Use Case 1: Simulate Elevator System
  Name: Simulate Elevator System
  Primary Actor(s): Administrator
  Stakeholders:
  Administrator- Configures and monitors the simulation.
  Passenger- Interact with the elevator system during the simulation.
  Building Safety Service- Responds to help requests and safety conditions.
  Pre-condition(s):
    ● The simulation software is installed and running.
    ● The administrator has access to the configuration interface.
  Success guarantee(s) (Post-conditions):
    ● The simulation completes all configured events.
    ● All elevators are idle at the end of the simulation.
    ● All safety conditions are handled appropriately.
    ● A log of all actions and system responses is generated.
  Main success scenario:
    1. Selects the number of floors (N) and elevators (M).
    2. Specifies the number of passengers and their behaviors (e.g., floor requests, car requests, help button presses, open/close door actions).
    3. Configures safety events (e.g., fire, overload, power outage) and the simulation time steps at which they occur.
    4. The simulation begins at time step 0.
    5. The log console displays the initial configuration and simulation start message.
    6. Simulation now runs, At each time step:
       6.1 The current time step is displayed.
       6.2 Passenger actions (e.g., pressing buttons, entering/exiting elevators) are processed.
       6.3 Elevator states (e.g., moving, idle) and positions are updated.
       6.4 Safety conditions are monitored and handled if triggered.
       6.5 Logs are updated with all actions and system responses.
    7. Administrator monitors the simulation:
       7.1 Views the log console for real-time updates.
       7.2 Pauses, continues, or stops the simulation as needed.
    8. Simulation completes:
       8.1 All events are processed.
       8.2 All elevators are idle.
       8.3 A final log is generated summarizing the simulation.
  Extensions:
  1a. Invalid Configuration
       1a1. The administrator inputs invalid values (e.g., negative floors, zero elevators). The system displays an error and prompts for correction before starting the simulation.

6a. Simulation Paused or Stopped

      6a1. The administrator pauses or stops the simulation. The system halts all actions, logs the current state, and resets elevators to idle.

6b. Safety Event Triggered

      6b1. A safety event (e.g., fire, overload) occurs during the simulation. The system handles the event (e.g., moves elevators to safe floors) and logs the actions.

6c. Logging Failure

      6c1. The system fails to update the log. The administrator is notified, and the simulation stops with a partial log generated.

6d. Elevator Malfunction

      6d1. An elevator stops functioning during the simulation. The system logs the issue, reroutes passengers, and notifies the administrator.

Use Case 2: Request Elevator

Name: Request Elevator

Primary Actor(s): Floor Button User

Stakeholders: Passengers, Building Management

Pre-condition(s):

- The elevator system is operational.
- Floor buttons are accessible.

Success guarantee(s) (Post-conditions): Elevator arrives at the requested floor, allowing passengers to board.

Main success scenario:

1. Passenger presses the "up" or "down" button on the desired floor.
2. The button illuminates, indicating the request is registered.
3. The elevator control system assigns an elevator to service the request based on its current position and state.
4. The assigned elevator arrives at the requested floor, rings the bell, and opens its doors.
5. Elevator doors remain open for 10 seconds allowing the passengers to board or exit.
6. Passengers board the elevator.
7. Elevator bell rings again and the door closes with passengers inside the elevator.

Extensions:

5a. Passenger presses "close door" button within the elevator

      5a1.Elevator closes before 10 second time limit

6a. Passengers/cargo load exceeds carrying capacity

      6a1. Overload alarm is raised (see Use Case 6)

7a. Light sensor is interrupted while door is closing

      7a1. Obstruction alarm is raised (see Use Case 4)

Use Case 3: Select Destination

Name: Select Destination

Primary Actor(s): On-board Passenger

Stakeholders: Passengers, Building Management

Pre-condition(s): Elevator doors are closed, and the passenger is inside the elevator.

Success guarantee(s) (Post-conditions): Elevator moves to the selected destination floor.

Main success scenario:
1. Passenger presses the button corresponding to the desired floor on the elevator panel.
2. The button illuminates to confirm the selection.
3. Elevator moves to the selected floor, displaying the current floor number during transit.
4. Upon arrival, the elevator rings the bell, opens the doors, and turns off the button's illumination.
5. Passenger exits the elevator.

Extensions:

3a. Fire alarm is raised from outside or within the elevator.

    3a1. Fire alarm signal is generated. (see Use Case 5)

3b. Power is cut

    3b1. The control system detects a power outage and activates battery backup. (see Use Case 7)

Use Case 4: Emergency Help

Name: Emergency Help

Primary Actor(s): On-board Passenger

Stakeholders: Passengers, Building Safety Team, Emergency Services

Pre-condition(s): Elevator doors are closed, and the passenger is inside the elevator.

Success guarantee(s) (Post-conditions): Passenger is connected to building safety or emergency services.

Main success scenario:
1. Passenger presses the "Help" button on the control panel.
2. Elevator establishes a voice connection with building safety.
3. If no response is received within 5 seconds, a 911 call is automatically initiated.
4. Audio and visual indicators confirm the emergency signal.

Extensions:

1a. No Response from Building Safety Services

    1a1. The system escalates the call to emergency services for additional support.

