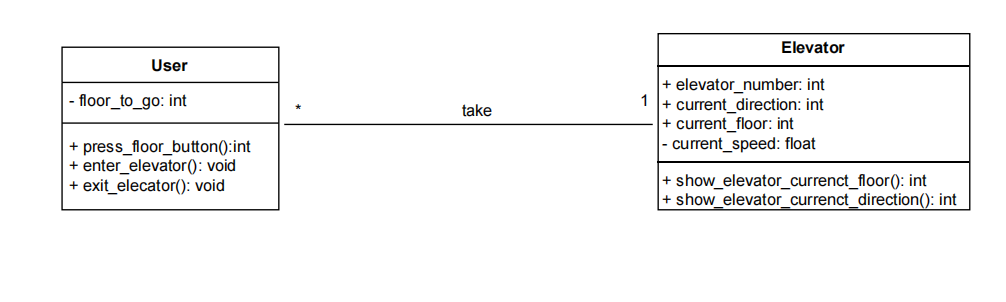
## System Objective

In this project, we are developing a software that can contral and schedule the movement of two cooperate elevators in a three-floor building according to people’s need. By providing interconnected interfaces to passengers outside and inside the elevators, the system can provide security guarantees and work efficiently, which can improve customer satisfaction.

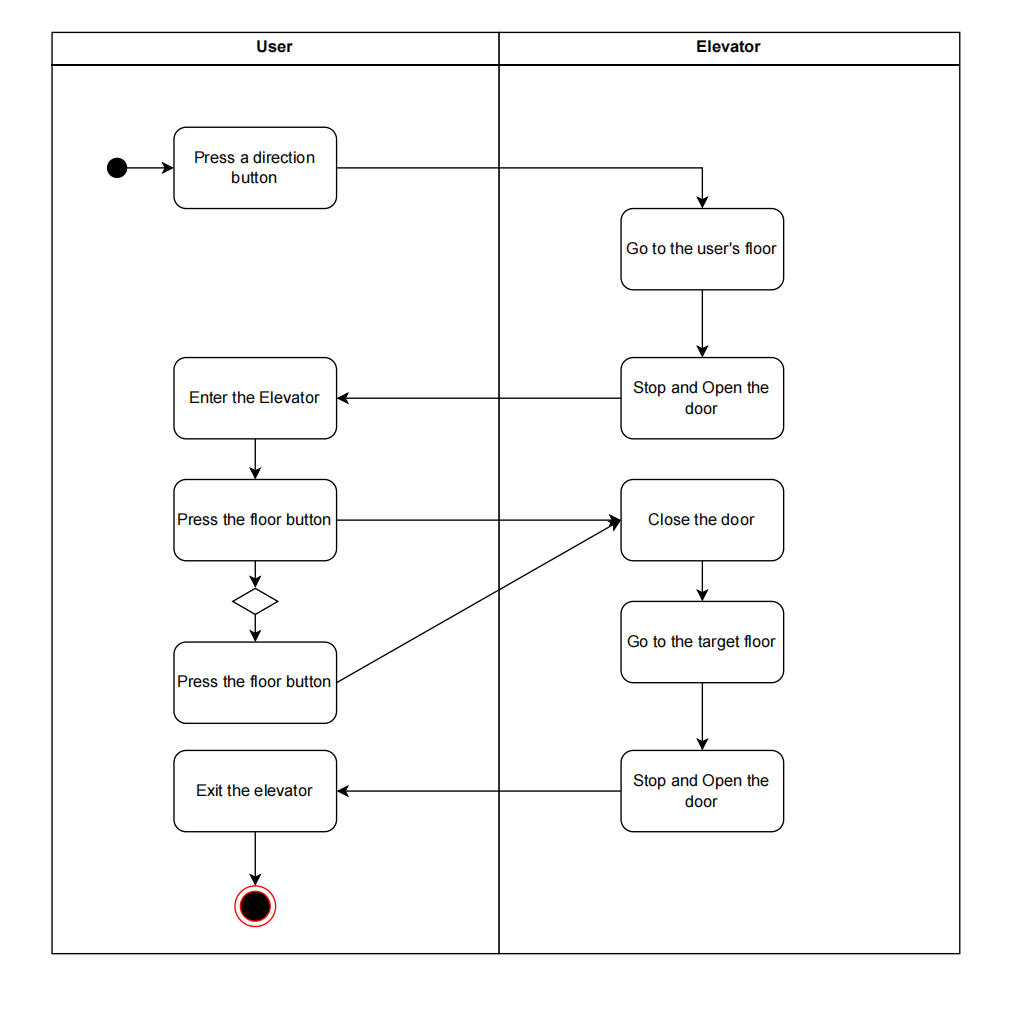
## Domain Analysis

The participants of activities of the whole system can be categorized into Elevator and User.

