

# **eTactica Mobile Final Report**

Guðrún Hauksdóttir  
Jón Valentínusson  
Oddur Aðalgeirsson  
Rúnar Freyr Rúnarsson

**Computer Science B.Sc  
Applied Computing**

**Spring 2012**

## Table of Contents

1.	Introduction.....	3
1.1	About ReMake.....	3
1.2	About the project .....	3
1.2.1	Hardware .....	3
1.2.2	Software .....	4
1.2.3	eTactica Mobile .....	5
1.3	Product vision.....	5
2.	Project Plan.....	6
2.1	Agile methods.....	6
2.2	Attendance .....	7
2.3	Workplace .....	7
2.4	Sprints.....	7
2.5	Work Log.....	8
2.6	Meeting Logs .....	8
2.7	Risk Analysis.....	8
3.	Technical Design .....	8
3.1	Premise.....	8
3.2	System overview .....	9
3.3	User Interface .....	11
3.4	Test Driven Development.....	13
3.4.1	JUnit & Selenium .....	13
3.4.2	JUnit .....	14
3.4.3	PyUnit & Flask-Testing.....	14
4.	Progress Overview.....	15
4.1	Sprint 0 - Van Damme (16/01/2012 - 01/02/2012).....	15
4.2	Sprint 1 - Lundgren (01/02/2012 - 15/02/2012) .....	15
4.3	Sprint 2 - Schwarzenegger (15/02/2012 - 29/02/2012) .....	16
4.4	Sprint 3 - Seagal (29/02/2012 - 14/03/2012) .....	16
4.5	Sprint 4 - Jet Li (14/03/2012 - 28/03/2012) .....	17
4.6	Sprint 5 - Willis (28/03/2012 - 13/04/2012).....	18
4.7	Sprint 6 - Stallone (13/04/2012 - 20/04/2012) .....	18
4.8	Sprint 7 - Weaver (20/04/2012 - 27/04/2012).....	19

4.9	Sprint 8 - Norris (27/04/2012 - 03/05/2012).....	20
4.10	Sprint 9 - Eastwood (03/05/2012 - 10/05/2012).....	20
5.	Future Vision .....	21
6.	ReMake's comments .....	21
7.	Conclusion .....	22

## 1. Introduction

This project, eTactica Mobile, is a final project at Reykjavik University in collaboration with ReMake Electric. This report covers the progress of the project throughout the semester. Additional reports and documents referenced here can be found on the attached CD.

The students working on this project are Guðrún Hauksdóttir, Jón Valentínusson, Oddur Aðalgeirsson and Rúnar Freyr Rúnarsson. The project instructor is Stefán Freyr Stefánsson. Our contacts at ReMake Electric are Axel Gunnlaugsson (acting as Product Owner), Bjarki Guðlaugsson (Lead Developer) and Guðmundur Þorri Jóhannesson (Software Engineer & DBA).

The project started on January 12th and the final hand-in is on May 11th 2012.

### 1.1 About ReMake

ReMake Electric is an award-winning startup company in the field of energy management. The idea behind the company is to provide businesses and homes with a way to monitor their electricity consumption in order to optimize usage and minimize cost. ReMake is a rapidly growing company with a customer base both in Iceland as well as Europe.

### 1.2 About the project

ReMake's system has two main components: The hardware that measures the consumption of electricity and the software, known as eTactica, that displays those measurements in a meaningful way to the end-user.

#### 1.2.1 Hardware

ReMake's hardware are physical components that are attached to a building's electrical installation. This hardware is installed in a building's electrical cabinets and is comprised of four main parts:

- **Circuit Breaker Meters (CBMs)** - A CBM is a small click-on device that attaches on top of a branch in an electrical cabinet and reads the load on that branch directly. The CBM has an LED indicator for that branch's current load, going from green (no load) to red (full load).
- **SmartGauge** - A SmartGauge is similar to a CBM, except that it reads data from the mains of a whole cabinet directly, instead of only a single branch.
- **SmartBar** - A SmartBar is a long horizontal strip that is plugged on top of a row of CBMs and is responsible for collecting data from those CBMs and sending it on to a SmartGate.
- **SmartGate** - A SmartGate is a small internet-connected device running on embedded Linux. The SmartGate's role is to collect data from all bars/meters/gauges and send it on to ReMake's database over the internet.



Image 1: Circuit Breaker Meter

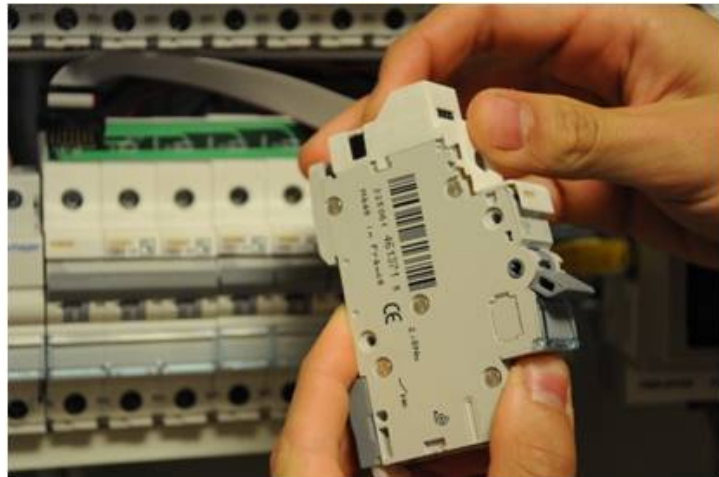


Image 2: A closer look at a Circuit Breaker Meter



Image 3: A typical hardware installation. A row of CBMs with a SmartBar on top (partly visible) and a SmartGate to the right.

### 1.2.2 Software

ReMake's software, eTactica, is an online information dashboard that displays the information collected from the hardware in a visual and meaningful way to the user. eTactica gives you the option of monitoring your entire installation down to a single branch, see the exact cost of electricity and an estimation for next month's bill as well as the carbon footprint for your installation, among other things. Various graphs and visualizations are readily available to help establish a quick and easy understanding of energy consumption and which parts of the user's system are costing him the most.

Alerts are a big part of eTactica as well. Alerts are pre-defined triggers that notify the user whenever irregular or unexpected activity occurs. For example, you might want to be notified when a branch in your building is at a very high load and at risk of tripping or when the branch that your freezer is on shows no load at all.



Image 4: The desktop version of eTactica.

### 1.2.3 eTactica Mobile

Our project involves the design and implementation of a mobile version of the company's eTactica web portal, formally known as eTactica Mobile. It is a web application for smartphones, displaying a compact version of a dashboard, similar to the one on eTactica, containing information about business or household electricity consumption. The cost, usage and carbon footprint information is available as well as the functionality of an alert inbox.

The role of eTactica Mobile is to act as a lightweight portal to the information contained on eTactica with an emphasis on being able to receive alerts on the go for those who are responsible for maintenance and monitoring of electrical installations. The goal was to make eTactica Mobile able to present information at a glance with a touch-friendly interface and a clear, minimalist presentation of useful information.

## 1.3 Product vision

This is the group's vision of the product, that is, what it does, what it is capable of and where its strengths lie.

*For business and home owners*

*who need to manage their energy consumption*

*eTactica Mobile is a web based electricity consumption dashboard*

*that supplies real time information about the use and cost of electricity.*

*Unlike other electricity monitoring solutions eTactica Mobile provides measurements down to branch level and custom alerts designed to reduce costs and prevent damage caused by abnormal usage.*

## 2. Project Plan

### 2.1 Agile methods

The team decided to employ agile methods by working the Scrum way. This means that the project's requirements were written down into a product backlog in the form of user stories and the project period divided into several one/two week sprints.

Scrum was well suited to the team as we were almost always working at the same time and place so we could easily coordinate who was doing what and divide the team down on the tasks best suited to each member's strengths. In addition, ReMake's software development team works using similar methods so staying in sync with them would help us build the application on top of their most recent services and requirements.

