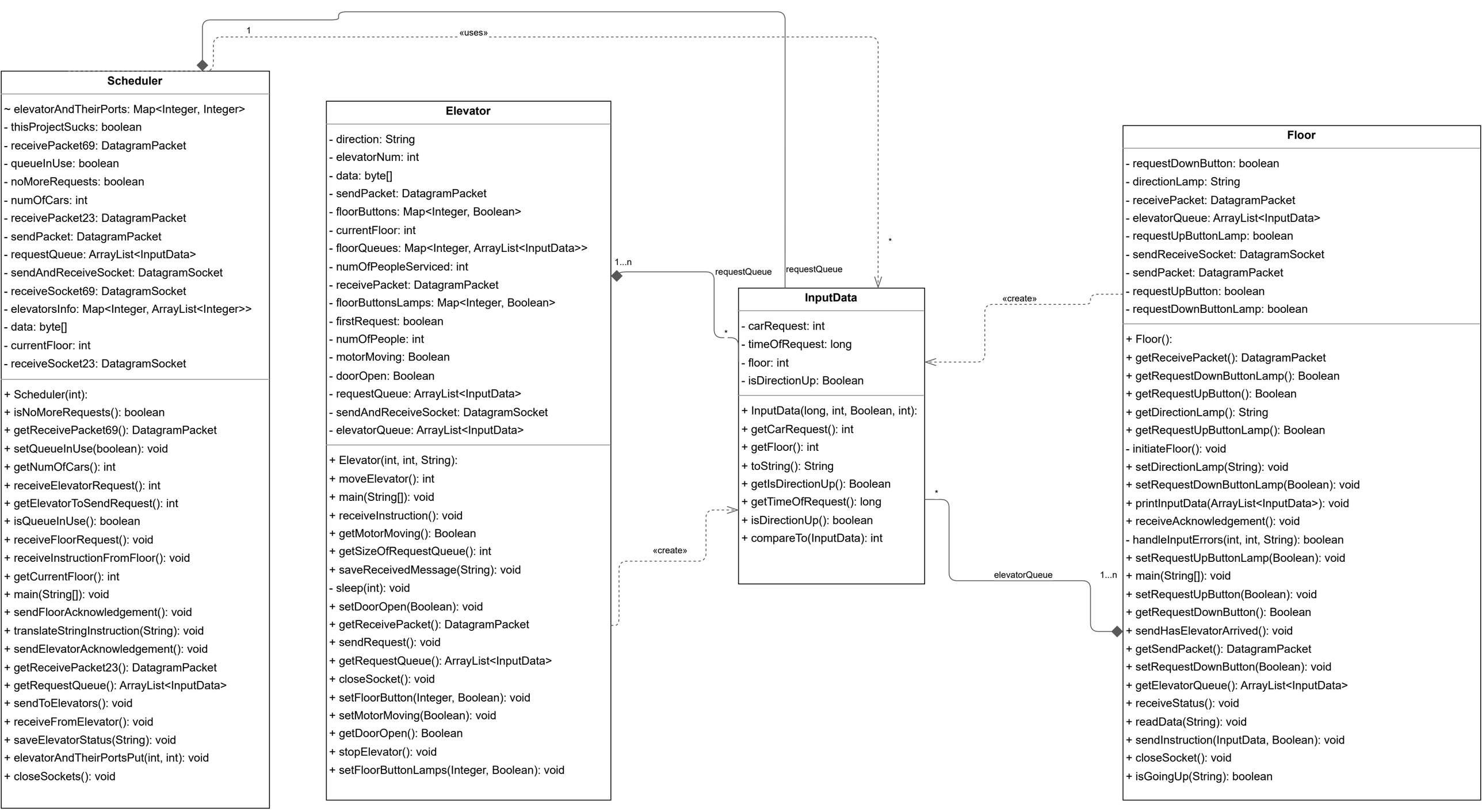
1
«uses»
Scheduler
~ elevatorAndTheirPorts: Map<Integer, Integer>
- thisProjectSucks: boolean
- receivePacket69: DatagramPacket
- queueInUse: boolean
- noMoreRequests: boolean
- numOfCars: int
- receivePacket23: DatagramPacket
- sendPacket: DatagramPacket
- requestQueue: ArrayList<InputData>
- sendAndReceiveSocket: DatagramSocket
- receiveSocket69: DatagramSocket
- elevatorsInfo: Map<Integer, ArrayList<Integer>>
- data: byte[]
- currentFloor: int
- receiveSocket23: DatagramSocket
+ Scheduler(int):
+ isNoMoreRequests(): boolean
+ getReceivePacket69(): DatagramPacket
+ setQueueInUse(boolean): void
+ getNumOfCars(): int
+ receiveElevatorRequest(): int
+ getElevatorToSendRequest(): int
+ isQueueInUse(): boolean
+ receiveFloorRequest(): void
+ receiveInstructionFromFloor(): void
+ getCurrentFloor(): int
+ main(String[]): void
+ sendFloorAcknowledgement(): void
+ translateStringInstruction(String): void
+ sendElevatorAcknowledgement(): void
+ getReceivePacket23(): DatagramPacket
+ getRequestQueue(): ArrayList<InputData>
+ sendToElevators(): void
+ receiveFromElevator(): void
+ saveElevatorStatus(String): void
+ elevatorAndTheirPortsPut(int, int): void
+ closeSockets(): void
Elevator
- direction: String
- elevatorNum: int
- data: byte[]
- sendPacket: DatagramPacket
- floorButtons: Map<Integer, Boolean>
- currentFloor: int
- floorQueues: Map<Integer, ArrayList<InputData>>
- numOfPeopleServiced: int
- receivePacket: DatagramPacket
- floorButtonsLamps: Map<Integer, Boolean>
- firstRequest: boolean
- numOfPeople: int
- motorMoving: Boolean
- doorOpen: Boolean
- requestQueue: ArrayList<InputData>
- sendAndReceiveSocket: DatagramSocket
- elevatorQueue: ArrayList<InputData>
+ Elevator(int, int, String):
+ moveElevator(): int
+ main(String[]): void
+ receiveInstruction(): void
+ getMotorMoving(): Boolean
+ getSizeOfRequestQueue(): int
