The setting. You're given a system with a number of use-cases that were elicited from its potential users.

The goal. Write down a short version of the software requirements specification document. The short SRS must include a list of features presented in Structured Natural Language format, with the following fields present in each feature: *Function, Description, Inputs, Source, Outputs, Destination, Action, Requirements, Pre-condition, Post-condition, and Side Effects*. An example may be found in Ian Sommerville's Software Engineering, chapter Requirements specification, p. 97.

Hints: You need to figure out the possible domain and system entities and their interfaces, and write them down. Use common sense and be parsimonious (no need to describe extra objects). Think who the users might be, except for the obvious case, and write them down.

Hints on evaluation: Your set of requirements will be evaluated by your peers on the standard requirements quality criteria (clarity, unambiguity, completeness, verifiability, consistency).

Variant 1. Specify the software system controlling a system of lifts.
Core use-cases: calling a lift from outside, calling a floor from inside, calling alarm from inside
Optional use-cases: calling several lifts with one button, have display updated with lift position for each floor, having one lift reserved for service purposes.

The setting. You're given a system with a number of use-cases that were elicited from its potential users.

The goal. Write down a short version of the software requirements specification document. The short SRS must include a list of features presented in Structured Natural Language format, with the following fields present in each feature: *Function, Description, Inputs, Source, Outputs, Destination, Action, Requirements, Pre-condition, Post-condition, and Side Effects*. An example may be found in Ian Sommerville's Software Engineering, chapter Requirements specification, p. 97.

Hints: You need to figure out the possible domain and system entities and their interfaces, and write them down. Use common sense and be parsimonious (no need to describe extra objects). Think who the users might be, except for the obvious case, and write them down.

Hints on evaluation: Your set of requirements will be evaluated by your peers on the standard requirements quality criteria (clarity, unambiguity, completeness, verifiability, consistency).

Variant 2. Specify the software system controlling a smart fitness bracelet.
Core use-cases: measuring heart rate and blood pressure, exporting daily measurements to a PC, beeping when heart rate exceeds a threshold, understanding activity type (resting, walking, and running)
Optional use-cases: setting up a timer, accepting calls by pairing with phone

The setting. You're given a system with a number of use-cases that were elicited from its potential users.

The goal. Write down a short version of the software requirements specification document. The short SRS must include a list of features presented in Structured Natural Language format, with the following fields present in each feature: *Function, Description, Inputs, Source, Outputs, Destination, Action, Requirements, Pre-condition, Post-condition, and Side Effects*. An example may be found in Ian Sommerville's Software Engineering, chapter Requirements specification, p. 97.

Hints: You need to figure out the possible domain and system entities and their interfaces, and write them down. Use common sense and be parsimonious (no need to describe extra objects). Think who the users might be, except for the obvious case, and write them down.

Hints on evaluation: Your set of requirements will be evaluated by your peers on the standard requirements quality criteria (clarity, unambiguity, completeness, verifiability, consistency).

Variant 3. Specify the software system controlling a smart home.
Core use-cases: turning on/off heating/cooling, turning on/of lights, heating/cooling the room to a specified temperature for a specified time, streaming surveillance video over the Internet
Optional use-cases: exporting temperature measurements to a PC