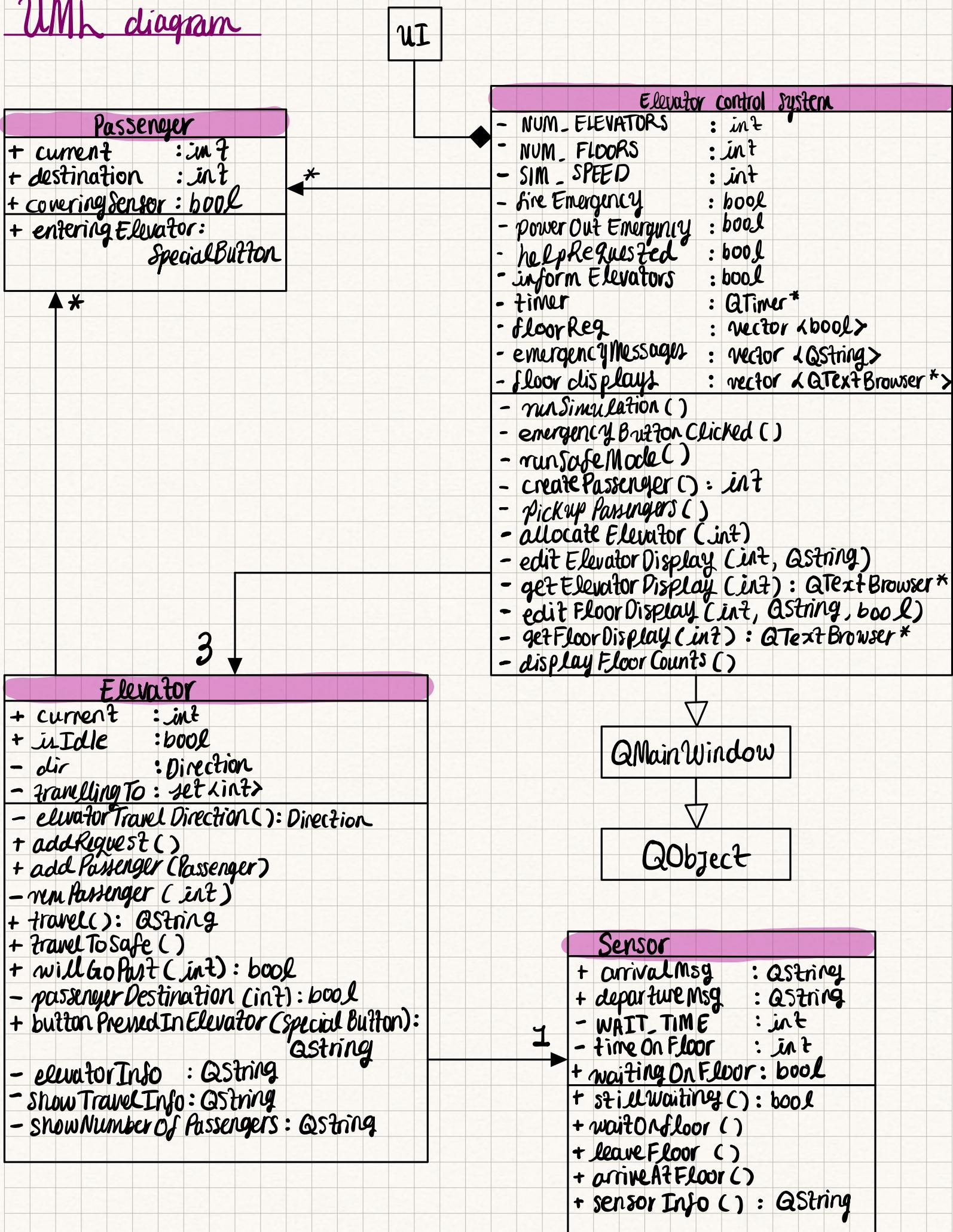