Use Case 5: Door Obstacles

Name: Door Obstacles

Primary Actor(s): Passengers, Object Sensor

Stakeholders: Passengers, Building Management

Pre-condition(s): Elevator doors are closing and the light sensor is operational.

Success guarantee(s) (Post-conditions): Doors reopen if obstructed.

Main success scenario:
1. Light sensor detects an obstruction while doors are closing.
2. Elevator stops the door closure and reopens them.
3. Audio and text warnings are displayed after repeated obstructions.

Extensions:

3a. Persistent Obstruction

    3a1. System alerts building management if the issue persists.

Use Case 6: Fire Alarm

    Name: Fire Alarm

    Primary Actor(s): Building Fire System, Elevator Sensors

    Stakeholders: Passengers, Building Management, Emergency Services

    Pre-condition(s): A fire alarm signal is received from the building or elevator sensors.

    Success guarantee(s) (Post-conditions): Elevators are evacuated to a safe floor.

    Main success scenario:

        1. The system receives a "Fire" alarm signal.
        2. All elevators move to a predefined safe floor.
        3. The system issues audio and text messages instructing passengers to disembark.
        4. Once all elevators are evacuated, they remain idle until reset by building management.

    Extensions:

    2a. Obstructed Floor Access i.e the safe floor is inaccessible.

        2a1. The system selects an alternate floor and informs passengers.

Use Case 7: Overload

    Name: Overload

    Primary Actor(s): Weight Sensor

    Stakeholders: Passengers, Building Management

    Pre-condition(s): Elevator sensors detect an overload condition.

    Success guarantee(s) (Post-conditions): The elevator does not move until the load is reduced.

    Main success scenario:

        1. The weight sensor triggers an "Overload" alarm signal.
        2. The elevator issues an audio and text message informing passengers to reduce the load.
        3. The elevator remains stationary until the load is within acceptable limits.

    Extensions:

    3a. Repeated Overload

        3a1. An alert is sent to building management for investigation.

Use Case 8: Power Out

    Name: Power Out

    Primary Actor(s): Building Power System

    Stakeholders: Passengers, Building Management

    Pre-condition(s): Power failure occurs, and the battery backup system is operational.

    Success guarantee(s) (Post-conditions): Passengers are safely evacuated to a predefined floor.

    Main success scenario:

        1. The control system detects a power outage and activates battery backup.
        2. Each elevator moves to a predefined safe floor.
        3. Audio and text messages inform passengers of the power outage and direct them to disembark.
        4. Elevators remain idle until power is restored.

    Extensions:

    1a. Battery Failure

        1a1. The system issues an emergency alert to building management for immediate response.