The relationships among different participants are shown as follows:



Here is the sequence of events for a user to take an elevator:



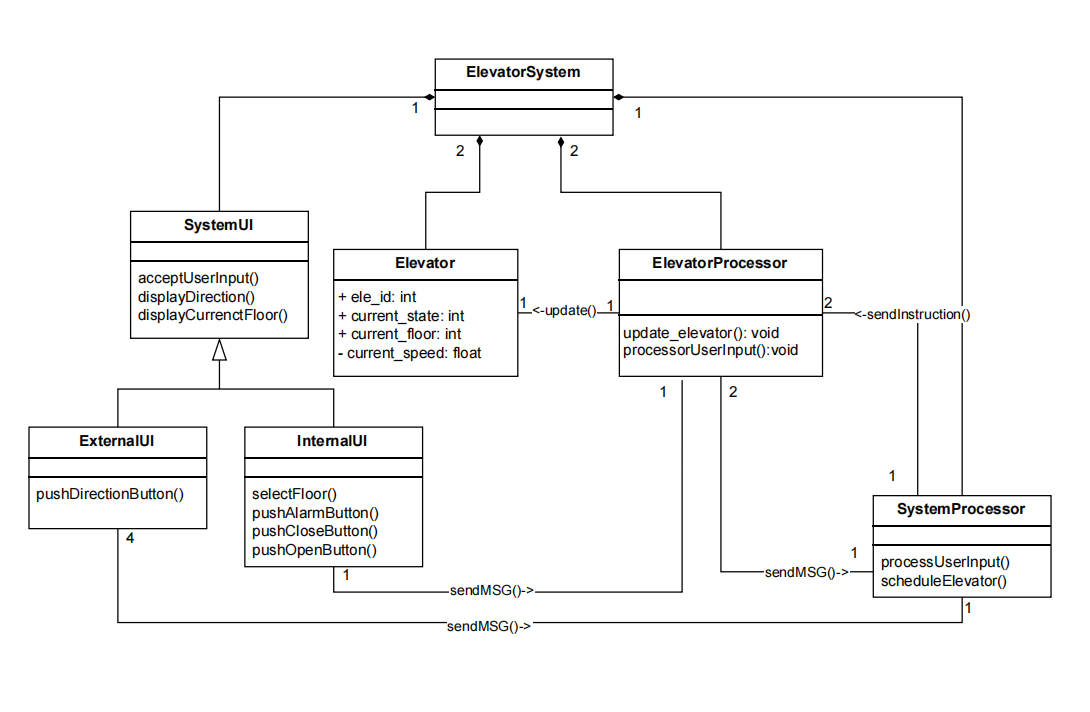
## System Architecture

From the information above, we will design a software system that allows the user to choose his/her target floor using the internal and external interface of the elevators. The elevator can bring the user to the target floor and provide the direction and currenct floor all the time.

We design a ElevatorProcessor to deal with the signal sent by the InternalUI and to update the state of the elevator. The ElevatorProcessor can also send message to the System Processor and receive instructions from the System Processor