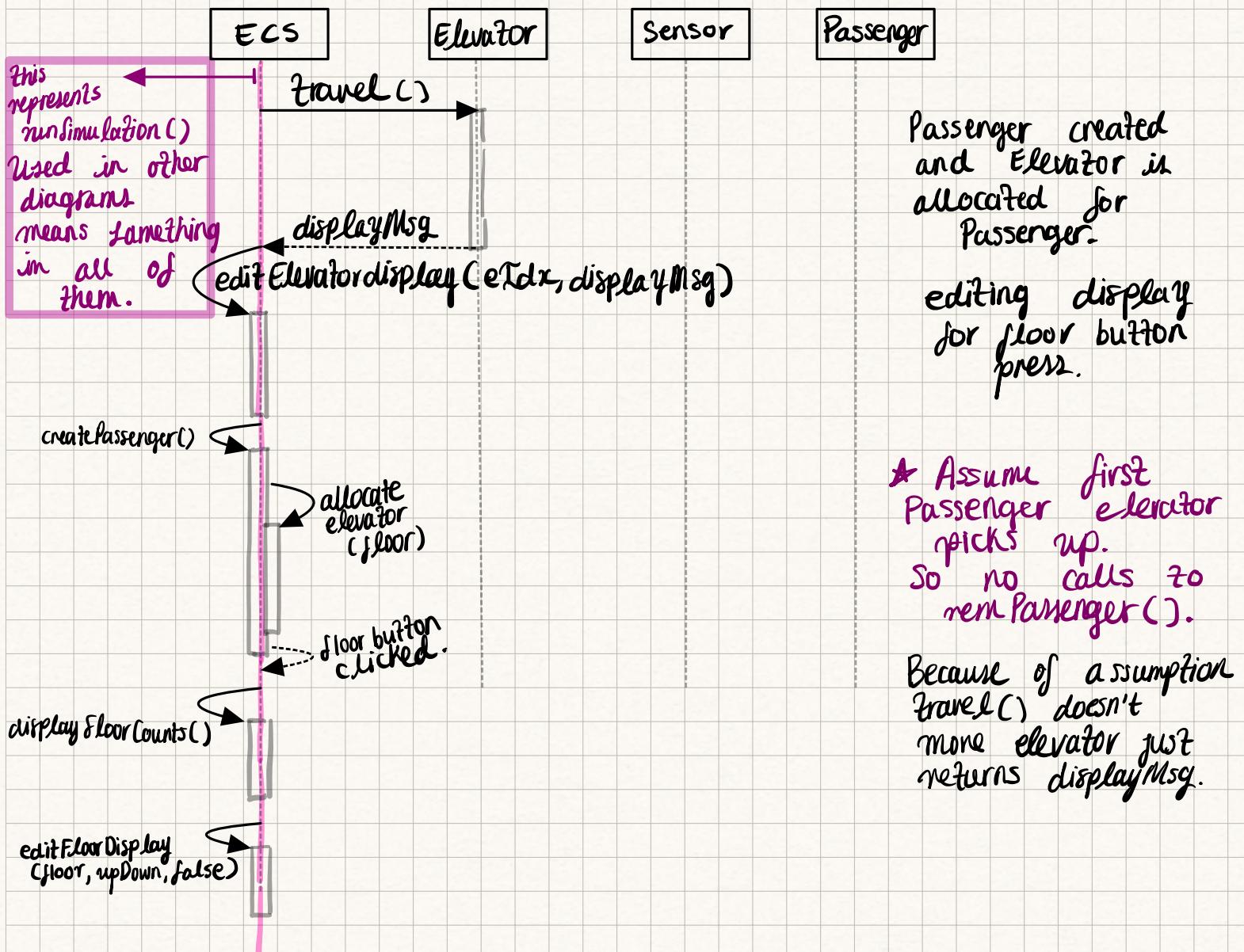