The team's Scrum roles were as follows:

Product Owner	Axel Gunnlaugsson (ReMake)
Scrum Master	Jón Valentínusson
Team	Guðrún Hauksdóttir Jón Valentínusson Oddur Aðalgeirsson Rúnar Freyr Rúnarsson

Table 1: Scrum roles

The team decided that Jón would be the Scrum Master since he would have the most consistent attendance and work hours of any member of the team. The other three would be interrupted by projects, final exams and so on throughout the semester.

At the start of each sprint, stories were pulled from the backlog and arranged by priority into the current sprint in collaboration with ReMake's product owner. Those stories were then broken down into smaller tasks required to complete the story. The stories themselves were then estimated in the form of story points using planning poker. Story points are an arbitrary unit of time on the following scale:

0 - ½ - 1 - 2 - 3 - 5 - 8 - 13 - 20 - 40 - 100

Each member of the team held a hand of cards with those numbers and after deliberation, all members laid down what they thought would be the required amount of story points for a specific tasks at the same time. If there was a large gap between any two members' estimations, the story was discussed further and either settled on or re-voted. Each task was also estimated in work hours, roughly corresponding to the amount of story points.

Each day, the team held a five minute stand-up meeting where the members stated their current task, what they were doing yesterday and whether anything was hindering them. Frequent meetings and Q&As with ReMake's employees served to keep us on the same page and make sure that their requirements were being met as work on the project went on.

Sprints were closed off with a sprint retrospective where the team discussed what went well and what didn't, what we should stop doing or start doing next sprint and by writing a short summary of what happened that sprint.

## 2.2 Attendance

A schedule was set up for when the team members would be attending ReMake. This schedule is for the first 12 weeks of the semester, while courses are still on-going. The last 3 weeks of the semester will be entirely focused on the project.

	Mon	Wed	Fri	Hours
Guðrún	10.30 – 17.00	10.30 – 17.00	09.00 – 17.00	21
Jón	08.30 – 17.00	13.00 – 17.30	13.00 – 17.00	17
Oddur	08.30 – 17.00	08.30 – 17.00	12.30 – 17.00	21
Rúnar	08.30 – 17.00	08.30 – 17.00	12.30 – 17.00	21

Table 2: Attendance plan

## 2.3 Workplace

The team was provided a workplace at ReMake's offices in Hlíðarsmári from January 16th which was used throughout this project. This was the team's main location to work from and a majority of the work hours were spent there.

## 2.4 Sprints

These are the sprints as organized by the team. Note that this is an up-to-date version of the planned hours and sprints which is slightly different from the one originally proposed at the start of the project. The main difference lies in adjustment of planned hours around the exam period and re-evaluation of planned hours of the later sprints.

Additionally, Weaver is a sprint which was added in. Stallone was shortened down to a week long sprint instead of two as originally planned and Weaver added to fill in the gap. This was done since the group was done with exams and was working full-time on the project so we wanted more rapid iterations to occur. ReMake was also implementing services crucial to the project at the time in week-long sprints, so we wanted to stay in sync with them in that regard.

Sprint name	Date	Length in days	Planned hours	Planned hours per member
Van Damme	16.01 – 01.02	16	200	50
Lundgren	01.02 – 15.02	14	160	40
Schwarzenegger	15.02 – 29.02	14	160	40
Seagal	29.02 – 14.03	14	160	40
Jet Li	14.03 – 28.03	14	100	25
Willis	28.03 – 13.04	16	100	25
Stallone	13.04 – 20.04	7	80	20
Weaver	20.04 – 27.04	7	180	45
Norris	27.04 – 03.05	7	220	55
Eastwood	03.05 – 10.05	7	210	52.5
<b>Total</b>		<b>123</b>	<b>1570</b>	<b>392.5</b>

Table 3: Final sprint plan



## 2.5 Work Log

Each member of the team kept his/her own work log of hours in attendance and what was done each day. To summarize, each member wrote down where he was working, how long he worked and a short description of what the team member was working on. This helped the team track the hours each member spent on the project as well as tracking down roughly which member worked on what. See the attached work log for the full version.

## 2.6 Meeting Logs

Most meetings, both with the project instructor and ReMake's employees were logged to give the team an oversight of the decisions made and remember pointers and helpful comments received. Please refer to the attached report for the full meeting log.

## 2.7 Risk Analysis

At the start of the project the team analyzed potential risks affecting its progress. The purpose of the risk analysis was not only to analyze risks but to have a backup plan ready for each risk so that the group would not get delayed in case these risks occurred.

The main risks included some outdated technology by ReMake as well as the fact that ReMake is a startup company with things constantly changing meaning the group had to be quick on their feet with changes on their way. Methods like Test Driven Development which the group had not used in a larger scale project were also something the group considered risky.

The full list of risks along with the team's evaluation of how serious they are and how likely they are to occur can be found in the attached Risk Analysis document.

## 3. Technical Design

This chapter aims to explain, shortly, the design premise for the project as well as design decisions and tools the team chose to use. A more detailed report along with a list of the various frameworks used for the application can be found in the attached Design Report.

### 3.1 Premise

eTactica Mobile is built as a mobile web application using HTML5/CSS3 and Javascript, running on a Play! server. The reason for designing a website rather than an app was to ensure compatibility with a broad range of mobile devices as well as to shorten the required development cycle within the limited time frame of the project.

The team also put emphasis on making the web application feel like a native app with regards to the look and feel of the UI as well as the general flow. This was accomplished by using mobile-friendly frameworks such as jQuery Mobile and using single-page navigation which wouldn't require full-page refreshing.

Due to the team's limited access to a wide variety of different devices, the application was mostly tested on the following Android phones:

- 2x Samsung Galaxy S2 on ICS 4.0.3 and Gingerbread 2.3.5 (480x800 px. resolution)
- Samsung Galaxy Ace on Gingerbread 2.3.6 (320x480 px. resolution)

- Samsung Galaxy Y on Gingerbread 2.3.6 (240x320 px. resolution)

### 3.2 System overview

A typical installation of ReMake's hardware and its relationship with the eTactica software could look something like this.

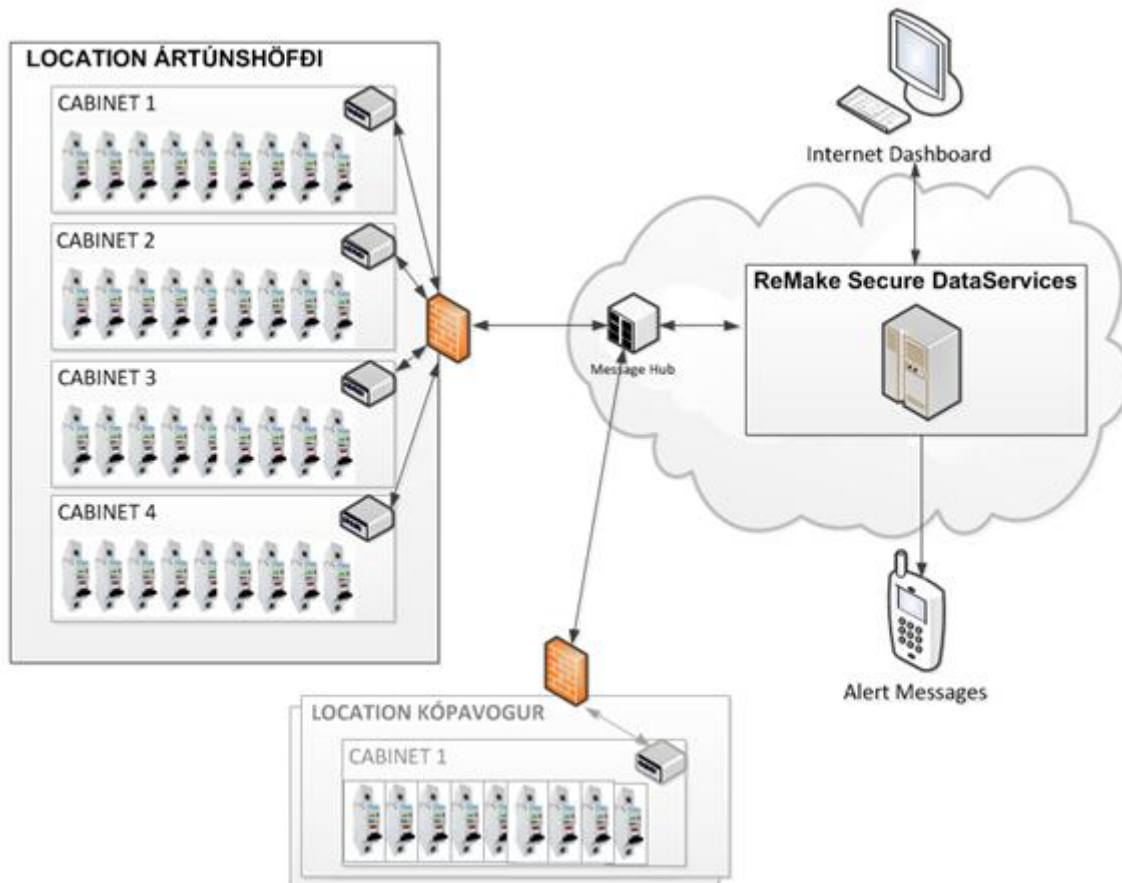


Image 5: An overview of a typical installation of ReMake's software and hardware.

