SOFTWARE REQUIREMENT

Elevator System

Team 1
Author: Tiansu Chen

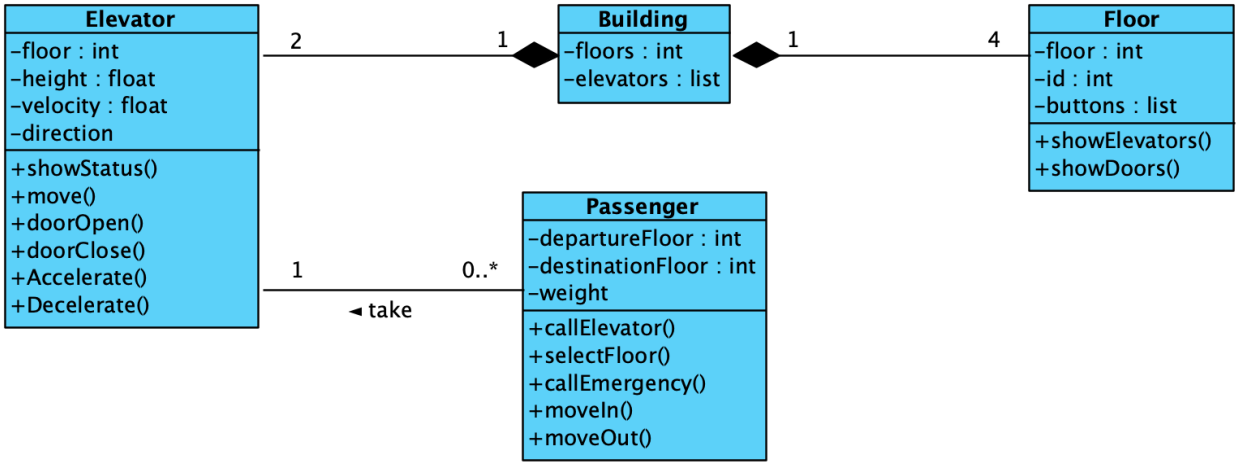
Table of Contents

System Objective

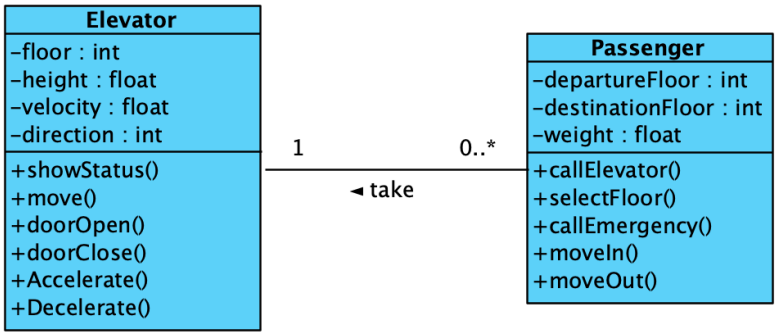
In this project, an elevator system software that fits to a 3-floor building with a basement is developed, with an automatic scheduling algorithm ensuring two elevators moving collaboratively. Through this system, users can call elevators from each floor and go to the destination safely and efficiently, getting a good riding experience.