Part 2: Design documentation and traceability matrix

UML Class Diagram-

Completeness Relative to Requirements
Simulation Setup and Execution Control: Covered by ControlSystem and MainWindow
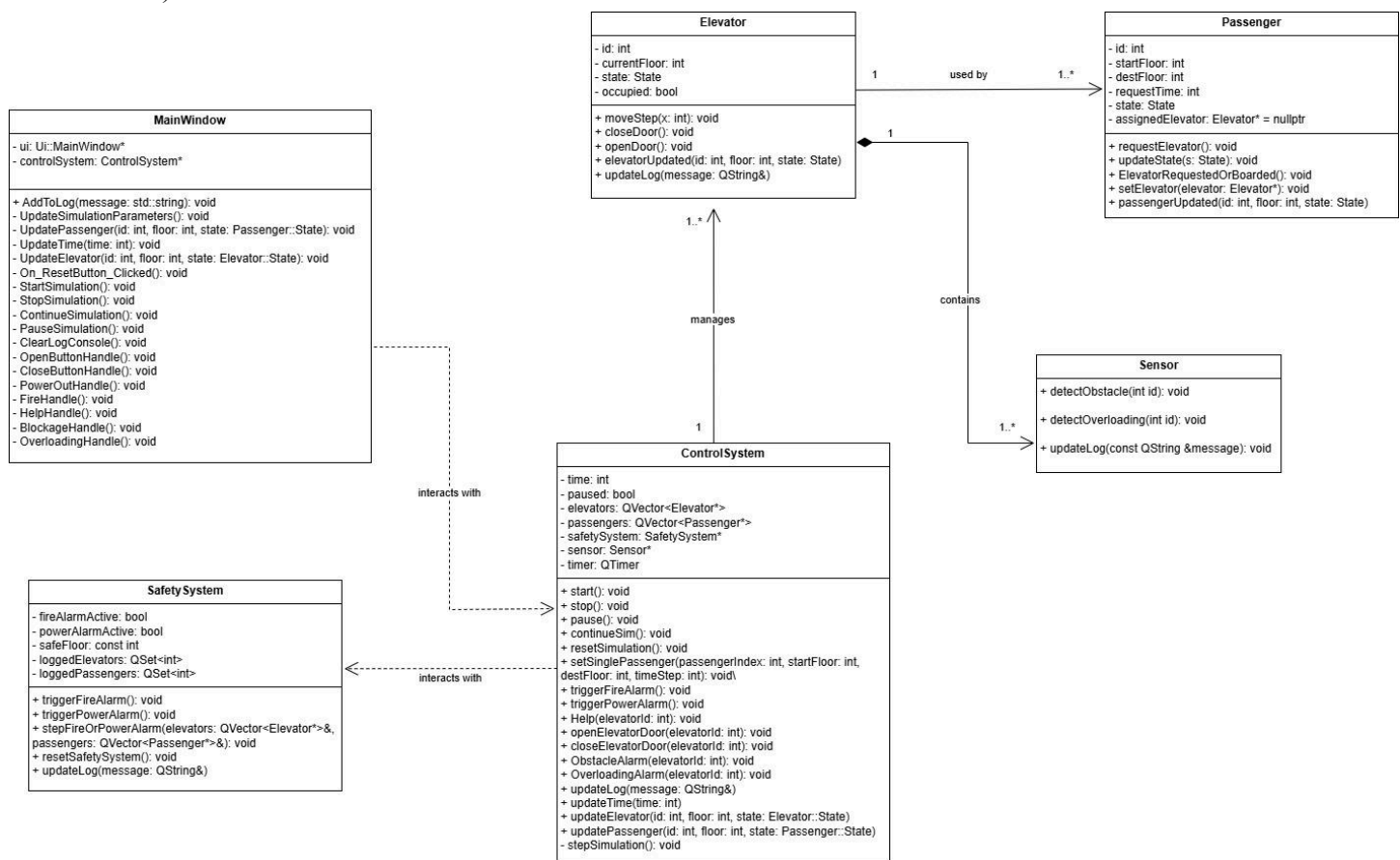Elevators: Covered by Elevator and Sensors
Sensors: Covered by Sensor.
Safety Conditions: Covered by SafetySystem.
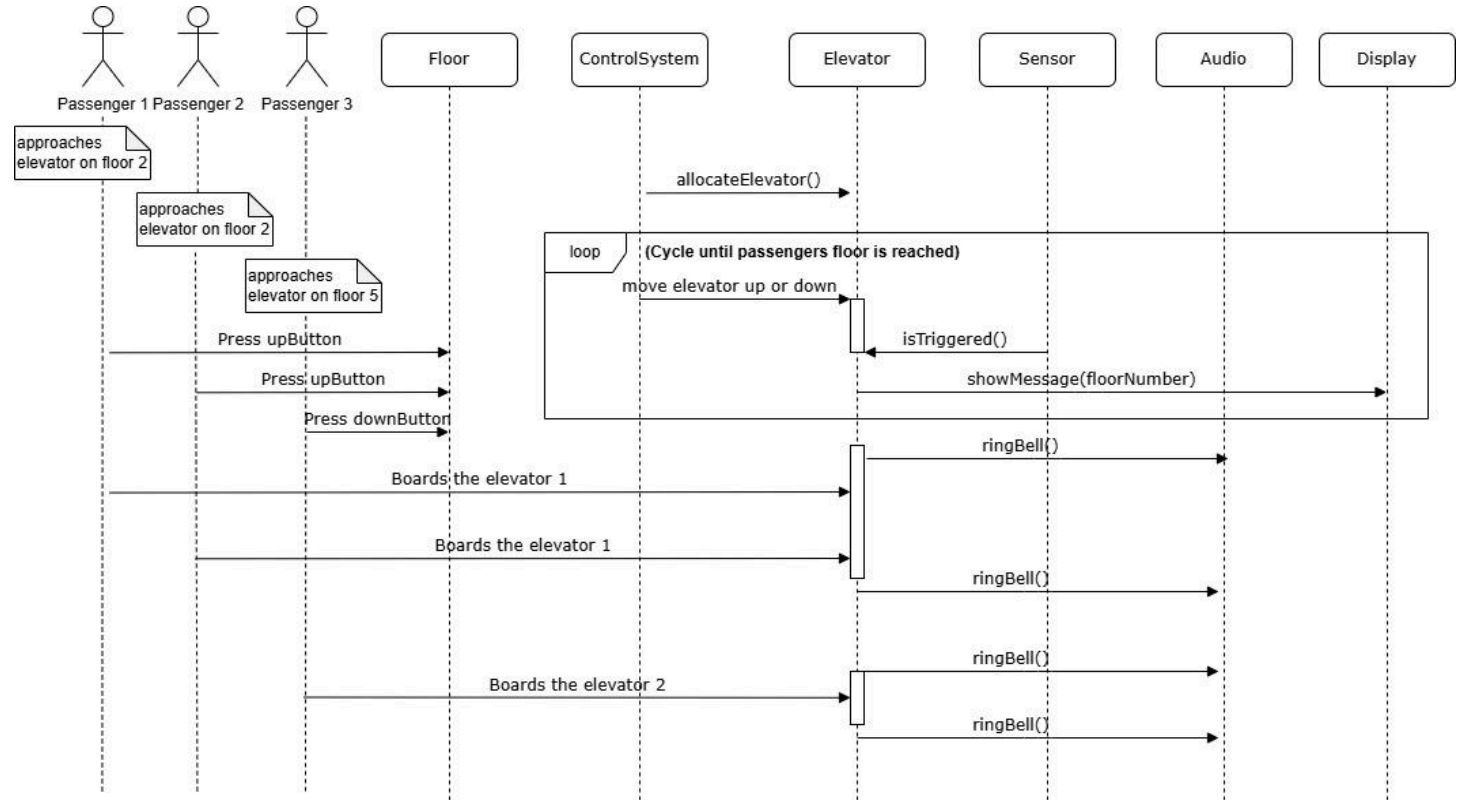Passengers: Covered by Passenger.