Electricity usage data is collected from the click-on branch meters in each cabinet and sent on to ReMake's database, where it is then displayed in an informative way on both the desktop and mobile versions of eTactica.

A closer look at the system and the flow of data from a click-on branch meter to the mobile or desktop version of eTactica is represented in the following way:

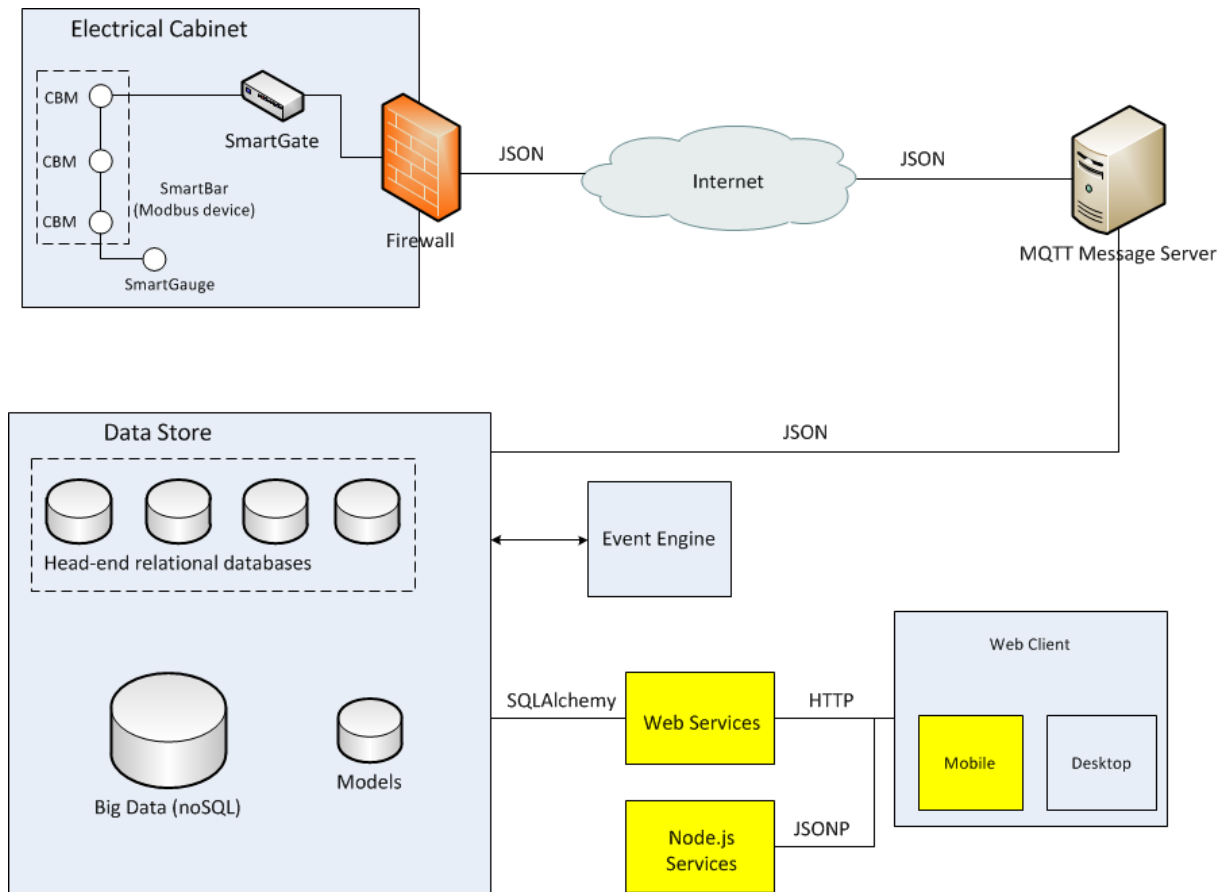


Image 6: A more technical overview of ReMake’s system.

This overview represents the flow of data from a single electrical cabinet. CBM is short for Circuit Breaker Meter, which is another name for the click-on branch meters previously discussed. With the aid of complementary hardware (the SmartBar and SmartGate), data is sent over internet to a standard message queue server.

From there, it is sent on to ReMake’s database. The head-end databases receive the data as it arrives and periodically send it down to the “Big Data” database which is a central store for data spanning all the head-end databases. All interfacing with the database is done through web services which can retrieve data in various forms, such as usage for a single location over an entire month or the current load in amperes on a single branch at a specific time.

The boxes highlighted in yellow are the parts of the system that the team worked on. The Node.js services were a back-up for ReMake’s web services while they were unavailable or in development.

A closer look at the “Mobile” box:

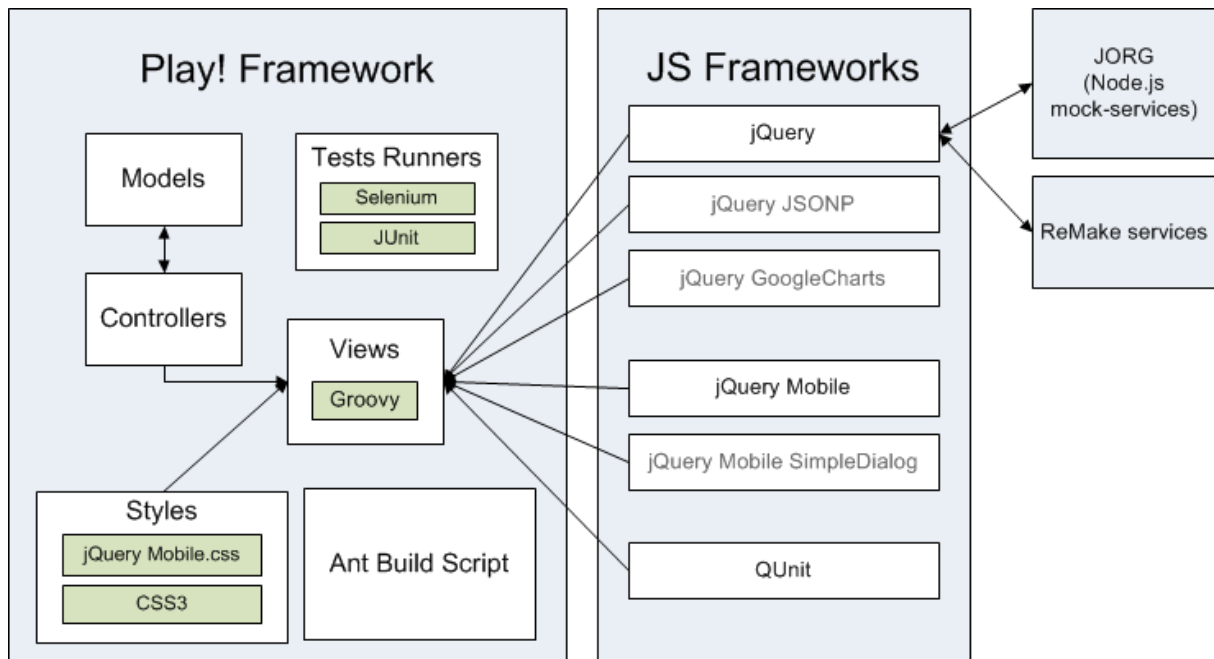


Image 7: An overview of the frameworks and tools used in eTactica Mobile.