Domain Analysis

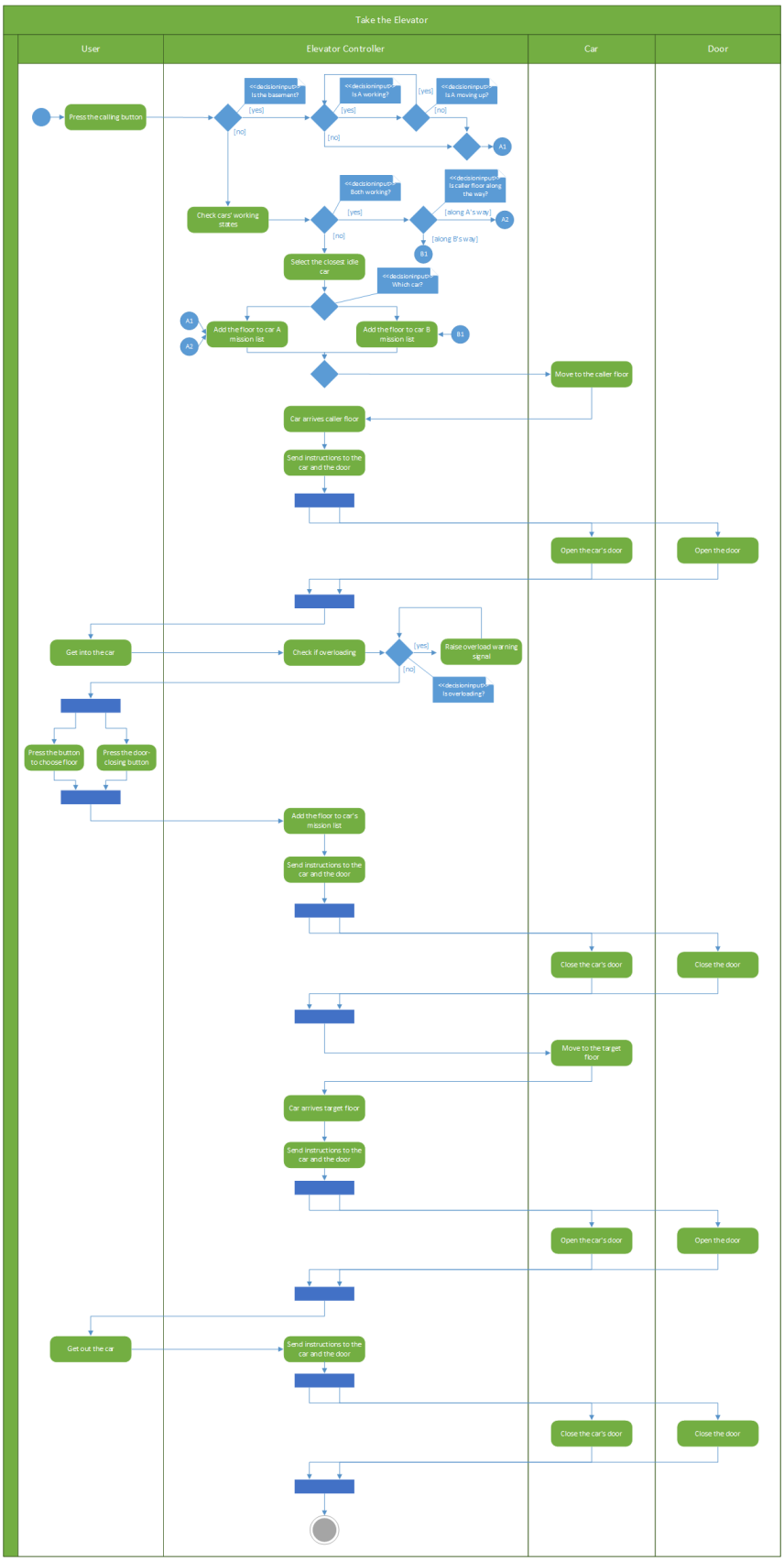
The subjects relating to the elevator system can be categorized into Elevator and Passenger.



The relationship among different participants are shown as follows.