The UML class diagram for the elevator control system simulator, designed to meet the requirements and use case model. The diagram includes all elements from the requirements, follows UML syntax, and is consistent with the implementation.

(C++ header files are included that are consistent with the class diagram. Please zoom in for a clearer view.)
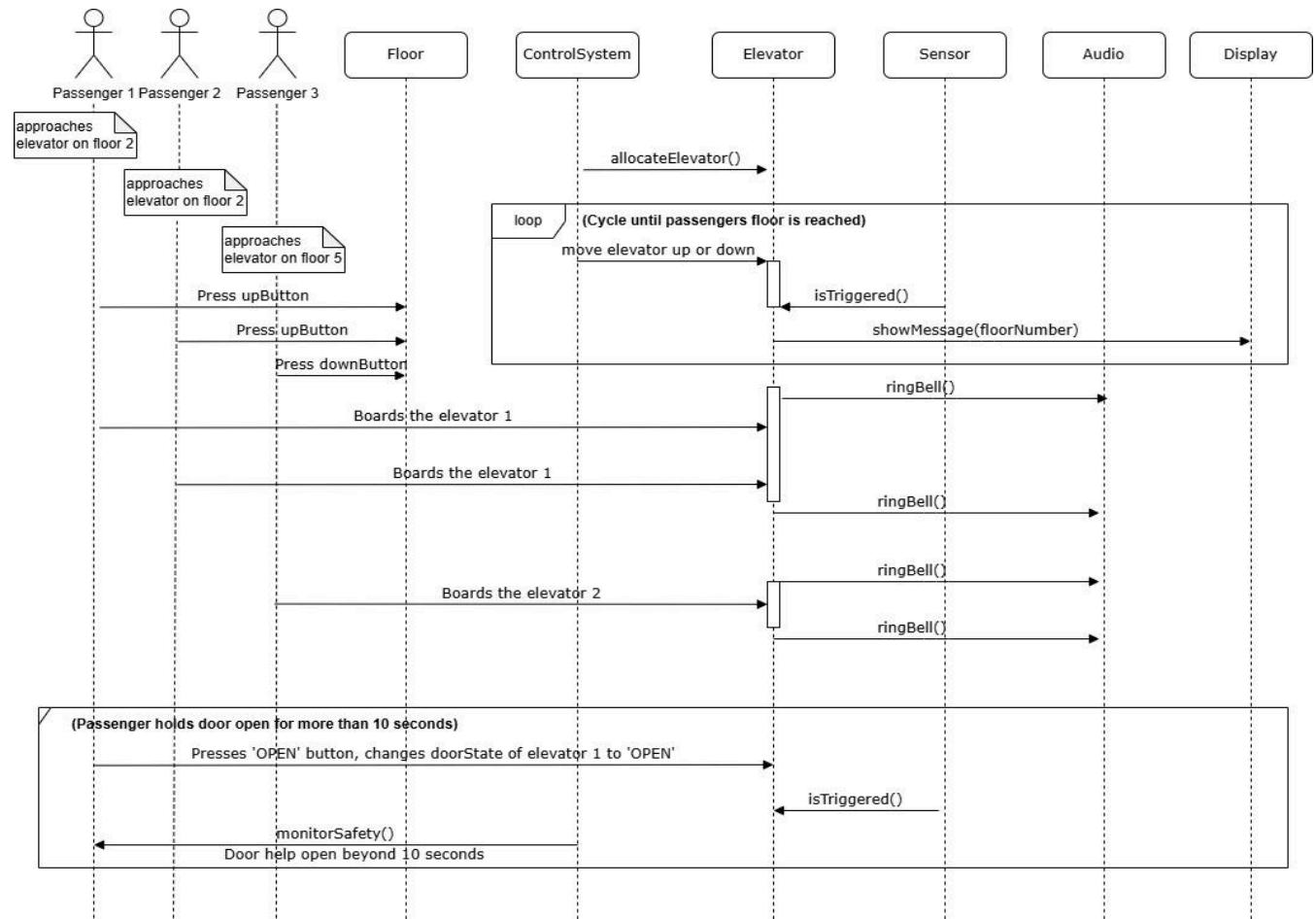
Elevator
- id: int
- currentFloor: int
- state: State
- occupied: bool

+ moveStep(x: int): void
+ closeDoor(): void
+ openDoor(): void
+ elevatorUpdated(id: int, floor: int, state: State)
+ updateLog(message: QString&)

Passenger
- id: int
- startFloor: int
- destFloor: int
- requestTime: int
- state: State
- assignedElevator: Elevator* = nullptr

+ requestElevator(): void
+ updateState(s: State): void
+ ElevatorRequestedOrBoarded(): void
+ setElevator(elevator: Elevator*): void
+ passengerUpdated(id: int, floor: int, state: State)