+ saveReceivedMessage(String): void
- sleep(int): void
+ setDoorOpen(Boolean): void
+ getReceivePacket(): DatagramPacket
+ sendRequest(): void
+ getRequestQueue(): ArrayList<InputData>
+ closeSocket(): void
+ setFloorButton(Integer, Boolean): void
+ setMotorMoving(Boolean): void
+ getDoorOpen(): Boolean
+ stopElevator(): void
+ setFloorButtonLamps(Integer, Boolean): void
1...n
requestQueue
*
requestQueue
*
«create»
InputData
- carRequest: int
- timeOfRequest: long
- floor: int
- isDirectionUp: Boolean
+ InputData(long, int, Boolean, int):
+ getCarRequest(): int
+ getFloor(): int
+ toString(): String
+ getIsDirectionUp(): Boolean
+ getTimeOfRequest(): long
+ isDirectionUp(): boolean
+ compareTo(InputData): int
«create»
*
elevatorQueue
1...n
Floor
- requestDownButton: boolean
- directionLamp: String
- receivePacket: DatagramPacket
- elevatorQueue: ArrayList<InputData>
- requestUpButtonLamp: boolean
- sendReceiveSocket: DatagramSocket
- sendPacket: DatagramPacket
- requestUpButton: boolean
- requestDownButtonLamp: boolean
+ Floor():
+ getReceivePacket(): DatagramPacket
+ getRequestDownButtonLamp(): Boolean
+ getRequestUpButton(): Boolean
+ getDirectionLamp(): String
+ getRequestUpButtonLamp(): Boolean
- initiateFloor(): void
+ setDirectionLamp(String): void
+ setRequestDownButtonLamp(Boolean): void
+ printInputData(ArrayList<InputData>): void
+ receiveAcknowledgement(): void
- handleInputErrors(int, int, String): boolean
+ setRequestUpButtonLamp(Boolean): void
+ main(String[]): void
+ setRequestUpButton(Boolean): void
+ getRequestDownButton(): Boolean
+ sendHasElevatorArrived(): void
+ getSendPacket(): DatagramPacket
+ setRequestDownButton(Boolean): void
+ getElevatorQueue(): ArrayList<InputData>
+ receiveStatus(): void
+ readData(String): void
+ sendInstruction(InputData, Boolean): void
+ closeSocket(): void
+ isGoingUp(String): boolean