

Institiúid Teicneolaíochta Cheatharlach



At the Heart of South Leinster

Utility Watch

Project Report

Student Name: Ray Shannon
Student ID: C00079959
Student e-mail: c00079959@itcarlow.ie
Supervisor: Paul Barry

BSc(Honours) in Software Development (4th Year)
Institute Of Technology Carlow,
Kilkenny Road,
Carlow.
Date: 17/04/2015

Department of Computing and Networking,
Institute Of Technology Carlow

Abstract

The purpose of this document is to provide a description of the submitted project. The document will contain an introduction to the project, a detailed description of the submitted project, conformance to specification and design, learning achievements and an overall review of the project.

Table of Contents

1. Introduction	4
2. Utility Watch – Project Description.....	4
2.1. What is Utility Watch?	4
2.2. System Description	6
2.2.1. Hardware.....	6
2.2.2. Software.....	6
2.3. Application Description – Front-End.....	7
2.3.1. Electricity.....	8
2.3.2. Heating.....	9
2.3.3. Control	11
3. Conformance to Specification & Design	12
3.1. Initial Project Proposal (October ‘14)	13
3.2. Defining the Scope	13
3.3. Conformance.....	14
Where were the changes?	14
4. Learning Outcomes	14
4.1. Technical achievements.....	14
Research.....	15
Knowledge Gain	15
4.2. Personal Achievement	15
One Task at a Time.....	15
Character Building.....	15
Project Management	16
5. Project Review	16
What went right?.....	16
What went wrong?	16
What could have been better?	17
Would I take a different approach?.....	17
What advice would I give?	17
6. Acknowledgements.....	18

1. Introduction

This document will provide a detailed and accurate description of the Utility Watch application. The document will have 4 major sections to it; project description, conformance to specification and design, learning outcomes and a full project review.

The project description will cover all features and functionalities of the project including hardware, software and the user interface.

Conformance to specification and design will discuss the difference between the initial proposed specifications and the actual submitted specifications, if any. If differences exist, the reasons and justifications for these changes will be discussed.

Learning outcomes will be reviewed on 2 specific levels; a technical level, describing your achievements on what new technologies or processes you learned, and a personal level, describing how the project may have changed you as an individual.

The project review will highlight and identify the correct and incorrect decisions in relation to technology that were made, what went right or wrong along the way, if any aspect of the project was incomplete, and what advice would be given to another party attempting a similar project.

2. Utility Watch – Project Description

2.1. What is Utility Watch?

Utility Watch is an application which enables any property owner to access their electricity consumption information anytime, anywhere, once they have internet access. It displays current live values and also provides historical information using graphs. The application also provides cost estimation for the selected period of usage.

Utility Watch also empowers the property owner with the ability to remotely control their heating and lighting systems. The control of the heating system is complimented with a live temperature gauge. Below is an overview diagram of the project.