This image represents the infrastructure of eTactica Mobile. The Play! MVC framework acts as a server runner with a built-in tests runner for UI and Java unit tests. The HTML views are constructed with the Groovy template language and enhanced with Javascript, most notably jQuery and jQuery Mobile.

Javascript takes on the largest part of the front-end workload, including page transitions and fetching data from the web services. QUnit is a unit testing framework for Javascript, which was needed since a large majority of our code is in Javascript. jQuery is also responsible for requesting data from the services and displaying them in the UI.

### 3.3 User Interface

This chapter is dedicated to the eTactica Mobile's user interface (UI) and we will discuss the reasoning behind our UI decisions as well as explain the flow of eTactica Mobile.

The first screen is that of the login screen. The user can then input a security key which can be found in the desktop version of eTactica. While there he can find two other methods of logging in:

1. Being sent the direct link to the website via e-mail.
2. Scan in the QR code on the website using a mobile phone.

The reason for why it is possible to access your site simply via a link is because the link has a so called access token in it which is used to redirect the user to the correct page.



Image 8: eTactica Mobile login screen

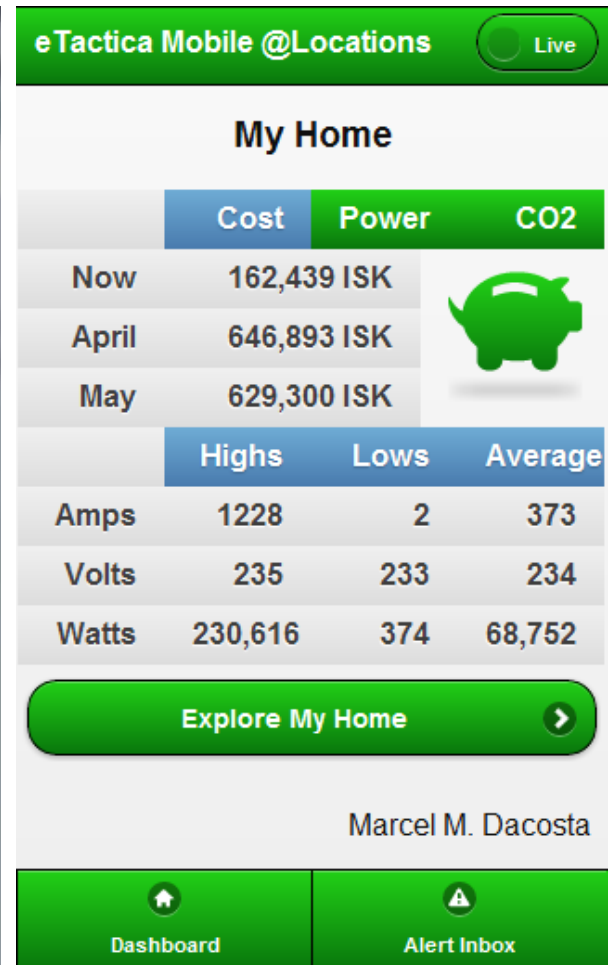


Image 9: eTactica Mobile main dashboard

In every window after this there will be a header and a footer, the header displays the title and on what page you are currently on and the footer allows you navigate to the main page, the dashboard, or the alert inbox at any time.

In the first tab the user can see a list of his locations. These locations can vary from being a simple home to a large business building. On this site the user can see information regarding his electricity usage as well the cost involved and other various aspects concerning electricity. The user can click on a button in the title to turn on/off live data globally meaning that the values will update every 10 seconds or so without the need to refresh. All of these locations have a button to drill down, to see more details about that particular location.

After drilling down you get a similar looking interface detailing the same electrical information as before but here you can see the breakdown of the location into more detailed parts. So if a location is split into two parts you can see those parts here. Similar to the locations, you can drill down further into branch level as well as go back to the locations tab.

In the branch level you can see a list of branches belonging to the model (for example, an electrical cabinet) that was drilled down to. Each branch displays a meter showing how much load there currently is on the branch, which is identical to the LEDs found on ReMake’s hardware. Additionally, it displays the load percentage in numbers. There is no further drill-down here but you can go back to the model tab by clicking the back button.

Navigating to the alert inbox will display a list of alerts and some details about them like when they were received, the label of the alert etc. Clicking on an alert will display a graph. This graph details the load on the branch a few minutes before and after the alert was triggered to give users insight into what really happened and allow them to base their response on that information.

### 3.4 Test Driven Development

One of the decisions the group made at the start of the project was to employ test-driven development. This means that almost every testable piece of code or function had its unit- and UI tests written before being written itself. For those tests, the team used various different tools for automating the tests.

#### 3.4.1 JUnit & Selenium

JUnit is a testing framework which is capable of testing code written in Java. We used JUnit for the small portion of the application which was written in Java, which mostly had to do with authentication.

Selenium is a tool used for browser automation. That means that we can make Selenium imitate a user using the system by doing tasks such as clicking links, filling in forms, opening pages and so on. Alongside this automation, we can run tests to see if the system is behaving as expected by checking for the presence of certain elements in the DOM, among other things.

Both JUnit and Selenium tests are run in Play!'s automated tests runner.

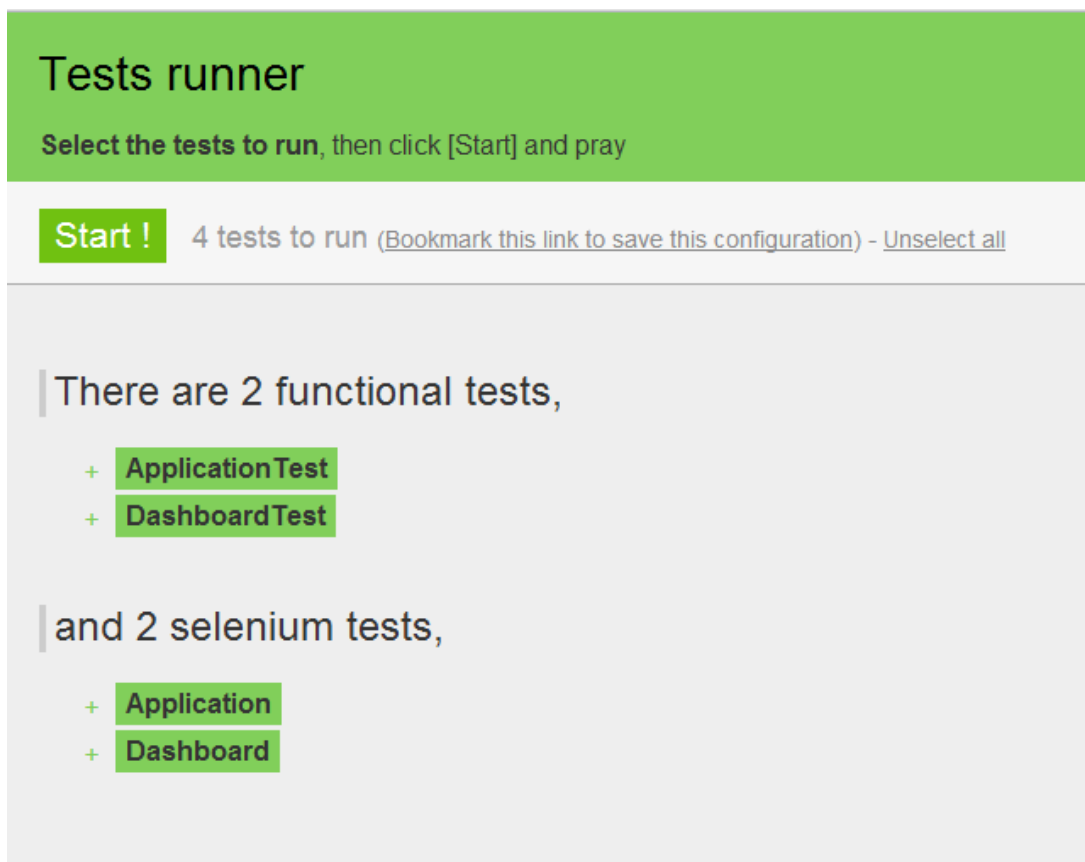


Image 10: An example image of the Play! test runner

### 3.4.2 QUnit

QUnit is a framework, similar to JUnit for unit testing Javascript code. It is written in maintained by the jQuery team, hence the “Q” in “QUnit”. QUnit also has its own tests runner in the form of an HTML file enhanced by Javascript, similar to JUnit and Selenium.