1 — used by — 1..*

MainWindow
- ui: Ui::MainWindow*
- controlSystem: ControlSystem*

+ AddToLog(message: std::string): void
- UpdateSimulationParameters(): void
- UpdatePassenger(id: int, floor: int, state: Passenger::State): void
- UpdateTime(time: int): void
- UpdateElevator(id: int, floor: int, state: Elevator::State): void
- On_ResetButton_Clicked(): void
- StartSimulation(): void
- StopSimulation(): void
- ContinueSimulation(): void
- PauseSimulation(): void
- ClearLogConsole(): void
- OpenButtonHandle(): void
- CloseButtonHandle(): void
- PowerOutHandle(): void
- FireHandle(): void
- HelpHandle(): void
- BlockageHandle(): void
- OverloadingHandle(): void

Sensor
+ detectObstacle(int id): void
+ detectOverloading(int id): void
+ updateLog(const QString &message): void

ControlSystem
- time: int
- paused: bool
- elevators: QVector<Elevator*>
- passengers: QVector<Passenger*>
- safetySystem: SafetySystem*
- sensor: Sensor*
- timer: QTimer

+ start(): void
+ stop(): void
+ pause(): void
+ continueSim(): void
+ resetSimulation(): void
+ setSinglePassenger(passengerIndex: int, startFloor: int, destFloor: int, timeStep: int): void\
+ triggerFireAlarm(): void
+ triggerPowerAlarm(): void
+ Help(elevatorId: int): void
+ openElevatorDoor(elevatorId: int): void
+ closeElevatorDoor(elevatorId: int): void
+ ObstacleAlarm(elevatorId: int): void
+ OverloadingAlarm(elevatorId: int): void
+ updateLog(message: QString&)
+ updateTime(time: int)
+ updateElevator(id: int, floor: int, state: Elevator::State)
+ updatePassenger(id: int, floor: int, state: Passenger::State)
- stepSimulation(): void

Safety System
- fireAlarmActive: bool
- powerAlarmActive: bool
- safeFloor: const int
- loggedElevators: QSet<int>
- loggedPassengers: QSet<int>

+ triggerFireAlarm(): void
+ triggerPowerAlarm(): void
+ stepFireOrPowerAlarm(elevators: QVector<Elevator*>&, passengers: QVector<Passenger*>&): void
+ resetSafetySystem(): void
+ updateLog(message: QString&)

manages
contains
interacts with
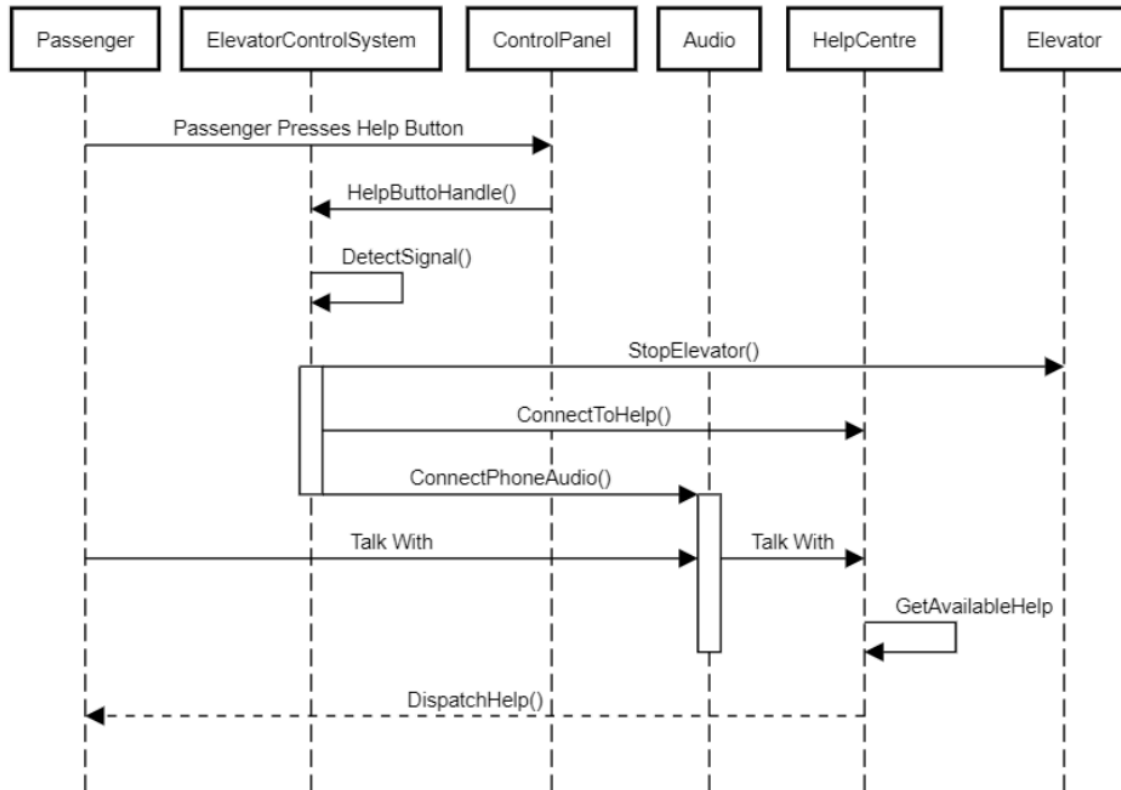
UML Sequence Diagrams-
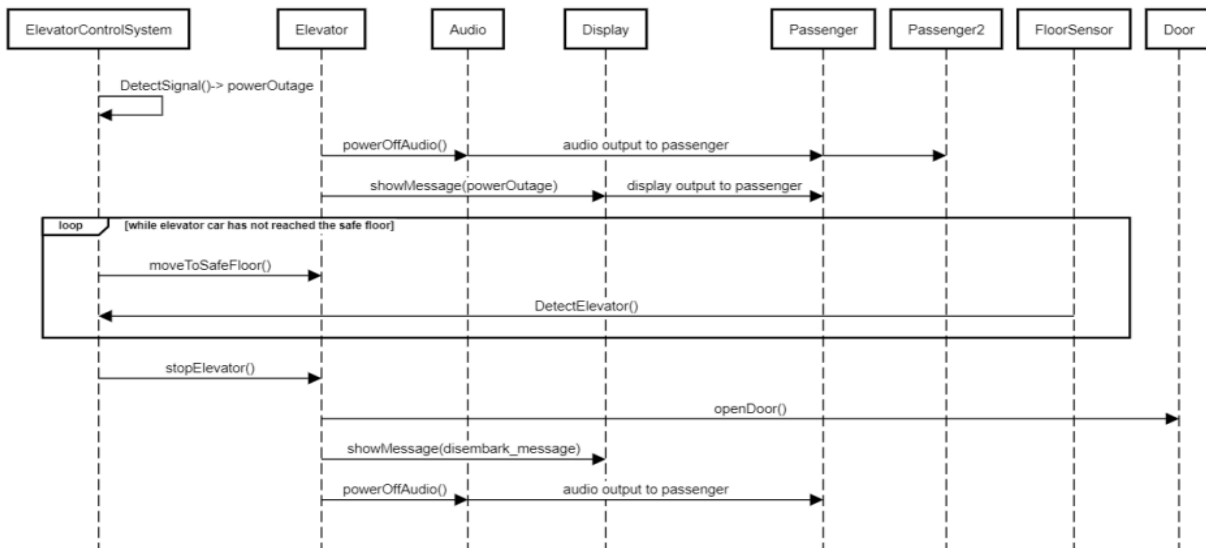## Success Scenario 1: Passengers Going in The Same Direction Ride