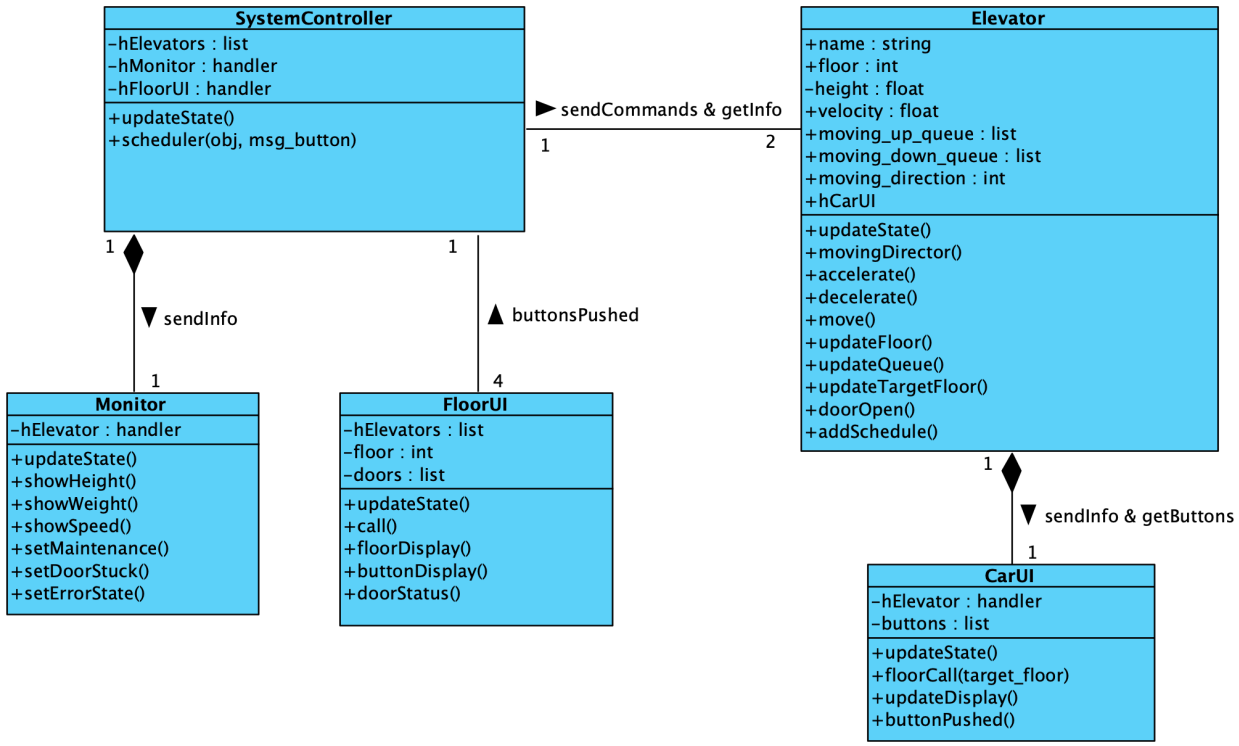


The activity diagram below shows a sequence of events takes place when a passenger is going to take an elevator to another floor.


Take the Elevator
User
Elevator Controller
Car
Door
Press the calling button
<<decisioninput>> Is the basement?
<<decisioninput>> Is A working?
<<decisioninput>> Is A moving up?
[yes]
[no]
A1
Check cars' working states
<<decisioninput>> Both working?
<<decisioninput>> Is caller floor along the way?
[along A's way]
[along B's way]
A2
B1
Select the closest idle car
<<decisioninput>> Which car?
Add the floor to car A mission list
Add the floor to car B mission list
Move to the caller floor
Car arrives caller floor
Send instructions to the car and the door
Open the car's door
Open the door
Get into the car
Check if overloading
Raise overload warning signal
<<decisioninput>> Is overloading?
Press the button to choose floor
Press the door-closing button
Add the floor to car's mission list
Close the car's door
Close the door
Move to the target floor
Car arrives target floor
Get out the car

System Architecture

The elevator system is controlled by a *SystemController*, receiving button information from each *FloorUI* and *Elevator*, and sending information and commands to the *Monitor* and *Elevator*. Each Elevator has a CarUI, with which users can observe the elevator status and select floors to go. The architecture of the elevator system is shown below.