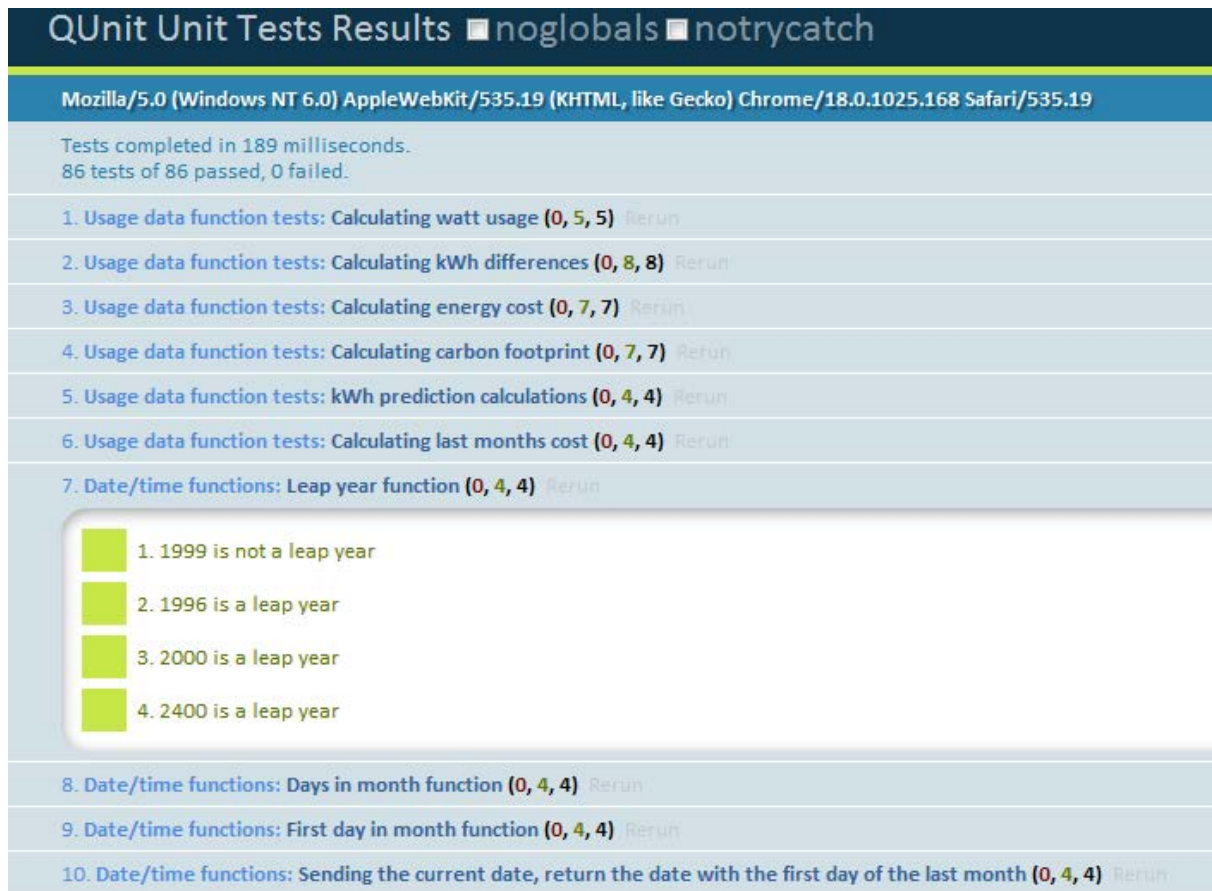


Image 11: An example of results from the Qunit tester

### 3.4.3 PyUnit & Flask-Testing

Web service unit test were run on PyUnit and implemented using Flask-Testing modules.

The tests are run in shell from project root by typing:

- `$ python alert_service_test.py`

PyUnit is the official Python unit testing framework and is basically the Python version of JUnit, which it is based on. Flask-Testing is a Python extension with good support for testing with the SQLAlchemy database wrapper. Although PyUnit support writing test cases the corresponding Flask-Testing module is better suited for the task especially when it comes to mocking database behavior and session for SQLAlchemy.

All ReMake server side unit tests are run the PyUnit test runner but the actual test cases are implemented using the TestCase module from Flask-Testing.

## 4. Progress Overview

The progress overview is a quick look at each sprint, what was planned and accomplished. A more in-depth description and review of each sprint can be found in the attached Progress Overview document. The product backlog and the full sprints along with their user stories and tasks can be seen in the Product Backlog.

### 4.1 Sprint 0 - Van Damme (16/01/2012 - 01/02/2012)

Quote: "I'm okay. I'm just half dead"

Our goals for this sprint were to start planning the project which included work hours, work location, meetings, creating a requirements list as well as finishing reports the school needed. We created prototypes for what the system could potentially look like. We also tried to look into the programming language and what possible frameworks we could use for the project.

Discussions about Test Driven Development (TDD) and Continuous Integration (CI) arose and we debated whether we should implement them into our project. Overall it was mostly discussions and general planning of the project and no programming.

### 4.2 Sprint 1 - Lundgren (01/02/2012 - 15/02/2012)

Quote: "I have the power!!!"

The goals here were to create an almost complete product backlog as well as having sprint 1 fully planned and sprint 2 planned roughly. We discussed TDD and CI was still being debated at the start of the sprint. There were more reports to work on as well, including the risk analysis and other various documentation that wasn't finished in sprint 0.

The results were that we decided to use TDD and picked up 3 Javascript frameworks: Sammy.js, Knockout.js and jQuery/jQuery Mobile as suggested by ReMake. Because of the changes in ReMake with new staff amongst other things there was not much programming done. However, this was a possibility that the group had predicted early on.

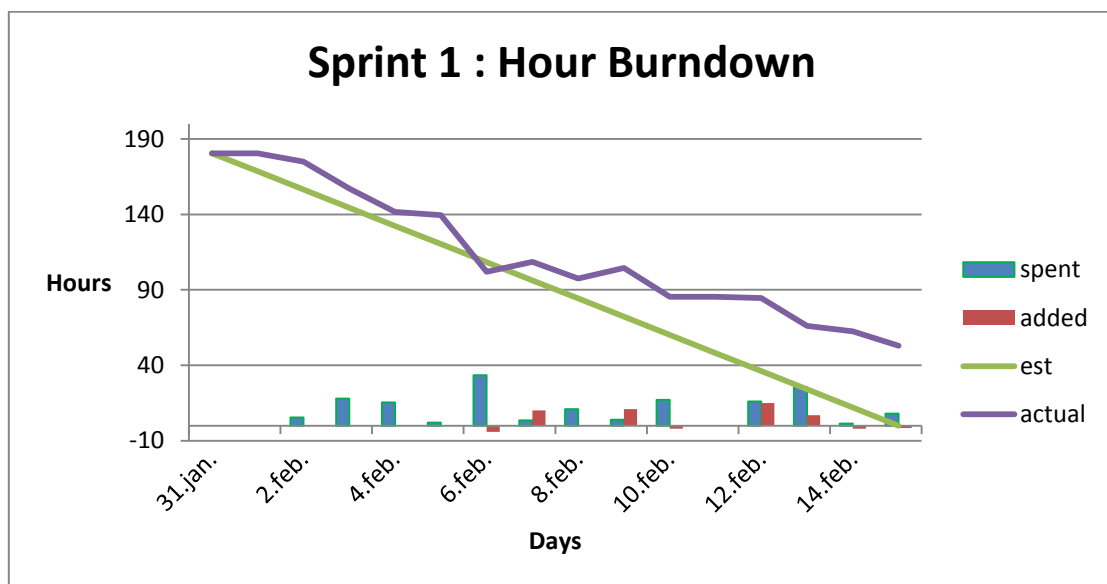


Image 12: Sprint 1 Hour Burndown



### 4.3 Sprint 2 - Schwarzenegger (15/02/2012 - 29/02/2012)

Quote: "Get to the chopper!!!"