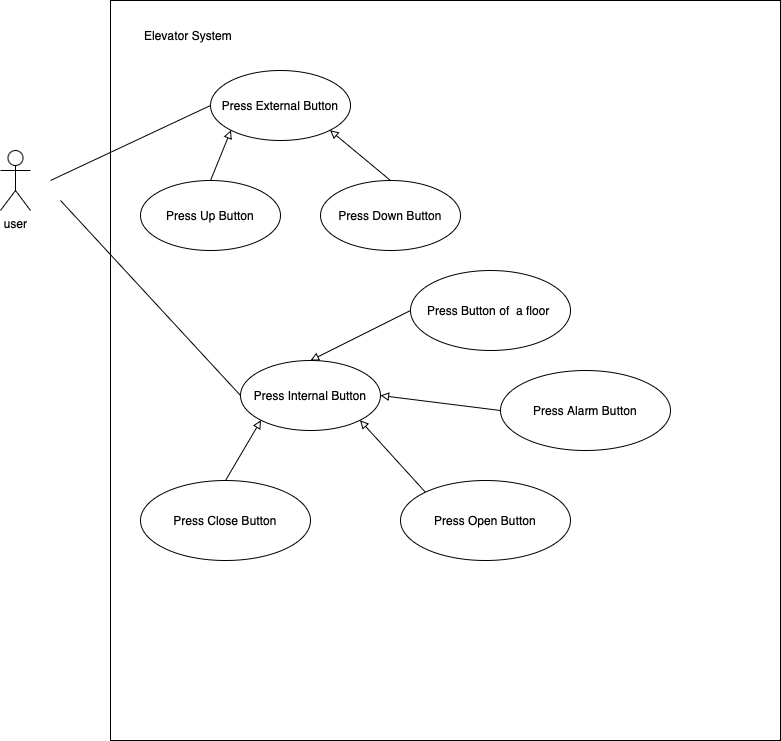
The SystemProcessor is designed to deal with the signal send by the ExternalUI and the preprocessed signal sent by the ElevatorProcessor and mostly importantly, schedules the elevator and controls elevator operation by sending messages to the elevator processor.

The system architecture is shown below:



## Use Cases

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



Software Requirements (of elevators)

### R1: SystemUI

* R1.1: InternalUI
  + R1.1.1: A user should take no effort to press the button on the InternalUI
  + R1.1.2: A user should be able to know the current floor he/she is on and the moving direction of the elevator at any time inside the elevator
  + R1.1.3: A user should be able to select the floor he/she want to go to
  + R1.1.4: A user should be able to open/close the door
* R1.2: ExternalUI
  + R1.2.1: A user should take no effort to press the button on the ExternalUI
  + R1.2.2: A user should be able to know the current floor and the moving direction of the two elevators
  + R1.2.3: A user should be able to select the direction(up or down)
* R1.3: The information about the elevator(its currenct floor and movement of the elevator) given by the externalUI and the internalUI should be the same
* R1.4: The systemUI should be able to send information to the processor

### R2: ElevatorProcessor

* R2.1: The ElevatorProcessor should let the elevator work normally
  + R2.1.1: The ElevatorProcessor should let the elevator bring the users to their taget floor
  + R2.1.2: The ElevatorProcessor should be control the door of the elevator
    - R2.1.2.1: The door should be opened when the elevator is arrived at a target floor
    - R2.1.2.2: The door should be closed/opened when the open/close button is on and the elevator is stopped at a floor.
* R2.2: The ElevatorProcssor should guarteen the safety of its users.
  + R2.2.1:The door should keep closed when the elevator is moving
  + R2.2.2: The door should keep open when detecting someone is passing through it, the least time for an open should be moderate
  + R2.2.3: The speed of the elevator should no more than 1.5m/s
* R2.3: The ElevatorProcessor should schedule the elevator to work efficiently.
  + R2.3.1: The elevator should stop when no button was on
  + R2.3.2: When the elevator is down, the ElevatorProcessor should let the elevator handle all reachable down request
  + R2.3.3:When the elevator is up, the ElevatorProcessor should let the elevator handle all reachable up request
  + R2.3.4: The expected speed of the elevator should always faster than a normal speed of a person using stairs

### R3: SystemProcessor

* R3.1: The SystemProcessor should work normally
  + R2.1.1: The SystemProcessorr should be able to receive information from the External UI
  + R2.1.2: The elevator should be able to add up/down request to the Elevator Processor
* R3.2:The SystemProcessor should schedule the two elecators to work efficiently.
  + R3.2.1: When there is a/an up/down request, the SystemProcessor should allocate it to “nearest” elevator (the elevator that may be able to handle the requst the fatest)
  + R3.2.2: All the request added by the SystemProcessor to the ElevatorProcessor is necessary