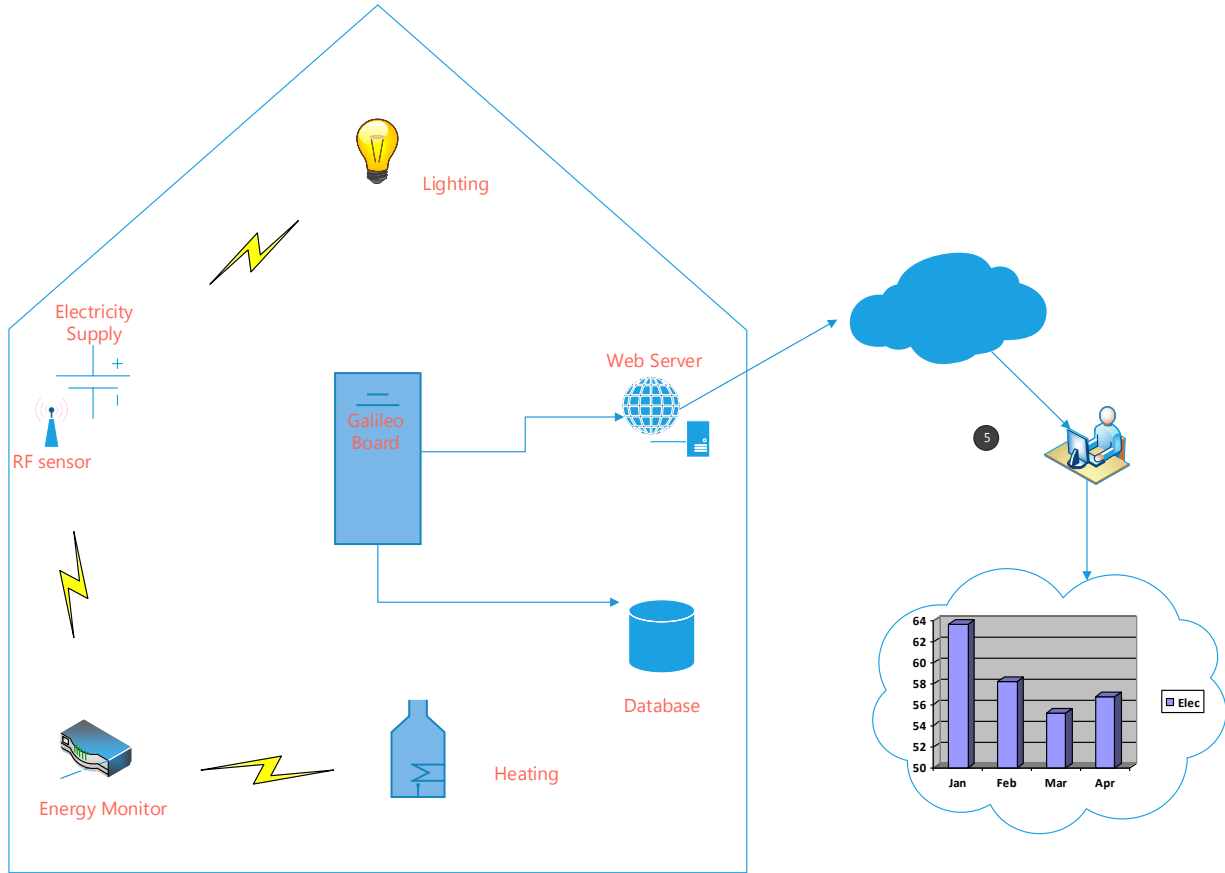


Fig. 1 System Overview

Below is a screenshot of the weekly chart display. The GUI will be discussed at a later stage.



Fig. 2 Weekly Chart Display

2.2. System Description

Utility Watch has a blend of disciplines that were required to achieve its goals. The project has a hardware element to it in the form of Intel's Galileo Board and Sailwider's Energy gateway; it has a large software presence controlling the GUI and also taking care of the backend data; and it also required intricate networking configuring to enable connectivity to the outside world.

2.2.1. Hardware

At the center of the project is Intel's Galileo Board Gen 1 machine. It is the brains of the project and everything that happens, goes through this device. The Galileo is employed as the web server, the database and the controller to all the listeners. It has a basic Linux OS which enabled the installation of Flask (a python web server) and SQLite (an embedded SQL database engine). It also is the point of contact with the local area network administrator when request are generated.

Another important piece of hardware was the energy gateway, a 3rd party device which transmitted the electricity consumption, allowing the Galileo to intercept the data. The energy meter monitored the premise's current draw on the mains supply and, using an RF protocol, relayed the data. I had no control over this device but I was able to intercept and parse the blocks of data it transmitted. The sensors used to control the heating and lighting systems were also RF controlled. I was able to send request to the energy gateway to tell these relays to open or close.

2.2.2. Software

The main component of the project was the software section. It controls the board, retrieves the necessary data, generates the various tables required to display the graphs, setting up network configurations and carries out the requests of the GUI. It is designed to a model-view-controller (MVC) architecture.

The backend software had a vital role in the project in the way of retrieving and generating data. I had to create several services and scripts to continuously operate to maintain reliable and accurate data used by the application. These services had to work together and use each other's data to create a hierarchical structure of tables as one table fed off the next and so on.

These services were also time dependent. For example, one service would log data every 20 seconds to the main database in the form unit reading and timestamp. Another service would log data every hour; this would be an average value for the previous hour. There were more services updating data every midnight and every Sunday at midnight. All of these services were critical as any error or issue would generate a problem at the frontend.

The flask server is another vital service that must be running constantly. If this service drops out the application is not contactable. In this situation the system needs rebooting. There

are scripts that are setup to run on any system reboot, ensuring the server gets deployed along with the previous mentioned data services.

2.3. Application Description – Front-End

The main application software is a python based flask app, blended with html templates, JQuery and javascript web services, SQLite queries for generating the graph data, and AJAX calls which update live data. It is running on the server waiting for user interaction in the form of a GUI. The initial interaction from the user must be a login. The system requires the user to enter a valid username and password. For the purpose of demonstration, the default username is **'admin'** with a default password of **'password'**. Once the login details are accepted, the user is shown a dashboard page which allows them to navigate to their preferred destination, as shown in Fig 3 below.

