ID	Requirement	Use Case	Fulfilled by	Test	Description
1		Normal	CarModel, MainWindow	Run the application, select normal button	Starts Normal use case scenario with Passengers already established. Sequence continues on reselection of Normal, or every 10 seconds using QTimer. Each cycle calls the elevator's move() function. Each cycle attempts to add passengers to and elevators upcoming passengers.
2	Help Safety Feature	Help	CarModel, MainWindow	Run the application, select normal button, then select help. Selecting help again to establish connection with building safety	MainWindow sends a send_help signal to the CarModel, which triggers the receive_help slot. The program uses helpCounter to keep track of time since the help scenario was triggered to detect if a 911 call must be made. Reselecting the help button before this occurs assumes contact with the building safety was established.

3	Door Obstacle Safety Feature	Door Obstacle	CarModel, MainWindow	Run the application, select normal button, then select door obstacle. Selecting door obstacle again will remove the obstacle	MainWindow sends a send_door signal to the CarModel, which triggers the receive_door slot in CarModel. The program uses blockedCounter to keep track of how long the door has been blocked, and if it has occurred for too long it will warn passengers.
4	Overload Safety Feature	Overload	CarModel, MainWindow	Run the application, select normal button, then select Overload. Reselecting the Overload button will toggle the scenario and allow the program to continue as normal.	MainWindow sends a send_overload signal to the CarModel, which is handled by the slot receive_overload. This will stop the elevator from moving until the elevator is no longer overloaded.
5	Fire Safety Feature	Fire	CarModel, MainWindow	Run the application, select normal button, then select Fire	MainWindow sends a send_fire signal to the CarModel, which the CarModel handles using the receive_fire slot. This causes the elevator to reassess its movement strategy in order to reach the safe floor. Passengers can no longer be added to the upcoming passengers.
6	Power Out Safety Feature	Power Out	CarModel, MainWindow	Run the application, select normal button, then select Power Outage	MainWindow sends a send_power signal to the CarModel, which the CarModel handles using the receive_power slot. This causes the elevator to reassess its movement strategy in order to reach the safe

					floor. Passengers can no longer be added to the upcoming passengers.
7	Required for Normal	Boarding	CarModel, MainWindow	Run the application, select Normal	When the elevator arrives at the requesting passengers' floor, the passenger’s destination is added to the elevator’s destinations.
8	Requirement for Normal	Moving between floors	CarModel, Passenger	Run the application, select Normal	When moving between floors the elevator calculates its direction, notifies the passengers of floor changes and when passengers leave, updates the elevator’s required destinations
9	Requirement for Normal	Exiting	CarModel, Passenger	Run the application, select Normal	When the elevator arrives at a destination, it checks if any passengers are at their destination floor, if so, the passenger is removed from passengers