Our goals here were to put programming into full swing, which included login and a very simple User Interface for the main page with some simple data being displayed.

After this sprint and sprint 1 not as much had been completed as we had planned and hoped for. The reasons were mostly the fact that ReMake was making a lot of changes in their underlying system architecture as well as us needing to get comfortable with Knockout and jQuery. In the end there were only a few tasks completed in this sprint.

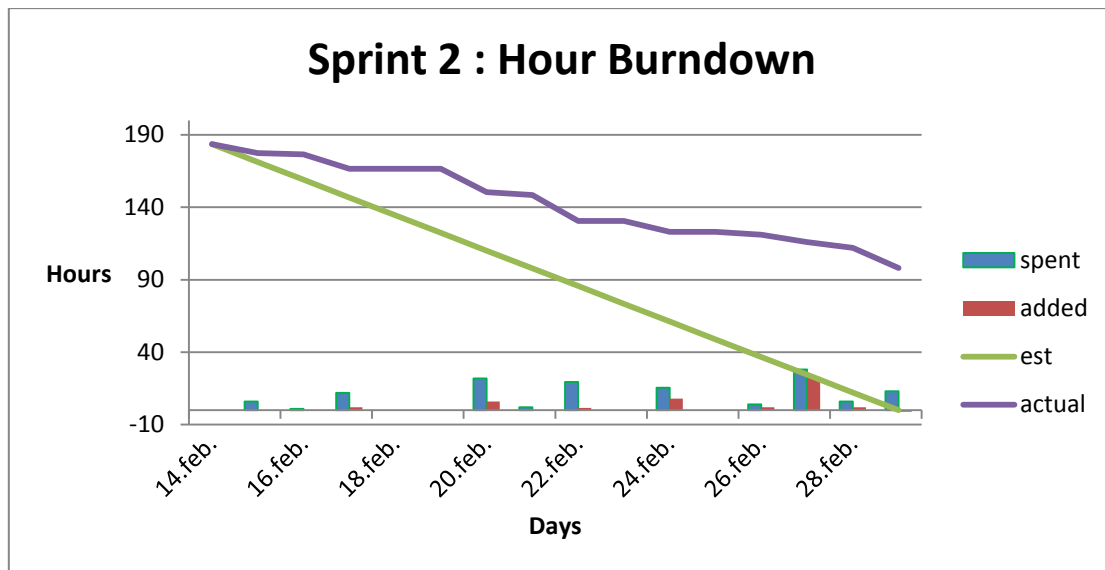


Image 13: Sprint 2 Hour Burndown

### 4.4 Sprint 3 - Seagal (29/02/2012 - 14/03/2012)

Quote: "It's not a job, it's an adventure"

For this sprint we had similar goals as we did in the last sprint. ReMake decided to change the way of logging in from the usual username/password into OpenID (See: <http://openid.net/>). We felt our progress was being hindered somewhat by the fact that ReMake was not fully ready with their data services so we decided to mock our own data so we could keep on working.

With similar goals in mind as the last sprint we managed to complete several of the tasks we had set for us. We set up a mock data server which we managed to get active at the very end due to cross-origin problems we ran into. In this sprint we found out that one of our methods of routing was being handled by both Sammy.js and jQuery mobile which resulted in conflicts between the two frameworks. We ended up dropping Sammy.js because it was mostly being used only for routing but jQuery mobile had the same function along with good UI features we wanted to use to our advantage.

With us using TDD and the group realized that most of the coding was happening in Javascript we started looking into methods/frameworks to help us with unit tests in Javascript, eventually picking up QUnit.

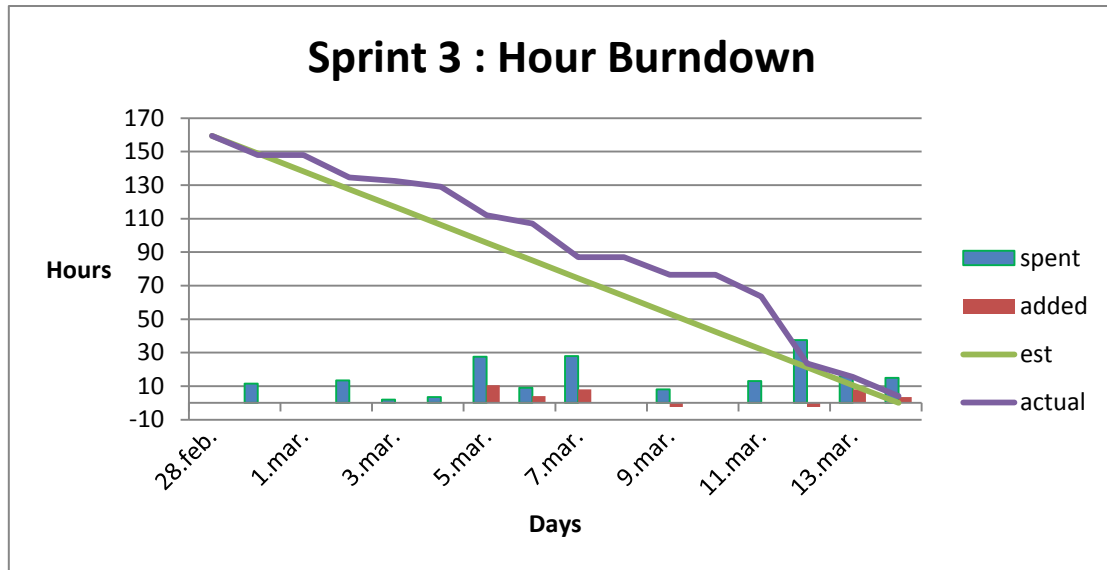


Image 14: Sprint 3 Hour Burndown

### 4.5 Sprint 4 - Jet Li (14/03/2012 - 28/03/2012)

Quote: "Your lucky ring stinks"

For sprint 4 we had yet again somewhat similar goals of mocking data and creating the basic flow of the site with that mocked data as well as finishing the login with OpenID. A part of this sprint also went into preparing for status meeting 2.

This sprint went very well, after having been somewhat stuck the first few sprints we managed to get the mocked data running smoothly and started to display various mocked data on the website as well as being able to login.

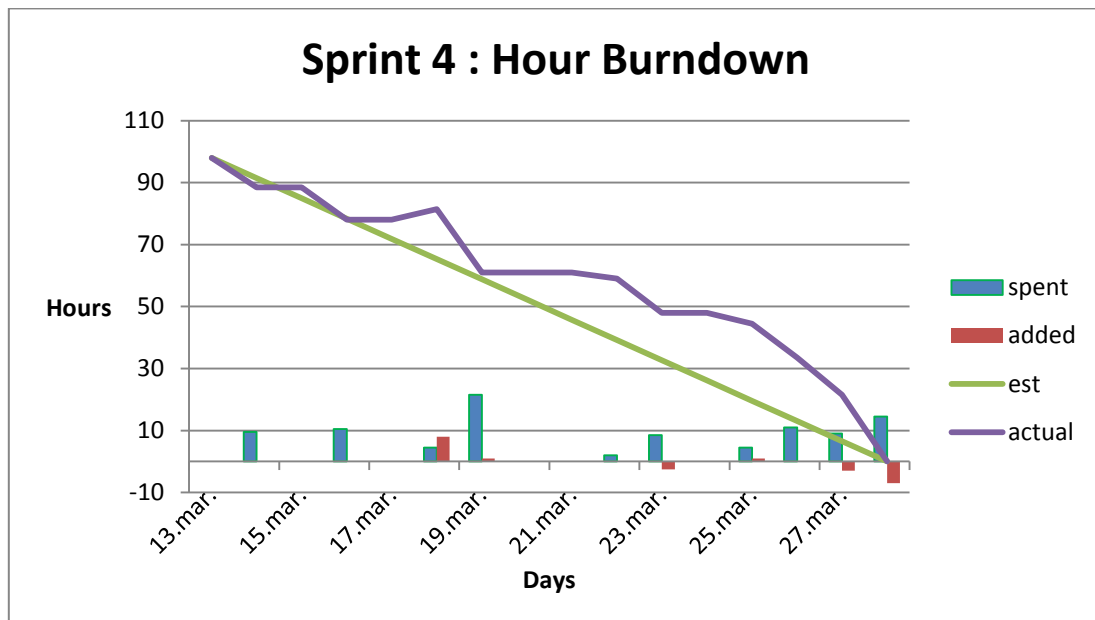


Image 15: Sprint 4 Hour Burndown

### 4.6 Sprint 5 - Willis (28/03/2012 - 13/04/2012)

Quote: "Yippee-ki-yay mother\*\*\*\*er"