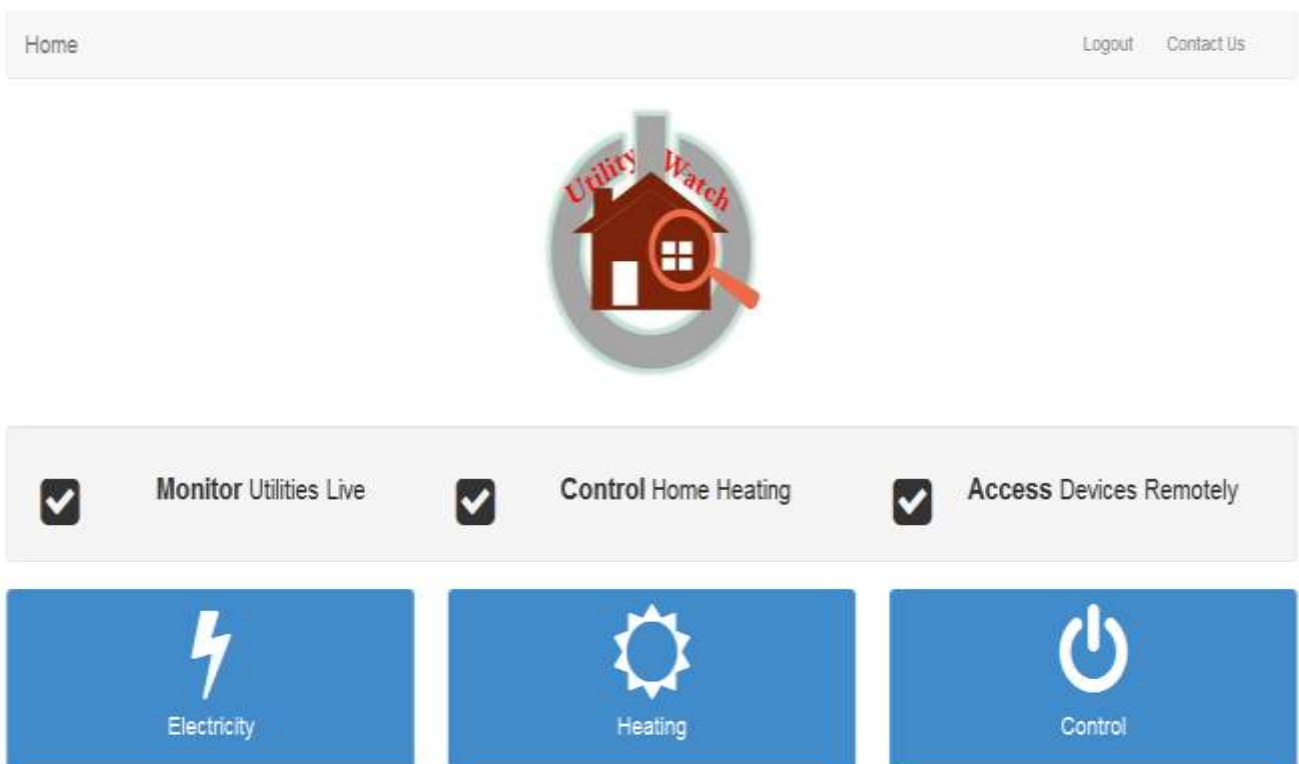


Fig. 3 Dashboard Screen

The dashboard displays the 3 main services provided by the project, each being discussed separately.

2.3.1. Electricity

The initial electricity page presents the user with the current day’s energy consumption. Upon selection, the app requests the data from the ‘DAY’ table using SQLite; it passes the retrieved data, in JSON format, to the HighCharts API and then renders the graph. The graph displays the data in hourly slots which helps the customer to identify the periods of the day where most of the energy is used.

A table is also produced in the right column which tracks the running cost of the day period. It also shows the current usage amount in watts as displayed in Fig. 4.

