**Requirements of the system**

This system will be made up of three sub-systems (database, smart devices and web service.) All systems must communicate and exchange information successfully using the deployment of IoT devices to enforce interoperability.

**MySQL Database Sub-System**

* A database will be used within the system to store and analysis data from the users IoT devices.
* The system will perform C.R.U.D operations in order to store and handle data successfully within the tables.
* This system must store information from IoT devices (e.g. fridges, heating systems, water and energy monitoring systems) to be displayed on the client machines.
* This system must be able to successfully store and handle data from the web service (server).

**Smart Devices Sub-System**

* This system must allow the user to view the list of devices they have, viewing their current and past status.
* It must be able to successfully display the web service providing the information on the clients IoT devices.
* Each device must display the same consistent content on each smart device the user wishes to use.
* This system must be able to successfully connect and navigate the web service system (View device, Add device etc.)

**Web Service Sub-System**

* This system must support at least three computing devices such as laptops, desktops and smartphones running on different operating systems, and allow for extendibility of more devices.
* It must allow seeing trends over time for each device’s data which is produced from the simulated IoT devices (e.g. the usage of each device.)
* The system must allow for extendibility and maintainability of all aspects (e.g. functionality of the design interface.)
* This system must be able to successfully connect and send data to the database system for storage and analysis.
* This system must allow at least three IoT devices to be monitored, and allow for extendibility of more devices. (It must have an appropriate capacity.)

**Usability**

* The interface must be easy to learn how to use and easy to remember how to use.
* Users must not need to consult a manual each time they want to use the web service. (Should be easy to use the first time around without instructions)
* (Buttons, headings and navigational bars etc. must be easy to understand and use)
* Low perceived workload: the interface must appear easy to use, rather than demanding, time consuming and frustrating.

**Performance**

* The web service should be very responsive and have a quick response time.
* The web service should be able to deal with a high work load if required. (Unlimited number of IoT monitored at once.)
* Scalability is required as this will be a new system and could gain an unexpected growth in popularity and increase in workload as it becomes used by more and more users.

**Interface**

* It must take into consideration all types of users with disabilities and the interface should support accessibility.
* Elements must be easy to use and understand. A navigational bar must be used in order for users to easily navigate the web service without any problems.
* Content must be simple and easy to read. Not too much content which will result in the user going elsewhere.

**Security**

* The system must allow confidentiality to all of its users. (The system shall only show information and data on IoT to authorized users.)
* Access control – the system must contain a login of some kind to provide protection and privacy for users.

**Difference between phase 1 and 2 of requirements:**

In contrast to the first requirements documentation, this second documentation contains more refined and in depth requirements of the system. It contains requirements about usability, performance, interface and security which weren’t present in the first draft. These requirements were discovered to be needed when in the designing stages of the development (wireframes).