Scheduler Subsystem Dispatcher - elevators: ArrayList<Elevator> Nested Classifier \oplus + MAX_DIFF: integer + Dispatcher() Elevator + Dispatcher(elevNum: integer) + upDateElevatorInfo(id: integer, dir: Direction, floor: integer): boolean dir: Directions + getNearestElevator(dir: Direction, callingFloor: integer): integer - currFloor: int - id: int has a... + Elevator(id: int, dir: Directions, floor: int) + getFloor(): int + setFloor(floor: int): void Communicator + getDir(): Directions + setDir(dir: Directions): void # floorSocket: DatagramSocket + getId(): int # elevatorSocket: DatagramSocket # dispatcher: Dispatcher # TIMEOUT_TIME: int # elevatorReturnPort: int # floorPort: int # destinations: ArrayList<Set<Integer>> # deniedReqs: ArrayList<byte[]> - tempDeniedHolder: ArrayList
byte[]> # pendingReqs: ArrayList<byte[]> - currReqld: int Extends + Communicator() Extends + handleNewMessage(message: byte[]): boolean - processStatussReport(dir: byte, floorNum: byte, elevatorNum: byte): boolean - processConfirmation(yesNoVal: byte, floorNum: byte, elevatorNum: byte, id: byte): boolean ElevatorListener - openCloseDoor(openClose: byte, floorName: byte, elevatorNum: byte, dir: byte): boolean - processNewRequest(dir: byte, origFloor: byte, destFloor: byte): boolean - packet: DatagramPacket updateDistpatcher(): void # retryDeniedReqs(): void + ElevatorListener() - checkForMessages(): void + run(): void FloorListener - packet: DatagramPacket ScheduleRunner + FloorListener() - checkForMessages(): void + run(): void + main(args: String[]): void

Elevator Subsystem Elevator ElevatorMotor + MAX_FLOOR: int elv: Elevator + MIN_FLOOR: int currentElvState: ElevatorState + MAX_SERVICE_QUEUE_CAPACITY: int previousElvState: ElevatorState - elevNumber: Integer has a... - isDoorOpen: boolean + ElevatorMotor(elevator: Elevator) - currFloorPosition: Integer + run(): void - floorDestionation: Integer move(): void has. upList: SortedSet<Integer> serviceFloor(); void - downList: SortedSet<Integer> currentServiceList: SortedSet<Integer> - elevatorPassengerButtons: Set<Integer> - status: Direction - motor: ElevatorMotor <<enumeration>> - elevReciever: ElevatorReciever Elevator State + Elevator(elvNumber: Integer, elvRecieve: ElevatorReciever, initialFloor: int) STANDBY DOOR_CLOSE DOOR_OPEN + updateFloorToService(): boolean + updateDirection(): void + canServiceCall(floorRequested: integer): boolean MOVE + getServiceList(): SortedSet<Integer> STOP + setCurrentServiceList(direction: Directions): void + moveUp(): void + moveDown(): void + openDoors(): void + closeDoors(): void + addToPassengerButtons(floor: int): void + removeFromPassengerButtons(floor: int): void + addToServiceList(floor: int, direction: Directions): void 1..* has. + addToServiceList(floor: int): void + serviceFloor(): void + generateDoorOpenMsg(): byte[] + generateDoorCloseMsg(): byte[] + generateAcceptMsg(floorDest: int, id: byte): byte[] + generateDeclineMsg(floorDest: int, id: byte): byte[] ~ generateStatusMsg(): byte[] ~ generateOpenMsg(): byte[] ~ generateCloseMsg(): byte[] - getNextDirection(): Directions + getElevNumber(): Integer + getElevatorReciever(): ElevatorReciever + getCurrFloorPosition(): Integer + getFloorDestionation(): Integer + isServiceListEmpty(): boolean + getElevatorPassengerButtons(): Set<Integer> + getStatus(): Directions FlevatorReciever elevators: List<Elevators> - schedulerSocket: DatagramSocket ~ messagePort: int + ElevatorReciever() + recieverCommunicationLoop(): void + processSchedulerMsg(packet: DatagramPacket): void + addFloorToService(elevatorNumber: Integer, floor: Integer, direction: Directions): void + sendMessage(msg: byte[]): void

+ sendResponse(msg: byte[], port: int): void

+ getElevators(): List<Elevator> + main(args: String[]): void + closeSocket(): void