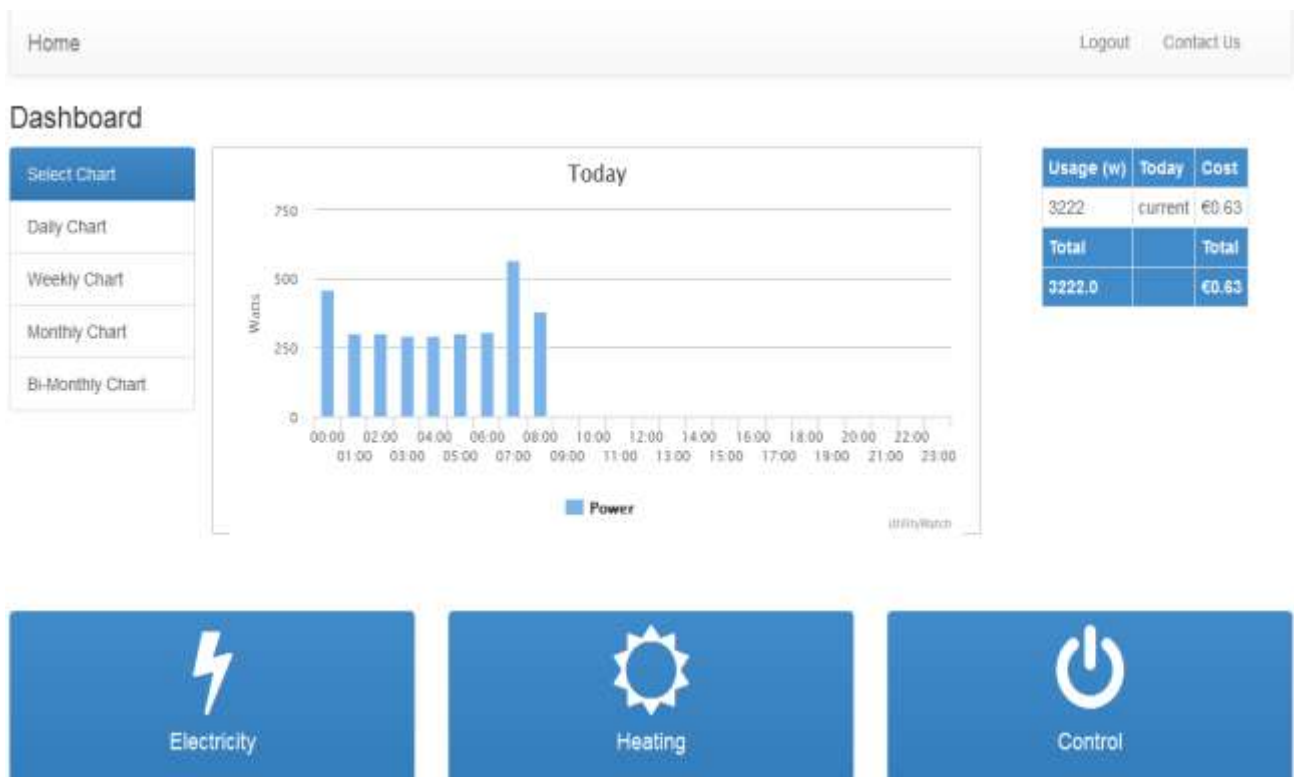


Fig. 4 Day Screen

Similarly to the ‘Day’ period, the weekly, monthly and bi-monthly options render the up-to-date graph along with the informative data table. The only difference is in the table, where the cost is broken down into the chosen segments; i.e. if you select the bi-monthly option, the table displays the cost for each of the weeks and gives a total. I felt this was necessary for the larger time frames to have this extra feature as it added another dimension where the customer could easily compare weekly costs in this example. The weekly timeframe was broken into day slots. Fig. 5 below shows an example of the bi-monthly display.

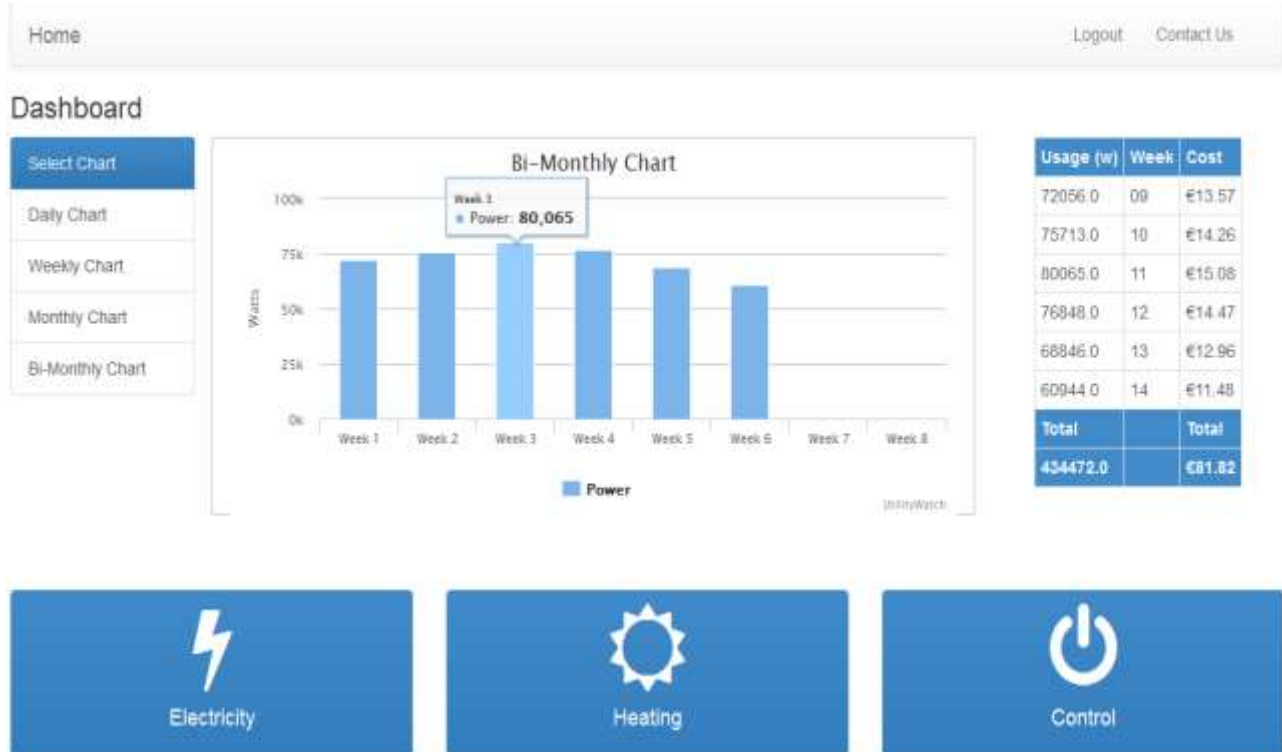


Fig. 5 Bi-Monthly Screen

The graph has some interactive capabilities. By clicking on any of the bars in the chart, a tooltip-like message appears, giving tailored information about the selected bar. This offers a quick reference mechanism to the user, as illustrated for week 3 in Fig. 5 above, instead of having to scan through a table looking for the specific period.