UML diagram

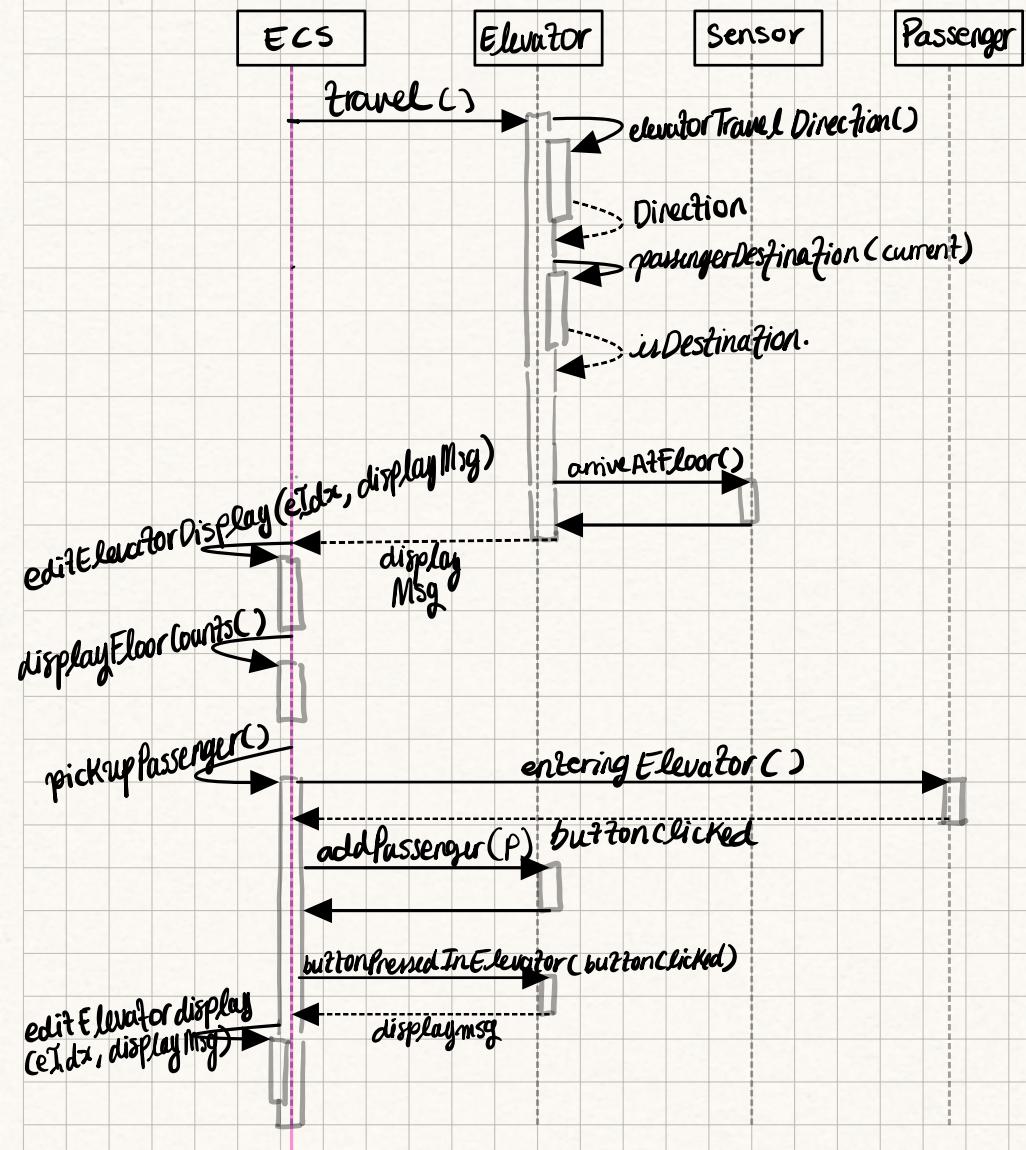


SEQUENCE DIAGRAMS

Basic Use Case: Passengers gets picked up by elevator (I)



Basic Use Case: Passengers gets picked up by elevator (2)



Elevator reaches Passenger and picks them up from floor.