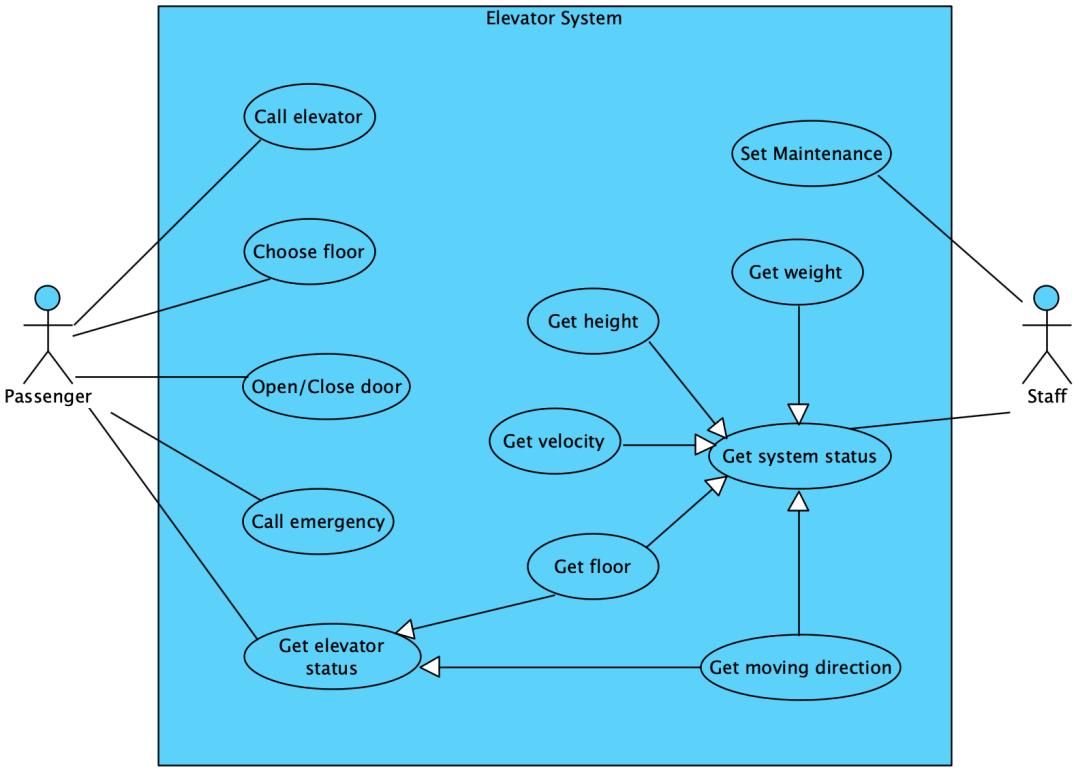


Use Cases

In the waiting area on each floor, passengers can call an elevator to move upward or downward. In the meanwhile, the user can get the floor the elevators are and their moving directions.

In the elevator, passengers can press buttons to select floors to go or make an emergency call, see the floor they are in, and get the moving direction of the elevator.

In the control room, the maintenance staff can observe all the information of two elevators, including height, velocity, weight, etc.



Software Requirements

R1: FloorUI

- R1.1: Passengers should be able to call elevators to go upward/downward by pressing up/down buttons on the FloorUI.
- R1.2: Passengers should be able to know whether their call to elevators are in progress or has be finished through the color of light on the buttons.
- R1.3: Passengers should be able to know the status of two doors (close/closing/open/opening) in their floor through the panel of FloorUI.
- R1.4: Passengers should be able to know the status of two elevators through the display zone on the FloorUI:
  - R1.4.1: Passengers should be able to know which floor each elevator is at.
  - R1.4.2: Passengers should be able to know the moving direction of each elevator.
- R1.5: Passengers should know which floor they are on through the floor label on the FloorUI.

R2: CarUI

- R2.1: Passengers should be able to know the moving status of the elevator through the CarUI display zone:
  - R2.1.1: Passengers should be able to know which floor the elevator is at.
  - R2.1.2: Passengers should be able to know the moving direction of the elevator.
- R2.2: Passengers should be able to know the safety status of the elevator through the signal lamps and the door simulation:
  - R2.2.1: Passengers should know whether the elevator is overweight by the Overweight lamp.

    - R2.2.2: Passengers should know whether the elevator is in an emergency by the Emergency lamp.
    - R2.2.3: Passengers should know whether the door is open/opening by the green/red color of the Door Open lamp and simulating animation of the door.
    - R2.2.4: Passengers should know whether the door is close/closing by the green/red color of the Door Close lamp and simulating animation of the door.
- R2.3: Passengers should be able to select floors to go by pressing the floor buttons.
- R2.4: Passengers should be able to open/close the door by pressing the door open/close button, only when the elevator is stopping at a certain floor.
- R2.5: Passengers should be able to report emergency by pressing the EMERGENCY button.

R3: Monitor

- R3.1: Staffs should be able to monitor the height of two elevators in real time by the sliding bars on the left panel.
- R3.2: Staffs should be able to select which elevator's information is displayed by the drop-down menu.
- R3.3: Staffs should be able to monitor the details of the elevator by the displayed information on the right panel:
    - R3.3.1: Staffs should be able to monitor the weight of the elevator.
    - R3.3.2: Staffs should be able to monitor the speed of the elevator.
    - R3.3.3: Staffs should be able to monitor the height of the elevator.
    - R3.3.4: Staffs should be able to monitor the safety status of the elevator.
- R3.4: Staffs should be able to set/cancel the maintenance status of the elevator by pressing the MAINTAIN/RECOVER button on the right panel.

R4: Controller

- R4.1: The system controller should ensure no request is unmet.
    - R4.1.1: All the calls from the FloorUI should be finished.
    - R4.1.2: All the floors selected in the elevator's CarUI should be arrived at.
    - R4.1.3: All the requests related to the basement should only be executed #2 elevator.
- R4.2: The system controller should be able to schedule the two elevators efficiently.
    - R4.2.1: The disengaged elevator should be called immediately.
    - R4.2.2: If the two elevators are both occupied when a new call from FloorUI appears, the one with less planed-distance should accept the mission.
    - R4.2.3: The elevator should keep its previous moving direction until there is no call in that direction.
    - R4.2.4: The elevator should be able to handle a sequences of calls in one pass.
- R4.3: The system controller should be able to schedule the two elevators safely.
    - R4.3.1: The controller should be able to control the speed of each elevator to let them stop at the right height and not go over than the velocity limit.
    - R4.3.2: The doors inside and outside the elevator should never open if the elevator is not stopped at the right height.
    - R4.3.3: The elevator should never move if the doors are not closed.