After having successfully managed to display the basic information on our website, this sprint's goals focused on getting started on the alert system with mocked data. ReMake decided again to change the way of logging in from OpenID into a security key where each user has a unique access token.

The sprint as a whole went well and we got the alert system working with mocked data along with some more detailed information on the dashboard. We also decided to change the sprint times a bit, to suit our needs with regards to ReMake and the exam weeks that followed this sprint.

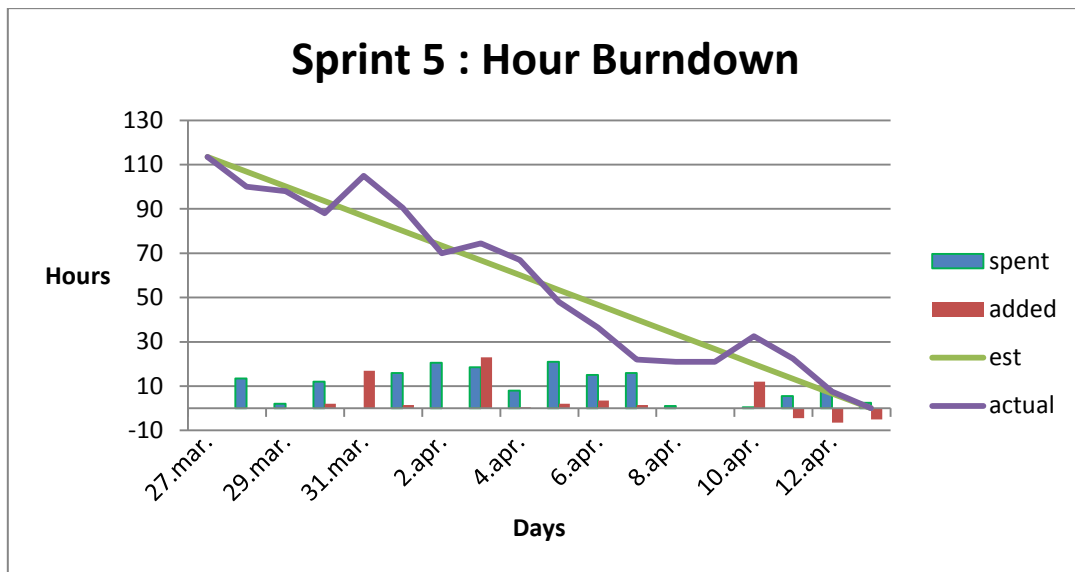


Image 16: Sprint 5 Hour Burndown

### 4.7 Sprint 6 - Stallone (13/04/2012 - 20/04/2012)

Quote: "AAAADRIAAAN!!!"

The team in collaboration with ReMake decided to write an alert service which ReMake would use as well as ourselves. Alongside the alert service we improved upon our own alert system in eTactica mobile. This sprint also included creating a QR code for each user to scan with their mobile phones for logging in if they so desired.

What we managed to complete in this sprint was creating a mocked version of the eTactica website which we inserted our QR code system as well as creating charts for our alerts showing behavior before and after the alert was triggered, giving the user a sense of what exactly happened.

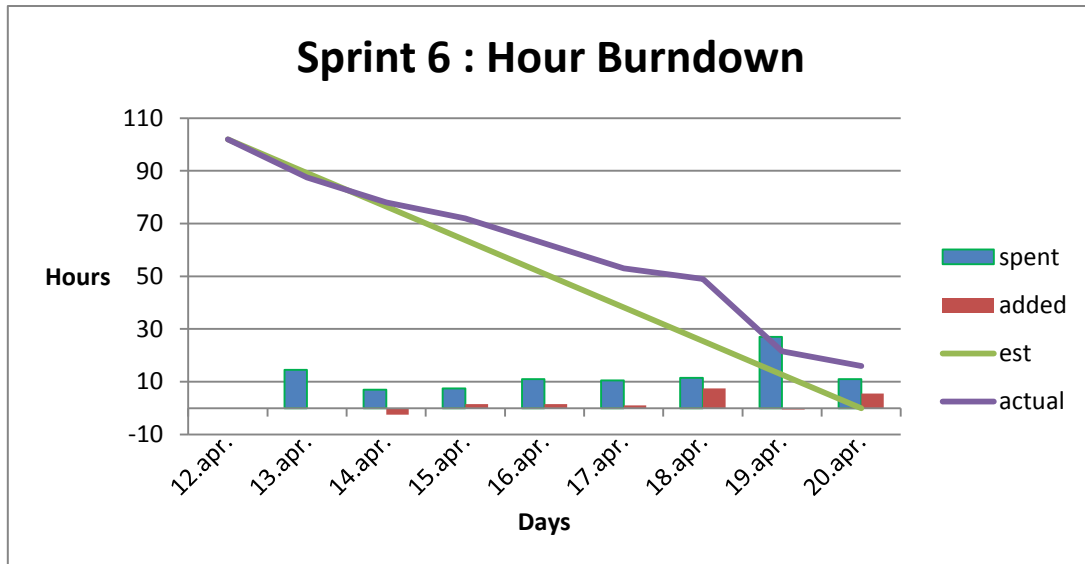


Image 17: Sprint 6 Hour Burndown

### 4.8 Sprint 7 - Weaver (20/04/2012 - 27/04/2012)

Quote: "Get away from her, you BITCH!"

Here we focused less on adding functionality and more on making the UI smoother and more compact as well as working with actual live data from ReMake databases. Alongside this we included bug fixing as a backup plan for this sprint since ReMake was working on their live data connections this sprint.

The group had started working full time here after the exams and we managed to get things rolling very well. The team connected the application to live data as well as fixing a couple of recurring bugs. The User Interface looks a lot better now as it had previously been taking up way too much space but looks much more compact now.

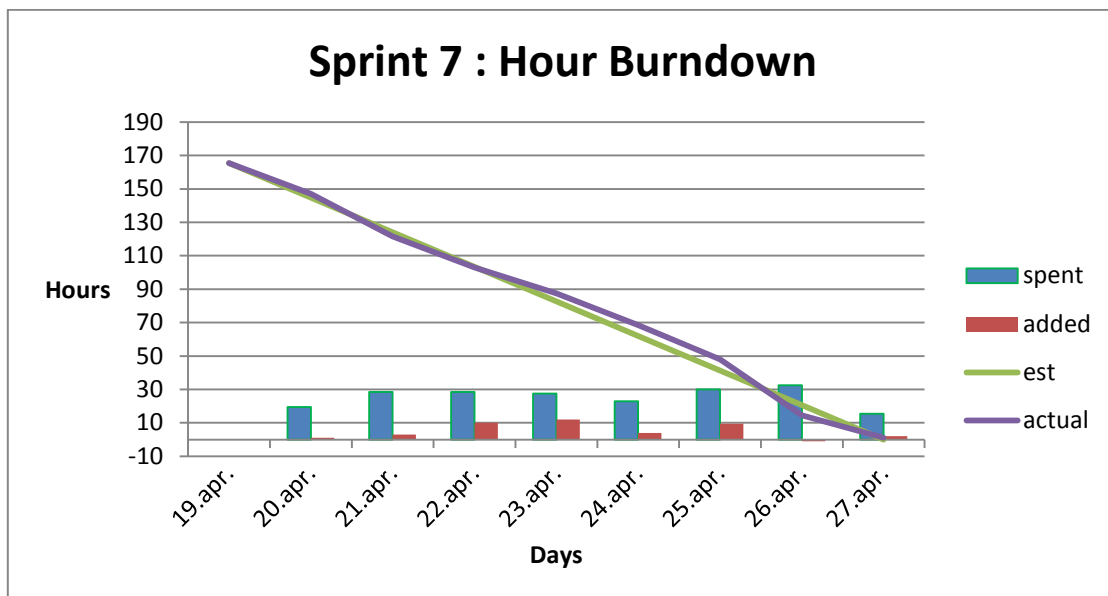


Image 18: Sprint 7 Hour Burndown

### 4.9 Sprint 8 - Norris (27/04/2012 - 03/05/2012)

Quote: "It's a go, TAKE THEM DOWN!"