2.3.2. Heating

As mentioned previously in the functional specification document, Utility Watch has the ability to control a customer's heating system. Upon selecting the Heating option, the user is presented with a heating controller panel which contains a master switch and an auto switch. The system can be in **'full off mode'**, **'full on mode'** or **'auto mode'**. Both 'full on' and 'full off' are self explanatory; auto mode is where the user has more control over the system. The user can select a turn on or 'cut-in' temperature and a turn off or 'cut-out' temperature. The state of the switch changes when pressed and the range controller becomes available. A before and after screen shot are shown below to clarify the functionality.



Fig. 6 Heating Screen - Off Mode

The auto switch, if pressed, will change its text from 'Turn On' to 'Turn Off' and also offer a range selector for more efficient control, as shown in Fig. 7 below.



Fig. 7 Heating Screen – Auto mode

The 'Master Switch' will override the 'Automatic' switch. If the master is pressed, the auto cuts out and the range selector is no longer available. The 'full on' mode has a safety temperature of 34 degrees Celsius, where the system will automatically cut out.

2.3.3. Control

The customer has the option to remotely control any electric powered device or system. For this project I chose the lighting system. In the installation manual, a guide sets out the required steps to install the relay into the light circuit. The application triggers a relay which controls the feed to the light. As mentioned, this functionality could be applied to a security gate or entry/exit door. Fig. 8 shows the control screen page which, again like the heating, has a control panel and a gauge. The purpose of the gauge is to show an increase in consumption when the light is turned on.

The system controls the upstairs light circuit by controlling the live feed to that circuit. Once the relay is closed the light switches have control of each individual light. Each light could be controlled individually but they would need an independent relay. It works on the same principle as the main relay controlling the full upstairs circuit.

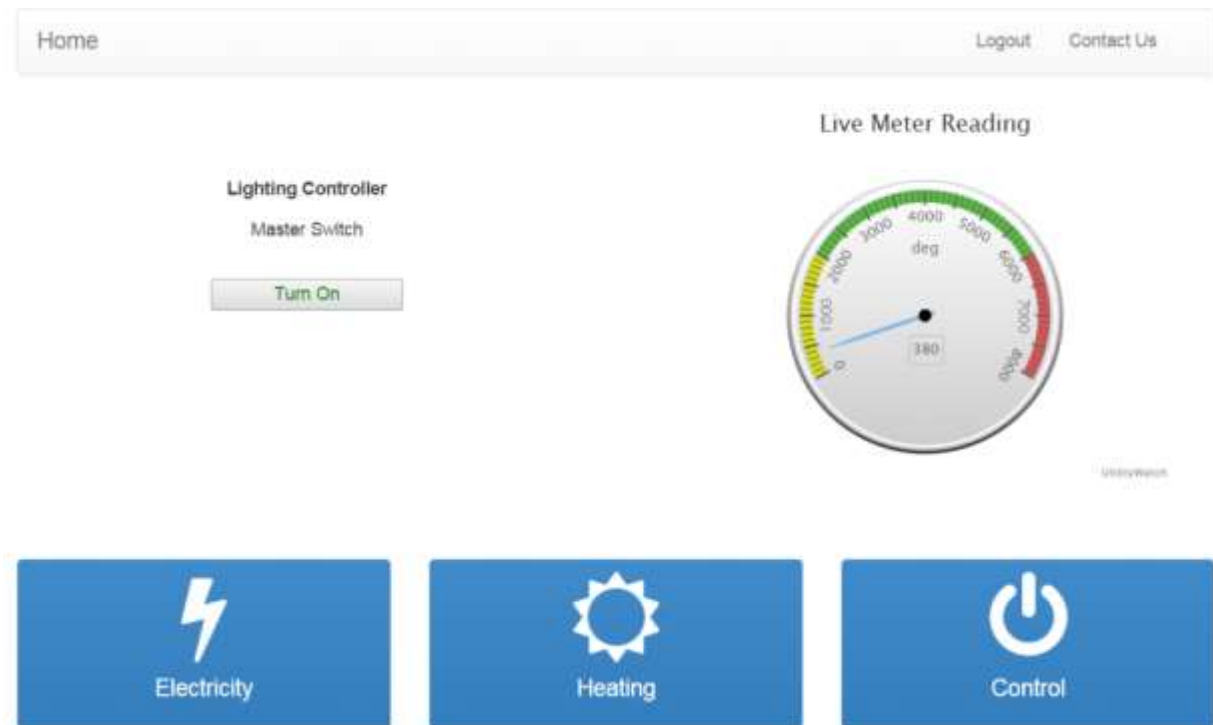


Fig. 8 Lights turned off

Note the difference between the live gauge readings for both different states. The units being used increase by 60 units in Fig. 9 compared to Fig. 8. This indicates a change in consumption due to the lights being turned on.