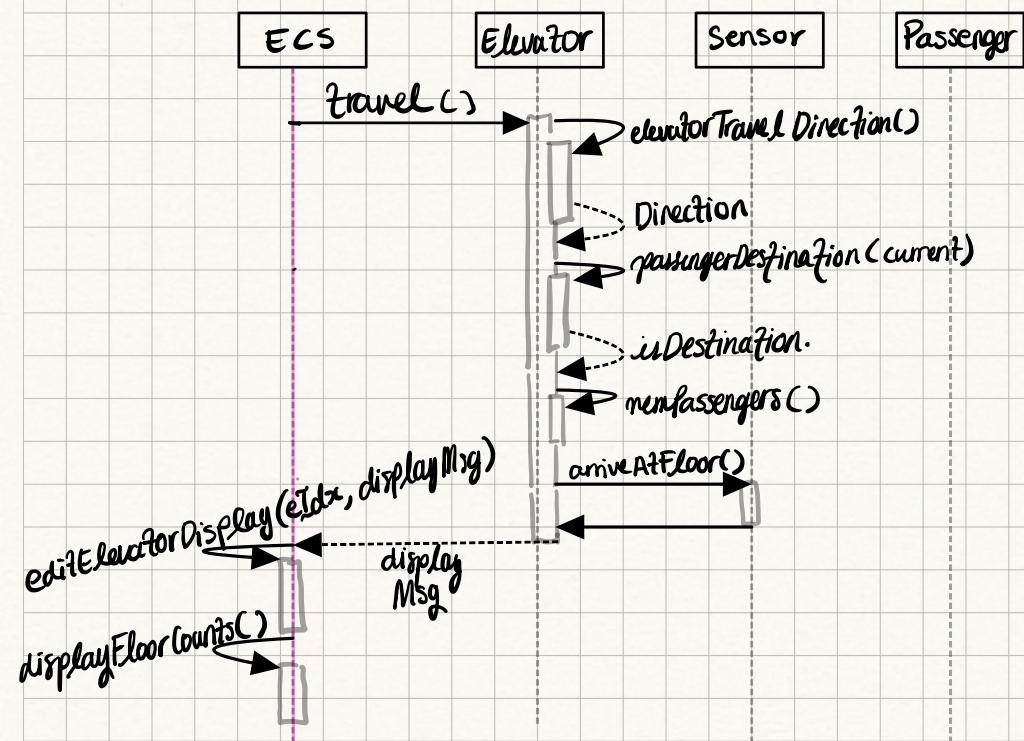
passengerDestination() returns false

→ elevator told to travel to passenger floor which is why **passengerDestination()** called.

BUT no passengers in elevator so **newPassenger()** is not called

* Skipping or not **createPassenger()** relevant.

Basic Use Case: Passengers Gets Dropped off (3)



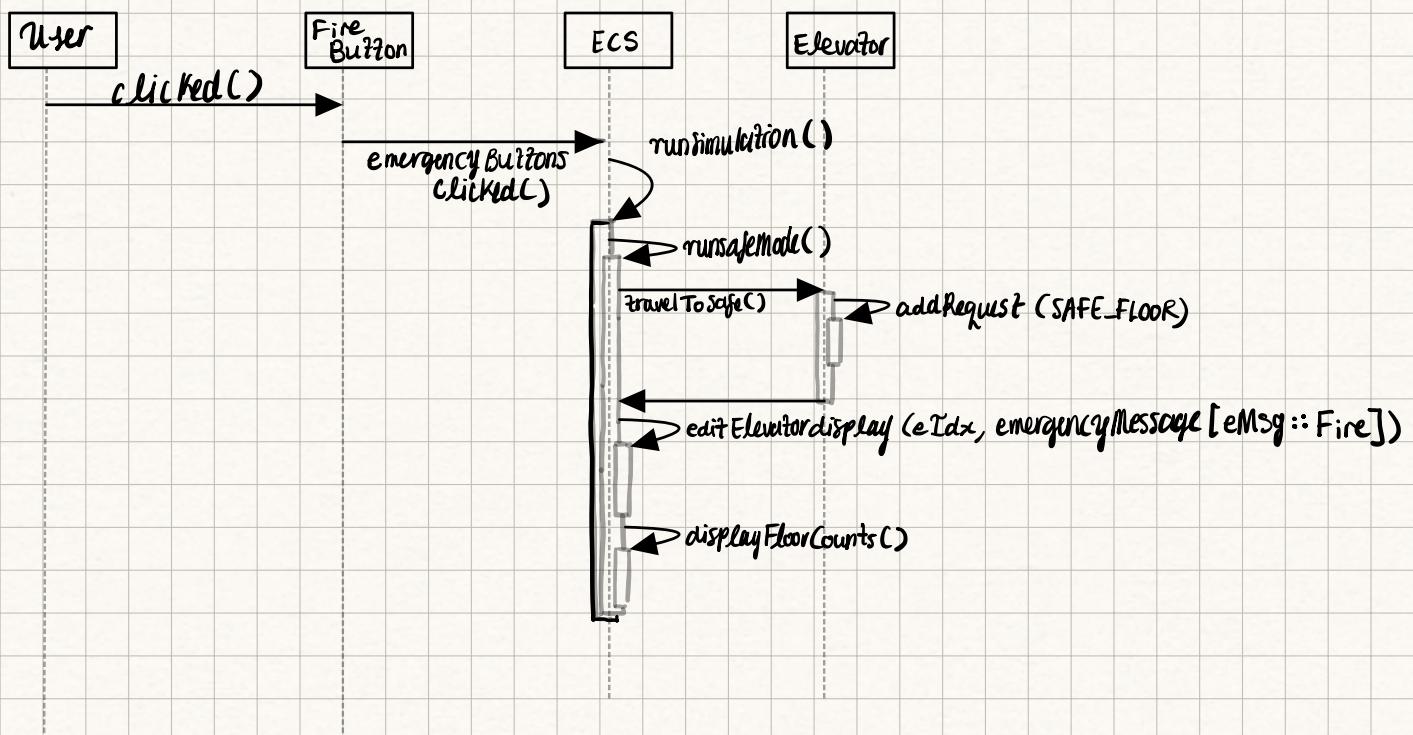
Elevator reaches Passenger destination and drops them off.