Sprint 8 focused on putting the final touches on the underlying data services as well as polishing and ironing out any remaining bugs. A lot of time went into documentation and preparation for the final hand-in and the third status meeting. A lot of co-operation with ReMake's software team took place in this sprint to make sure that everything was working as needed. In conclusion, the sprint went well and was closed a day early since every story had been completed.

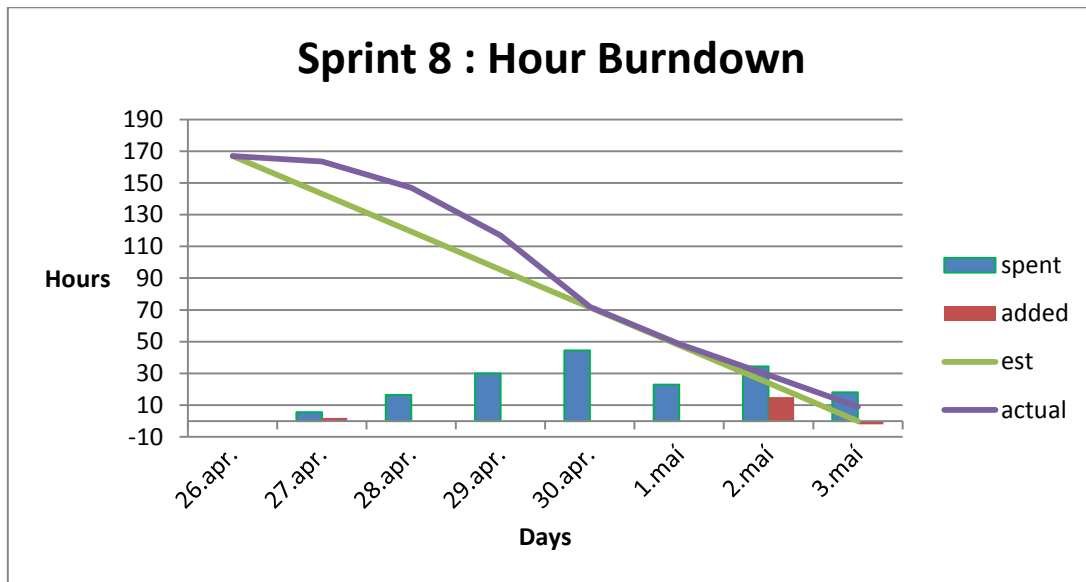


Image 19: Sprint 8 Hour Burndown

### 4.10 Sprint 9 - Eastwood (03/05/2012 - 10/05/2012)

Quote: "Go ahead, make my day, punk!"

We decided after status meeting 3 to take in two new stories: Being able to drill down to branch level as well as changing the live functionality and look. Most of the group focused on doing reports for the final turn in which was our main focus. Some were working on making the branch drill down possible as well as changing the live functionality.

The results turned out well and we managed to finish the drill down to branches as well as changing the live functionality quite quickly. All reports were finalized on the 10th of May and everything went quite smoothly.

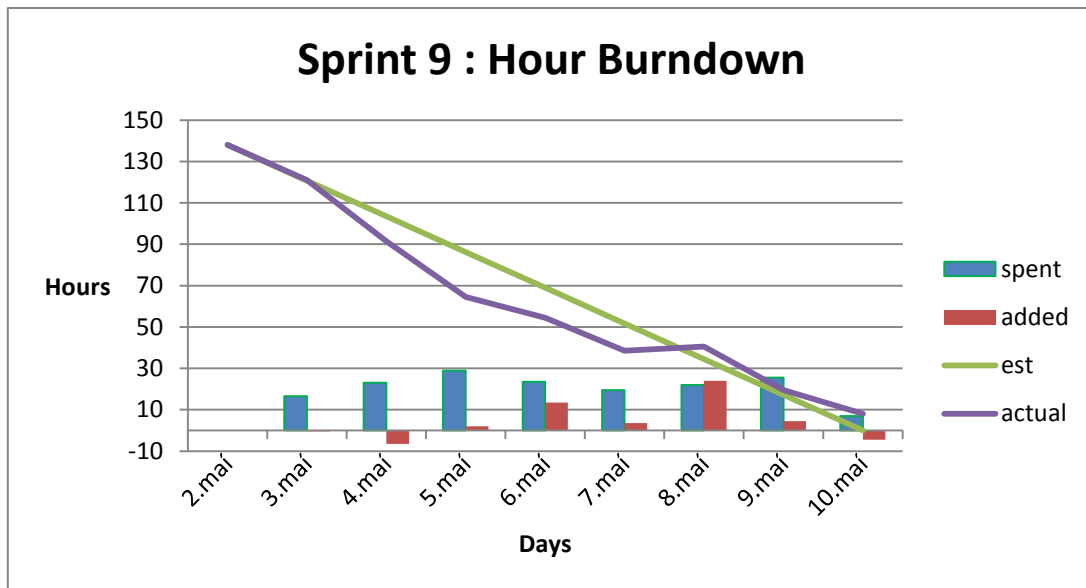


Image 10: Sprint 9 Hour Burndown

## 5. Future Vision

For the future of eTactica Mobile, we envision being able to view your consumption information in a more visual and customizable way. For example, being able to choose the desired period for which you're viewing data as well as the option of having graphs and other visualizations of that information.

Possible additions to the system as it is now could include being able to see an alert history of a particular branch as well as a way to go directly to the branch page from the alert inbox and inspect a particular branch, perhaps recently after an alert was triggered.

Another possible idea which was not implemented would be to create a version of eTactica Mobile for a "technician" role. This version which would have less data for the regular user and more technical data which could act as a central location for setup and maintenance information for ReMake's systems.

As for the technical side, the biggest problem is the fact that we originally established on a Play! framework but as the project went on, more and more of the code was moved over to Javascript. As it stands now, the Java code left is minimal and could easily be replicated elsewhere. As such, Play! could be substituted for a much more lightweight platform such as an Apache server or similar.

## 6. ReMake's comments

"We took on a four man team from RU to design and develop a mobile application for our eTactica energy measurement solution.

In the beginning the team was faced with several challenges like understanding electricity and the metrics of energy consumption.

On top of that the team had to tackle (b)leading-edge approaches to development and an ever changing resources and approach landscape at Remake-Electric inc.

In the beginning things evolved slowly but surely. Then in the latter stages of the project, development took off and a BETA version of eTactica Mobile was demoed beyond expectation within ReMake management.

Good team, good architecture, good delivery.”

*Axel Gunnlaugsson*  
*ReMake CIO*

## **7. Conclusion**

All in all, the team was very satisfied with the end product and the project as a whole. We feel that we managed to accomplish a large majority of what we set out to do in terms of system requirements. We have learned a lot, through guidance, trial and error about the Scrum methods and the general process and development cycle involved in making software.

The team clicked well and there were no disputes worth mentioning other than a natural difference of opinions and the discussion involved in coming to a conclusion in those situations. We'd like to thank ReMake as a whole for providing us with this excellent project and the opportunity to work with a promising startup company.