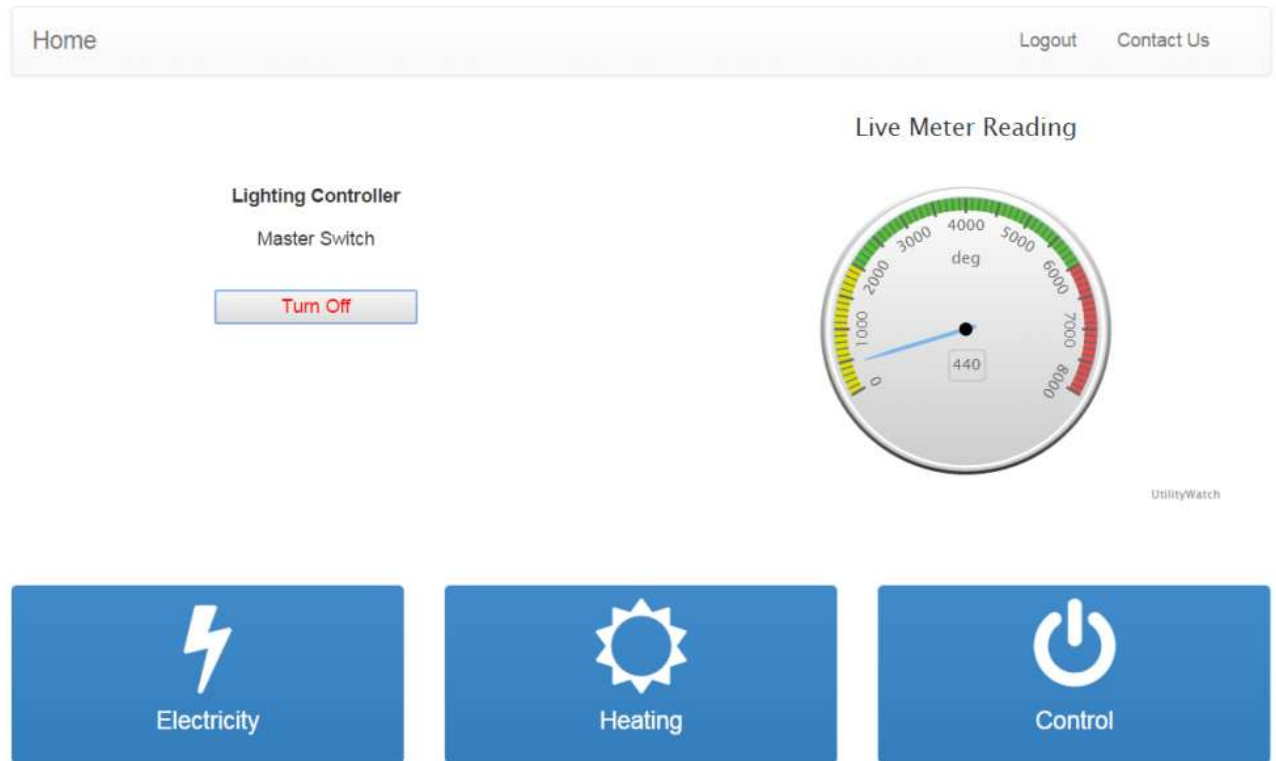


Fig. 9 Lights turned on

The GUI also contains some basic administration functions such as Logout, Contact and Home. These are basic navigation links to required pages and provide basic functionality value to the overall project.

3. Conformance to Specification & Design

In this section of the document, I will compare and contrast the submitted proposal and specifications with the final product. In some cases there may have been changes and I will provide reasons why these changes were necessary. These changes may reflect limitations of hardware or extra features and functionality.

In the initial stages of this project, back in October, all students were required to select a project from a pool of project ideas or submit a project they may be interested in. I proposed my own project idea, which reflected an area I was interested in and wanted to learn about. I tried to incorporate various disciplines into the project which would help me get a broad an experience as possible; which was the purpose of the project. I have provided my initial proposal to assist with the compare and contrast.

3.1. Initial Project Proposal (October '14)

My project will be a responsive web/mobile app that will display, in detail, my home electricity usage. It will display a daily/weekly/monthly graph which will show the peak usage times and also display the cost of the day/week/month.

I would like to demonstrate the change in wattage being used by remotely turning on some electrical appliance in my home, which will be displayed on the graph.

I would like to show that device being turned on as proof by way of an in-house ip camera.

I propose to use the Galileo Board as the brains for communicating all my devices through as a gateway to my home router/modem. My Electricity data will be sent to a remote server on the cloud which will have MongoDB to store all my details, and also host the application. There will be web services also running on the server.

I will need to somehow connect through my public ip address (home router) into the Galileo board in-order to turn on the appliances through some port forwarding method.

Hardware

My project hardware would consist of:

- *Galileo Board with wifi, Ethernet , video, USB capability ... maybe more*
- *energy monitor device with current clamps*
- *RF controlled 3-pin socket*

Platform/Technologies

I propose to design a responsive web app – html/python/flask/css/...

Database – MongoDB

Cloud – not sure exactly yet but may contact the Blacknight for information

3.2. Defining the Scope

My proposal was accepted and my supervisor assigned, and I was ready to begin. My initial meetings with my supervisor were discussions about the scope of the project, explaining the features proposed and setting the initial goals.

One item which was removed from the project was the ip camera functionality. There were a few reasons for this decision. Primarily, the project was aimed at utility monitoring, energy efficiency and automation, so it seemed to be an odd add-in. Also, the feature it was providing, an indication a light had been switched on, seemed a little overkill for the amount of work involved. This would have been a major part of a separate project. Finally, the live feed would have had a major drain on the Galileo Board resources, compromising the performance of the project.