passengerDestination (current) returns true hence why removePassenger() is called. current floor is a destination for passenger

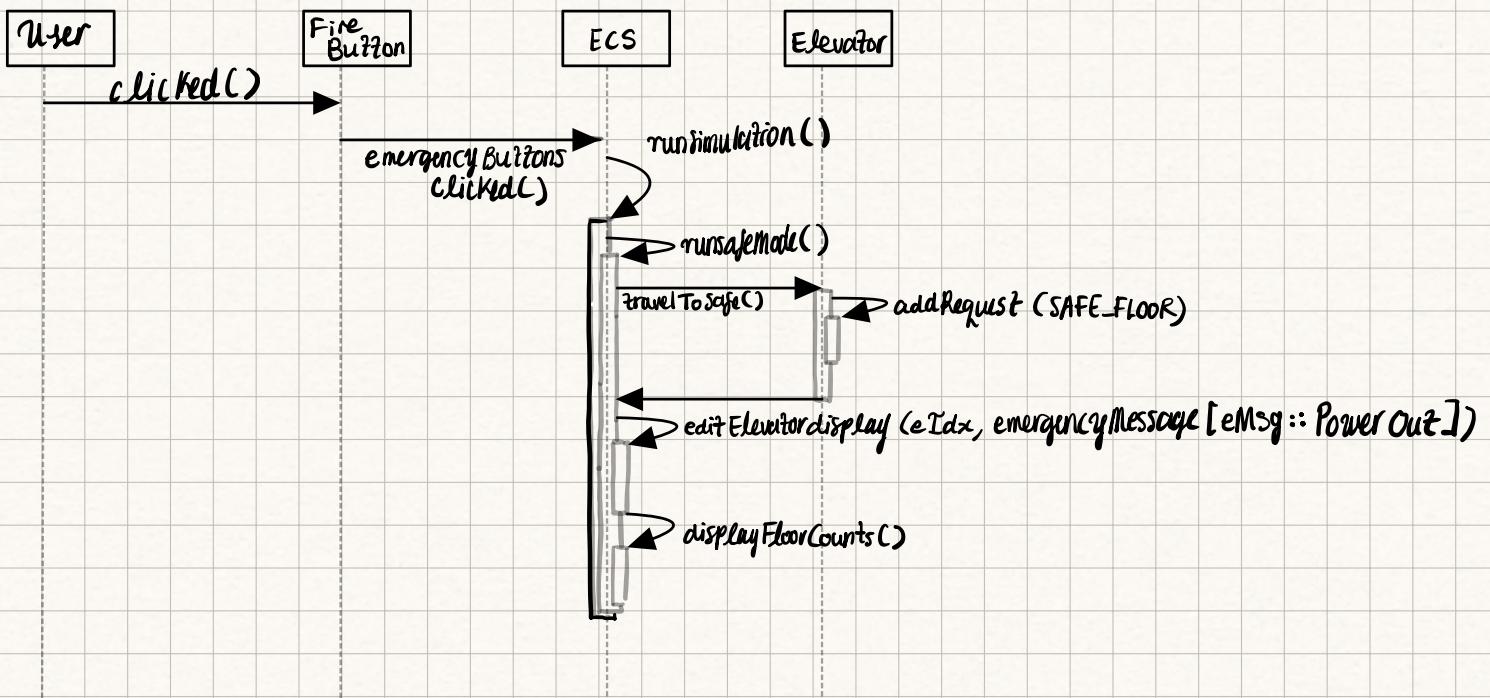
* Skipping `createPassenger()` as not relevant.