Success Scenario 2: Passengers holds elevator door open longer than 10 seconds
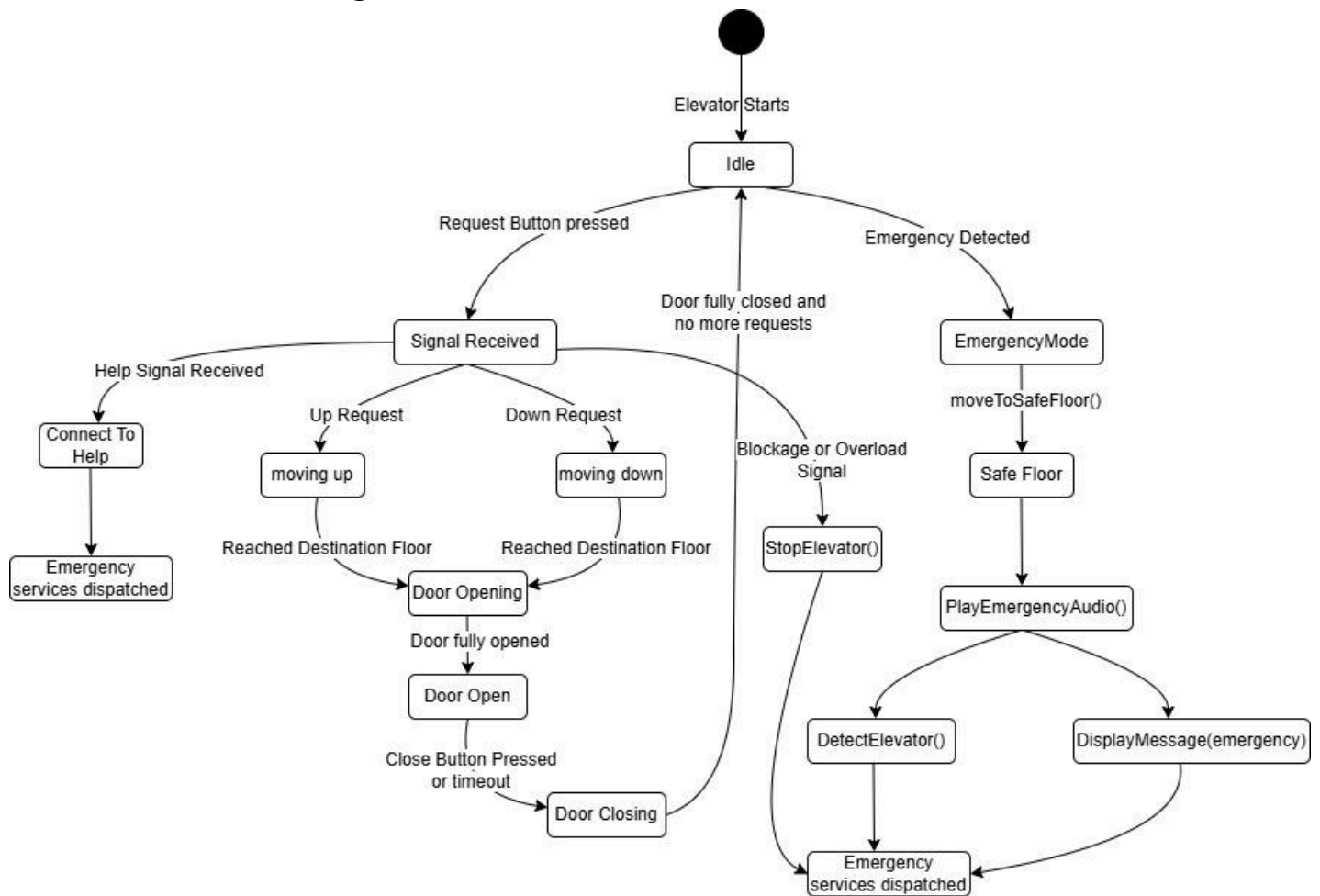
Safety Scenario 1: Passenger presses the 'HELP' button