The remaining proposed features were accepted as being challenging enough and worthy of a substantial project, with an open-mind as the project progressed.

3.3. Conformance

Overall, the submitted project has achieved what was proposed. The functionality of the application has been delivered and the end user will recognise a product which was detailed in the functional specification. However, there were changes made in how those functionalities were achieved. This would have been unnoticed by the end user, but the design spec would have been changed to accommodate these alterations.

Where were the changes?

The proposed project mentioned a cloud server hosting the application. Also, the data storage would reside in this location. A decision was made to try and achieve a standalone unit where all features and functionality were incorporated in the one 'box'. We felt this would add value to the product when shipping to customers, similar to a 'plug and play' product. However, this forced a rethink in the technologies I was proposing.

If the Galileo was to act as a web server, flask would be required to be installed. This is a light-weight python web framework and fitted perfectly in with the resource load our Galileo could handle.

The proposed data storage technology was MongoDB; the reasons were simply it was a technology I wanted to learn about, and also the uncertainty of the structure of the data I was going to be handling. Again, conscious of the resource drain, and trying to maximise efficiency of the application, I considered an inherent database system with python and linux, SQLite. There were some drawbacks to this decision which will be discussed later.

That was more or less the changes I had to make. There were technologies added in that I did not anticipate. At the frontend of the application, I had missed the necessity to update certain sections of the GUI. As this application had dynamic content, that had to be handled in a way that did not take from the customer experience. I had to employ AJAX to actively update gauges without refreshing the full page.

4. Learning Outcomes

The main purpose of the project, for me, was to test myself, push the boundaries and to step outside of my comfort zone and see how I would cope, and to learn new technologies. This project would take me into situations and scenarios where I probably haven't been before or not been for a while; staring at a problem and not knowing the solution. These situations teach you a lot about yourself on a personal level and also on a technical level.

4.1. Technical achievements

As I mentioned before, my aim was to keep the project as broad as possible which would, in turn, expose me to the most amount of technologies as possible. I had some technologies that I wished to learn more about, and other technologies which I hadn't used for a while and needed refreshing.

Research

Firstly, as I was starting from complete scratch with a Galileo Board I knew nothing about, I developed my ability to research. The Galileo was documented for certain types of projects, and there was a lot of irrelevant information. I had to decipher what was relevant to my board, the image I wanted to run and then add all the necessary dependencies required to allow my project to develop. Also, with the need for SQLite, AJAX, advanced Networking techniques a 3rd party hardware, all adding to the development of my ability to research.

Knowledge Gain

On a technical level, the project offered a wide range of learning curves. For the backend, my understanding of servers, databases, networking and hardware communication greatly improved. Specifically, with servers I learned a lot about running applications locally and also pushing them remotely. I learned about troubleshooting techniques with server tools, interrogating the data at an instance of time if the application crashed. I learned, very quickly, that a local running application may not behave the same when pushed to the host.

I also gained knowledge in relation to my software development techniques; trying to produce a competent, comprehensive and coherent application, following the MVC model. My web dev skills improved also, as Javascript, html, AJAX, JQuery, Jinja 2 were frequently used on the frontend. At the database level, SQLite rekindled my fondness for sql commands, sad but true. I learned about new tools that helped me test my databases locally and helped produced the sql queries required to provide accurate data.

With every little mistake or large blunder that was made, my knowledge of the system or technology was improved or reinforced. Practical experience is the best way to learn a new skill.

4.2. Personal Achievement

One Task at a Time

The project thought me a lot about myself. There were times when I was facing challenges and got into situations where I was lost, had gone too deep, or was looking at the bigger picture instead of just focusing on one task at a time. The best advice I got was to pick a task and only focus on that task. When I broke the project down into little mini projects, I could think and see clearly again. Sometimes I wasn't looking too far ahead and worrying about things that didn't need to be worried about; yet. When that task was completed, I'd cross it off the list, and at the end of every week the list was reducing.

Character Building

I would class myself as outgoing and probably prefer to work with or lead a team or group. I feel I get along with most people and think I contribute well in such environments. This project was an individual project, and although you could talk to other students or discuss issues with your supervisor, at the end of the day the only way this was being achieved was if I got on

and did it. At the early stages, I found myself thinking about the project day and night, a little bit of self-doubt perhaps. But my confidence grew once I settled in and starting to make progress. Overall I was pleased with my work ethic. Fear of failure is good for focusing the mind.

Project Management

In the early stages I designated a slot every evening to get the project up and running. Once I got into the thick of it, the curiosity got me hooked and I spent more time on it than I probably should have. Although now I am glad I did. Overall, my project planning was surprisingly efficient. I found that when I listed what I wanted to get done and gave it an estimated timeframe, I achieved it most of the time. The timeframe pushes you a little, where as if I didn't estimate it, I would have dragged it out.

Along the same line, telling your supervisor what you plan to have achieved for next week is another way of putting yourself under that little bit of pressure to keep the motivation going; especially when your weekly meeting begins with "Now, what did you say to me last week", as he looks at the meeting record.