* Skipping `pickupPassenger()` as not relevant.

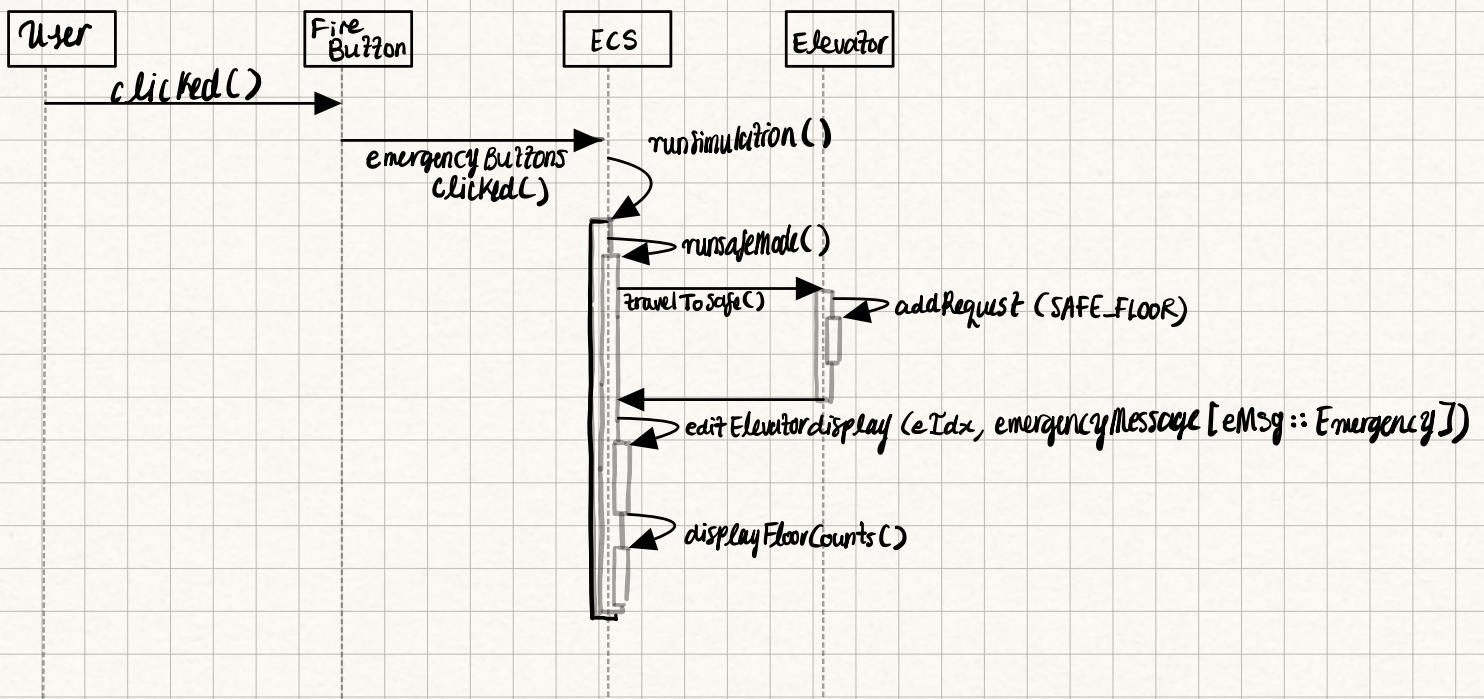
Fire Emergency use-case:



