Class

Elevator

-int elevatorId
-int currentFloor
-ElevatorState elevatorState
-bool isIdle
-int sockfd

+elevatorFunction(int elevatorId)
-receiveAssignment()
-moveToFloor(int floorNumber)
-openDoors()
-closeDoor()
sendStatus()

Floor

-vector<pair> requestList
-set<pair> sentRequests
-int timer
-int sockfd

+floorFunction()
-sendRequest(ElevatorMessage msg)

SchdulerElevator

+int id
+int position
+bool isMoving
+bool isIdle
+bool goingUp
+int passengerCount
+sockaddr_in address

uses

uses

TimeManager

+int currentTime

+updateTime(int newTime)

ElevatorState

IDLE
MOVING
STOPPING
DOOR_OPEN
DOOR_CLOSED

Schduler

-int sockfd
-SchedulerState schedulerState
-set<pair> processedRequests
-queue<ElevatorMessage> pendingRequests
-vetor<Elevator> elevators

+schdulerFunction()
-assignElevator()

ElevatorMessage

+int floorNumber
+int destination
+bool directionUp
+int assignedElevator
+int status
+int timestamp

+ElevatorMessage()
+ElevatorMessage(int floor, int dest, bool up, int elevator, int stat)
+ElevatorMessage(int floor, int dest, bool up, int elevator, int stat, int time)
+pinrt()

Schdulerstate

IDLE
PROCESSING
ASSIGNING

uses

```
     Floor              Schduler                        Elevator

Send elevator request(ElevatorMessage)

                          processRequest&selectElevator

                              Assign elevator task

                                                        Move to pick up floor

                                                        open/close doors

                                                        Move to destination f

                                                        open/close doors

                          Send completion status

                          Set to idle stae
```

Diagram 1

● (initial state)

IDLE

IDLE → PROCESSING : request recrived

PROCESSING → IDLE : Invalid request

PROCESSING → ASSIGING : valid request

ASSIGING → IDLE : Elevator Assigned

Diagram 2

● (initial state)

IDLE

IDLE → MOVING

MOVING → DOOR_OPEN

DOOR_OPEN → DOOR_CLOSE

DOOR_CLOSE → MOVING : Destination not reached

DOOR_CLOSE → IDLE : task completed