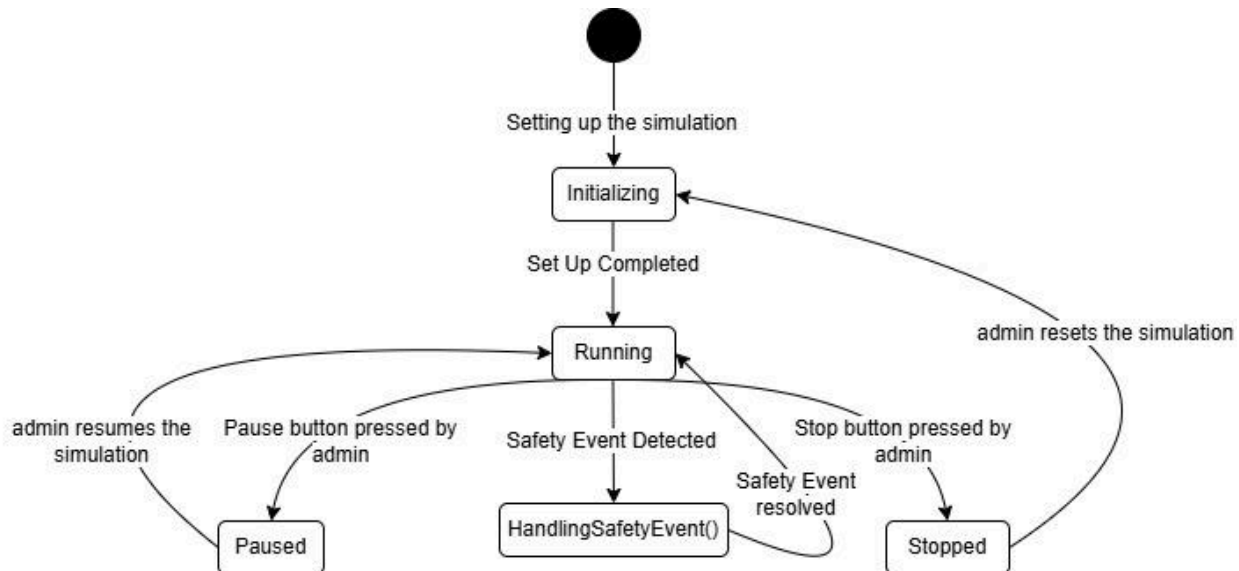| Passenger	ElevatorControlSystem	ControlPanel	Audio	HelpCentre	Elevator

Passenger Presses Help Button

HelpButtoHandle()

DetectSignal()

StopElevator()

ConnectToHelp()

ConnectPhoneAudio()

Talk With

Talk With

GetAvailableHelp

DispatchHelp()

Safety Scenario 2: Power Outage

ElevatorControlSystem	Elevator	Audio	Display	Passenger	Passenger2	FloorSensor	Door

DetectSignal()-> powerOutage

powerOffAudio()   audio output to passenger

showMessage(powerOutage)   display output to passenger

loop   [while elevator car has not reached the safe floor]

moveToSafeFloor()

DetectElevator()

stopElevator()

openDoor()

showMessage(disembark_message)

powerOffAudio()   audio output to passenger

UML State Machine Diagrams-
(next page)