5. Project Review

Overall, I was happy with my final project. I feel I put a big effort in and achieved the goals I set for myself and more. The project was fully completed by the final week; a little bit close for comfort but the pressure was worth it. A great sense of relief was felt when I finished it.

What went right?

There were some parts of the project that went better or smoother than others. I was glad not too much time was wasted on figuring out how other vendors hardware worked. I thought I would use more time than I did on this section, but there are some really good networking tools available and for luck, I had been introduced to them in my work experience.

Generating the data for the tables went well also. I like that line of work and it seems that when you like something you tend to be better at it; obviously. I hadn't much experience parsing raw data, but once I got into it, it went well. Again, once you have tools that compliment your work, it makes the task on hand a little easier.

What went wrong?

In relation to the Galileo, I had a lot of problems with dependency issues when I was importing packages into Python. I found there was a difference between Python 2.7 on my local machine and Python 2.7 on the Galileo Linux Board. It seemed to be a bare bones linux OS on the board. This gave me a lot of initial problems when I assumed that as things were working out locally, they would deploy the same on the Galileo. Not the case. I lost a lot of time trying to find relevant packages specific to the Linux image installed and found myself trying many similar packages to see if they were compatible. I had to change my 3d graph package from Matplotlib to HighCharts because of this. The incompatibility issue was by far the most frustrating part of the project.

That aside, nothing major went 'wrong', but I encountered some obstacles. Whatever problems I happened upon, with some discussions with my supervisor, I overcame them. One issue which arose, and I alluded to it earlier, was the decision to use SQLite instead of MongoDB. Upon running some test scripts to assess the performance of SQLite, I discovered that a database can only have one write command at any one time; i.e. single-thread. This limitation would force me to change my database structure as I required many database tables to be reading and writing concurrently as data was being generated and manipulated between tables.

The solution was, in the end, a simple one but it didn't conform to good practice. I generated a new database everywhere I needed a table; so in essence, I ended up with 6 databases each with 1 table, instead of 1 database with 6 tables.

What could have been better?

I was a little disappointed with the graphical user interface. Admittedly, I do not have an interface design background, but I was underwhelmed with the final view. I feel it is important for the user to be impressed with the frontend, as I had put a lot of work into the backend and that never gets recognised by the user. That rings through for all applications, not just Utility Watch.

The GUI does however get the message to the user in a clear and concise manner. It does separate the functionality for illustrative purposes.

Would I take a different approach?

If I was to redo the project I don't think I would change too much. I researched what was required at the time and compared my options. I think, for the most part, I used the best technologies that I could. I got the tasks completed and was happy with the outcome.

I would have liked a couple of extra features to enhance the product. I would like the ability to email my electricity provider with an accurate reading of my meter, providing all my details enabling them to calculate an exact bill as opposed to an estimated one.

I would have liked to have added a feature that automatically configures the network router on-site; maybe in the form of a script that runs on startup. This would prevent a trained or competent individual being required to do this task. It would also reduce the complexity of an installation in the customer's premise.

What advice would I give?

I think the best advice I received was to research well and take one piece of the project at a time and solve it. This really helped me when I got out of my depth and was struggling with the workload. I also would advise anyone to make a list of what you need to achieve, and put a rough estimate of how long they feel it might take. At least if it's on paper you won't forget about it, and also you know you are making progress with every task you cross off; If it is not measured, it is not managed.

Be open-minded. What stood to me was that I was not hard set on a specific technology. I researched what was applicable to do a job, and made an educated decision. But I was allowing myself

the freedom to reverse that decision if I felt I had made a mistake. Sometimes we keep going and put a lot of effort into something, and are unwilling to start again, even though we may be on the wrong path.

I definitely would advise and encourage regular meetings with your supervisor. It will save you much unwanted torment if you keep using your supervisor to bounce your plans or ideas off of. They have seen it all before and may be able to flag a potential issue which you are unaware of. It will also help to clarify things in your own mind. The more you talk about your plans and ideas, the clearer they become.

Finally, keep a running diary of absolutely everything. Any issue encountered, take note; any decision made, write down why you made it. All of these details will be lost otherwise, and when the time comes to produce your report, you will struggle to remember most things. Your main project is a great opportunity to learn a lot about yourself and your abilities. You will be a better developer, student and person after it.

6. Acknowledgements

I would like to thank my fellow students who gave me invaluable feedback about my project over the course of the year. I was surprised with the amount of interest in it and the range of questions that kept coming up.

I would like to thank the 4th year lectures who helped me to improve my presentation skills and also gave constructive criticism at both my sprint demos. I took onboard your questions and comments and applied them to my project.

I would like to thank Joe Kehoe for all the organising and informative sessions he has given us in relation to the project. His time and enthusiasm knows no bounds.

Finally, I would like to specifically thank my supervisor, Paul Barry. From day one, he was extremely helpful and always available if advice was needed, even out of hours. He gave me the confidence to achieve my goal by simply letting me get on with it. He has a wealth of knowledge that students feed off of and I certainly tapped in to it regularly. He was very supportive throughout the process and always finished our meetings on a positive note giving you a fresh enthusiasm; a true gentleman.