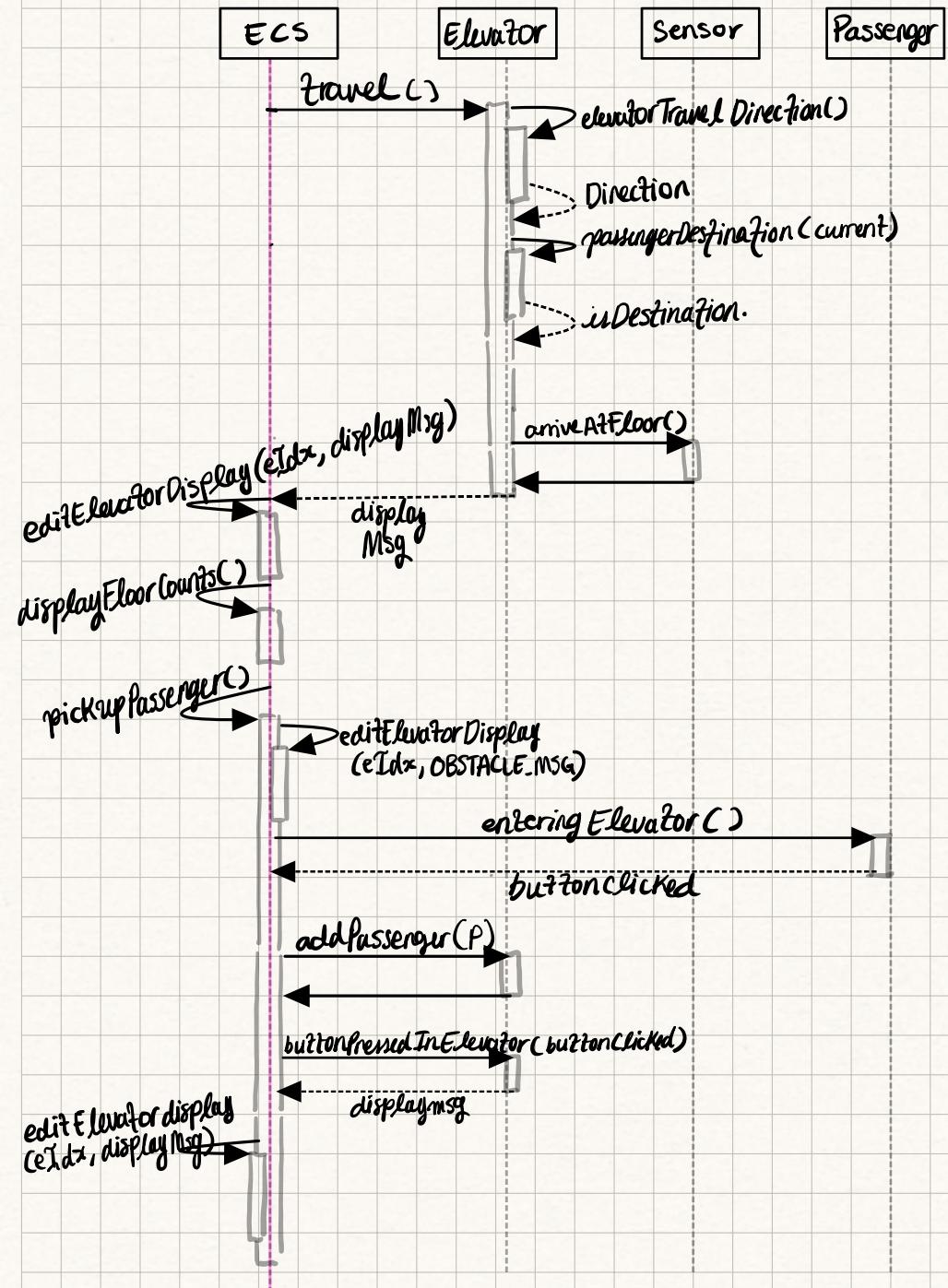
PowerOut use-case



Help use-case:



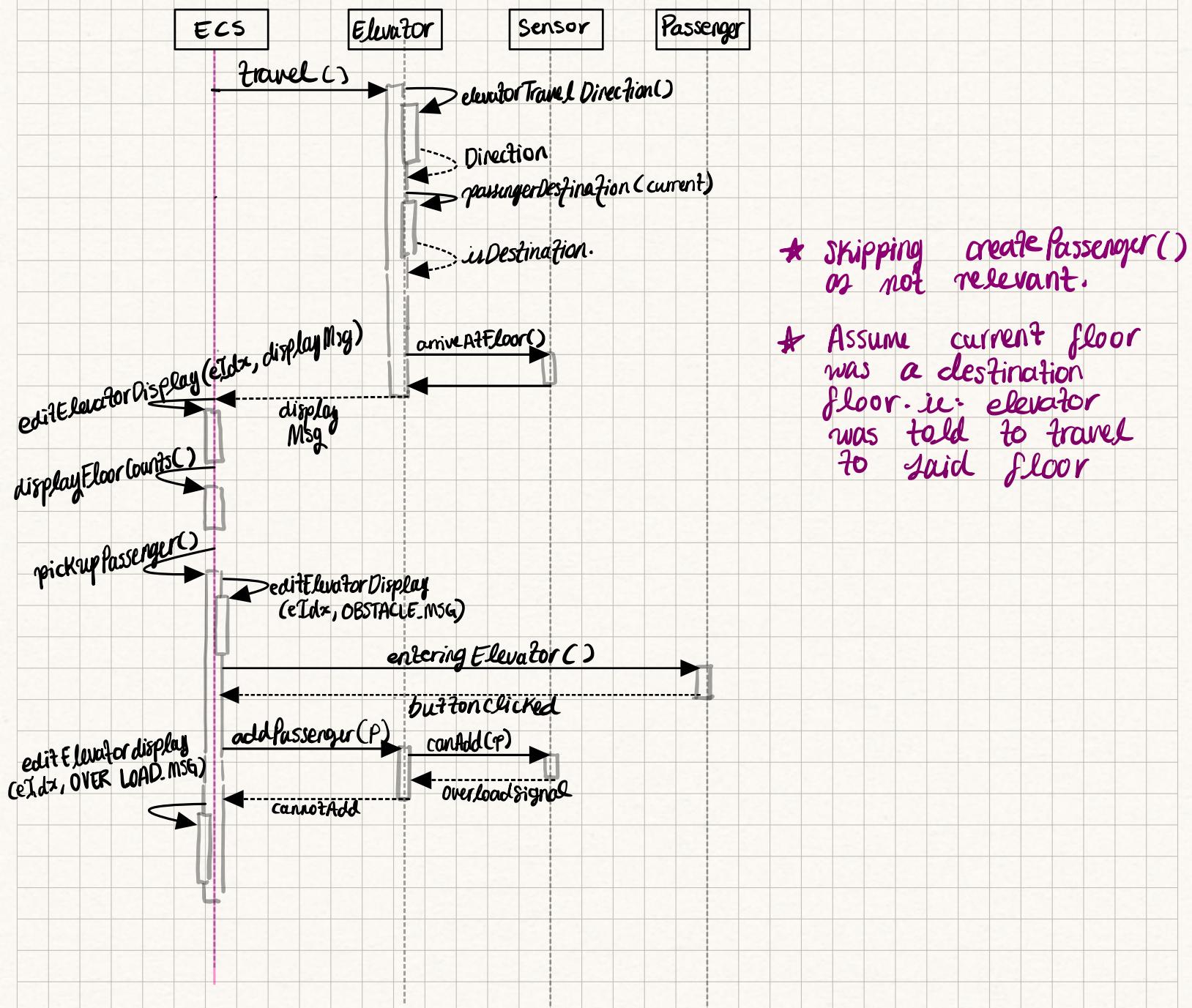
Door obstacle use case:



For obstacle all that is done is a message is displayed in the elevator display and then passengers are added as normal.

- * Skipping `createPassenger()` or not relevant.
- * Assume current floor was a destination floor. i.e. elevator was told to travel to said floor hence why `passengerDestination()` called

Overload use case:



* Skipping or not relevant.

* Assume current floor was a destination floor. i.e. elevator was told to travel to said floor