Elevator State Machine Diagram-

Simulation Controller State Machine Diagram-

Traceability Matrix-

ID	Requirement	Implemented By	Tested By	Related Use Case
1	On each of the N floors, is a pair of "up" and "down" buttons that illuminates when pressed. The buttons remain illuminated until an elevator arrives to transport the passengers, who have requested an elevator going in a certain direction.	Floor, Elevator, ControlSystem	On the floor, press either the up or down button. User inputs a floor number which creates an elevator request	Use Case 2
2	When the elevator arrives, it rings a bell, opens its doors for a fixed time allowing passengers to exit or board, closes its doors and proceeds to another floor.	Floor,Elevator, ControlSystem, Sensor	Observe the elevator arriving or departing from the floor.	Use Case 3
3	Once on-board passengers select one or more destination floors using a panel of buttons; there is one button for every floor. The elevator has a display which shows passengers the current floor of the elevator.	Floor,Elevator, ControlSystem, Sensor, Display	Observe the elevator's display while it is in transit	Use Case 3
4	There is also a pair of buttons on the elevator control panel marked "open door" and "close door". These buttons can be used by a passenger to override the default timing of the doors. The door will remain open beyond its default period if the "open door" button is held depressed; the doors can be closed prematurely by pressing the "door close" button. Inside the elevator there is also a help button linked to	Elevator, ControlSystem	Demonstrating the elevator system's functionality by physically using the 'open door,' 'close door,' and 'help' buttons within the elevator. Verify that the 'open door' button extends the door-opening time as expected when held, the 'close door' button closes the doors prematurely, and the 'help' button initiates a	

	building safety service.		voice connection to building safety service, responding to user input accordingly.	
5	Each elevator has a sensor that notifies it when it arrives at a floor.	Floor, Sensor, Elevator, ControlSystem	Conduct an elevator journey, select a floor, and observe the elevator's response. As you approach and arrive at the selected floor, monitor the elevator to ensure that it accurately detects and notifies its arrival at the chosen floor.	
6	Each elevator has a display and an audio system. The display shows the current floor number and warning messages that are synced with audio warnings.	Display, Audio, Elevator, ControlSystem	Enter the elevator and observe the display as the elevator moves between floors. Verify that the display correctly shows the current floor number. Trigger a warning situation (such as repeatedly interrupting the door sensor) and verify that the warning message on the display is synced with the audio warning.	
7	Safety(Help): The control system receives a "Help" alarm signal from an elevator indicating that the "Help" button has been pressed.	Floor, Elevator, ControlSystem	Press the "Help" button in the elevator. Verify that the control system receives and correctly identifies the "Help" alarm signal from the specific elevator	Use Case 4
8	Safety(Help): The passenger is connected to building safety service through a voice connection. If there is no response from building safety within 5 seconds or if there is no response from a	Display, Elevator, Audio, ControlSystem	Initiate a "Help" request in the elevator. Verify that a voice connection is established with the building safety service. If there is no response	Use Case 4

			within 5 seconds, or if the passenger does not respond, check that a 911 emergency call is automatically placed.	
	passenger a 911 emergency call is placed.			
9	Door obstacles: If the light sensor is interrupted when the door is closing, the control system stops the door from closing and opens it.	ControlSystem, Sensor,Elevator	While the elevator door is closing, interrupt the light sensor. Verify that the control system stops the door from closing and instead opens it.	Use Case 5
10	Door obstacles: If the sensor is interrupted repeatedly over a short period of time, a warning is sounded over the audio system and a text message is displayed.	ControlSystem, Sensor,Floor, Elevator	Place a large object in between landing and the elevator doors. Observe the display and audio	Use Case 5
11	Safety(Fire): The control system receives a "Fire" alarm signal from the building and commands all elevators to move to a safe floor. Similarly, a "Fire" alarm signal from the elevator itself will cause that elevator to go to a safe floor. An audio and text message are presented to passengers informing them of an emergency and asking them to disembark once the safe floor is reached.	ControlSystem, Floor, Display, Audio, Elevator	Trigger a "Fire" alarm signal from the building and verify that all elevators move to a safe floor. Similarly, trigger a "Fire" alarm signal from an individual elevator and verify that the specific elevator moves to a safe floor. Verify that the audio and text messages are presented to passengers informing them of an emergency and asking them to disembark once the safe floor is reached.	Use Case 6
12	Overload: The control system receives an "Overload" alarm signal from an elevator if the sensors indicate that the passenger or cargo load exceeds the carrying capacity.	ControlSystem, Elevator	Load an elevator with a weight exceeding its carrying capacity and verify that the control system receives an "Overload" alarm signal.	Use Case 7

13	Overload: The elevator does not move and an audio and a text messages are presented to passengers asking for the load to be reduced before attempting to move again	ControlSystem, Elevator, Audio, Display	Load the elevator with a weight exceeding its carrying capacity. Verify that the elevator does not move and that an audio and text message are presented to passengers, asking for the load to be reduced before attempting to move.	Use Case 7
14	Power out: The control system receives a "Power Out" alarm signal. In that case, audio and text messages are presented to passengers informing them of the power outage.	ControlSystem, Audio, Elevator, Display	Simulate a power outage in the control system and verify that it receives a "Power Out" alarm signal. Check that an audio and text message are presented to passengers, informing them of the power outage.	Use Case 8
15	Power Out: Each elevator is then moved to a safe floor and passengers are asked to disembark via audio and text messages. The battery backup power is sufficient to do all of this.	ControlSystem, Audio, Elevator, Display	Continue the power outage simulation and verify that each elevator is moved to a safe floor. Check that passengers are asked to disembark via audio and text messages. Confirm that the battery backup power is sufficient to perform these actions.	Use Case 8

(A YouTube video is linked in the README file for the demo.)

-X-X-X-X-X-X-X-