Floor Subsystem

Floor System floors: List<Floors> systemFile: SystemFile has. que: Queue<Message> datagramSocket: DatagramSocket STARTING_HOUR: int - STARTING MINUTE: int Floor + FloorSystem() - level: int + main(args: String[]): void - isUpButtonPressed: boolean + startFloorSchedule(): void isDownButtonPressed: boolean + schedulerCommunicationLoop(): void - floorDestinationButtonPressed: List<Integer> + getFloorObjectiveByLevel(floorLevel: int): Floor elevatorsAtFloor: List<ElevatorForFloor> + getFloors(): List<Floor> + printOutFloorInformation(floor: Floor, action: String): void + getQue(): Queue<Message> + Floor(level: int) + addFloorButtonPressed(floor: int): void + removeFloorButtonPressed(floor: int): void + getFloorDestinationButtonPressed(): List<Integer> + openDoor(): void + closeDoor(): void + getLevel(): int + isDoorOpen(): boolean + isUpButtonPressed(): boolean has.. + setUpButtonPressed(isUpButtonPressed: boolean): void + isDownButtonPressed(): boolean + setDownButtonPressed(isDownButtonPressed: boolean): void + isUpDirectionLampOn(): boolean + setUpDirectionLamp(isUpDirectionLampOn: boolean): void + isDownDirectionLampOn(): boolean + setDownDirectionLampOne(isDownDirectionLampOn: boolean): void + setLevel(level: int): void + getElevatorsAtFloor(): List<ElevatorForFloor> + setElevatorsAtFloor(elevatorsAtFloor: List<ElevatorForFloor>): void ELevatorForFloor index: int

- isDoorOpen: boolean
- isUpDirectionLampOn: boolean
- isDownDirectionLampOn: boolean
- + ElevatorForFloor(index: int)
- + isDoorOpen(): boolean
- + setDoorOpen(isDoorOpen: boolean): void
- + isUpDirectionLampOn(): boolean
- + setUpDirectionLampOn(isUpDirectionLampOn: boolean): void + isDownDirectionLampOn(): boolean
- + setDownDirectionLampOn(isDownDirectionLampOn: boolean): void
- + getIndex(): int
- + setIndex(index: int): void

Constants

- + ELEVATOR_PORT: int
- + FLOOR_PORT: int
- + MESSAGE_LENGTH: int
- + NUMBER_OF_ELEVATORS: int
- + LOWEST FLOOR: int
- + HIGHEST_FLOOR: int + NUMBER_OF_FLOORS: int
- + ELEVATOR_TRAVEL_SPEED_MS: int
- + ELEVATOR_STOP_TIME: int + SCHED_IP_ADDRESS: String
- + FLOOR_SYS_IP_ADDRESS: String + ELEVATOR_SYS_IP_ADDRESS: String
- + FORMATTER: SimpleDateFormat
- + ERROR: byte
- + NEW_REQUEST_FROM_FLOOR: byte
- + REQUEST_RECEIVED: byte
- + OPEN_CLOSE_DOOR: byte + NEW_ELEVATOR_DESTINATION: byte
- + ELEVATOR_INFO_REQUEST: byte
- + CONFIRM_VOL_DESTINATION: byte
- + STATUS_REPORT: byte
- + YES: byte
- + NO: byte
- + OPEN: byte
- + CLOSE: byte
- + MANDATORY: byte
- + VOLUNTARY: byte
- Constants()

Resources

<<enumeration>> Directions

UP DOWN STANDBY

- + isOpposite(dir1: Directions, dir2: Directions): boolean
- + getDirByInt(val: int): Directions
- + getIntByDir(dir: Directions): int

Message

- time: LocalTime
- startingFloor: int
- direction: Directions
- destinationFloor: int
- messageQueue: Queue<Message>
- ELEMENTS_IN_MESSAGE: int
- + Message(time: LocalTime, startingFloor: int, direction: Direction, destinationFloor: int)
- + getTime(): LocalTime
- + getStartingFloor(): int
- + getDirection(): Directions
- + getDestinationFloor(): int
- + getMessageQueue(): Queue<Message>
- + toString(): String

SystemFile

- + FILEPATH: String
- + FILENAME1: String
- filename: String
- + SystemFile(filename: String)
- validateFloorRange(floor: int): boolean
- + readValidateAndCreatMessages(): void
- validateLine(lineInfo: String): boolean
- readFile(): ArrayList<String>
- + testReadFile(): ArrayList<String> + testValidateLine(lineInfo: String): boolean