Quality Attributes and Architectural Design

## Brainstorm

Different roles

* User, producer, legislators, environmentalists

Reliable/accurate measurements from thermometers, within 0,5oC

Data quality check to filter out measurements from the sensors if the temperature varies more than 5oC within a minute.

If the gateway fails there should be a backup gateway

Be able to ask the actuators for their status, or have the actuators report their status every minute

The thermometers should be able to provide a temperature meassure every second.

The gateway should be able to receive a temperature reading from 100 thermometers every second.

The gateway should be able to turn up to 100 actuators on or off every sedond.

Should be able to add new types of actuators and sensors without changing the design or architecture of the system.

The system should be able to act as input to other devices. If temperature rises suddenly, possibility of fire, alarm on.

## Prioritization

I think that crucial for a system like HS07 is the performance quality attributes. If the measurements from the sensors aren’t accurate or if there are too few sensors, to get a realistic picture of the temperature in the home, then the system won’t satisfy the needs of the customers. Most of the quality attributes from the brainstorm are performance quality attributes. Of these I’ll rate the requirements for accuracy in the meassurements, data-quality checks and that many sensors and actuators can be used at the same time highest.

Important but not as crucial are the interests of the producer of such a system. They want modifiability, so that the system can be used with a variety of sensors and actuators. Thereby the application can fit more customers, and hence has a bigger sale potential. So after the performance quality attributes i rate modifiability with respect of adding new types of sensors and actuators.

## Most important Quality Attribute scenarios

Accuracy

Data quality

Many sensors

|  |  |
| --- | --- |
| Scenario | Measurements from each thermometer should be within 0,5oC of the actual temperature |
| Relevant Quality Attributes | Performance |
| Source | Thermometer |
| Stimulus | Measures temperature |
| Artifact | System |
| Environment | Normal operation |
| Response | Current temperatur |
| Response measure | The measured temperature is within 0,5oC of the temperature measured by a calibrated thermometer. |
| Questions |  |
| Issues | Requires calibrated reference thermometers |

|  |  |
| --- | --- |
| Scenario | Up to 100 sensors and 100 actuators should be able to be on the system at the same time, with measurements every second |
| Relevant Quality Attributes | Performance |
| Source | The gateway |
| Stimulus | Requests temperature |
| Artifact | Thermometer |
| Environment | Normal operation |
| Response | Current temperature |
| Response measure | 100 measurements pr. second |
| Questions | Is it nessecary to measure the temerature every second? Would it be enough to measure once every minute? |
| Issues |  |

|  |  |
| --- | --- |
| Scenario | Up to 100 sensors and 100 actuators should be able to be on the system at the same time, with measurements every second |
| Relevant Quality Attributes | Performance |
| Source | The gateway |
| Stimulus | Turns the actuators on |
| Artifact | Actuators |
| Environment | Normal operation |
| Response | Actuators are turned on |
| Response measure | Latency less than 1 second |
| Questions | Is it beneficial to have a larger latency to avoid the actuators flip-flopping when the temperature is close to max and min limits? |
| Issues | Might be relevant to require actuators that can accept broadcast messages. |

|  |  |
| --- | --- |
| Scenario | Adding new types of actuators |
| Relevant Quality Attributes | Modifiability |
| Source | The producer |
| Stimulus | Wants to add a new type of actuator |
| Artifact | System |
| Environment | At design time |
| Response | The new actuator type is added with no side affects |
| Response measure | Less than an hour |
| Questions | What basic requirements must actuators comply with? |
| Issues |  |

## Evaluation

Its not easy to estimate how well the performance scenarios can be fulfilled bye HS07, because these by large depend on the hardware the system is running on. The operations of getting a list of temperature measurements, calculating a mean value and turn the actuators on or off are very simple and neither data nor computationally intensive.

That said, there is room for performance optimizing. The logic for deciding wheter to turn on or off the radiators is in the radiators themselves. It would be more efficient if this logic was in the gateway, so that the gateway only communicates with the actuators when they need to be turned on or off, and all the potentially many actuators won’t need to do the same comparison agains the max and min temperature values. An additional benefit is that this way the actuators will get simpler, benefitting the modifiability. On the other hand, if the customers want to be able to differentiate the wanted temperature in different rooms of a house, it can be argumented that the decision logic is best placed in the actuators.

The modifiability scenario is well fullfilled by HS07. The gateway dosen’t have any direct notion of the radiators. It only has a list of observers. It would be “prettier” if there was a actuator interface that all radiators implemented. If the mechanism that decides wheter to turn on or off the actuators is moved to the gateway, the job of the actuator would be simpler, allowing for more types of actuators. This clearly makes the system more modifiable

## Redesign

I’ll keep the same basic structure of the system but change the way the actuators are controlled. This means moving the settings for the wanted temperature from the actuators to the gateway and the control mechanism that decides wheter to turn the actuators on or off.

An interface should be made for all the controllers (gateway, sensor and actuator) that has the getController method. An additional interface that extends the controller interface for the sensors and one for the actuators. These should specify the getTemperature and setState methods that all sensors respectively actuators should implement.

The movement of the control mechanism has been implemented, and also the actuators are only notified when they need to change status.