Angular web-application based on Excel calculation tool

Bachelor's thesis

Häme University of Applied Sciences

Riihimäki, Fall 2016

Tommi Koponen

HAMK
HÄMEEN AMMATTIKORKEAKOULU
HÄME UNIVERSITY OF APPLIED SCIENCES

ABSTRACT

Tietotekniikan koulutusohjelma
Riihimäki

Author Tommi Koponen **Year** 2016

Title Angular web-sovellus Excel laskentatyökalusta

TIIVISTELMÄ

Tämän opinnäytetyön tarkoituksena oli muuntaa olemassa oleva Excel laskentatyökalu web-sovellukseksi. Työn pää tavoitteena oli tehdä helposti ylläpidettävä ja jatkokehitettävä web-sovellus, joka laskee energiatehokkuusluvun rakennuksille yhtä tarkasti kuin Excel-työkalu. Opinnäytetyön tilasi Suomen ympäristöopisto Sykli.

Valitsin tämän aiheen, koska olin keskittänyt opiskeluni web-sovelluksien tekemiseen ja ajattelin että Excel-laskentatyökalun muuntaminen web-sovellukseksi olisi sopiva minun taidoilleni ja antaisi minulle syventävää kokemusta web-sovellusten kehittämisestä.

Tässä opinnäytetyössä käymme läpi mikä on energiatehokkuusluku ja mikä sen tarkoitus on, ja selitämme Syklin EEnavi-projektista ja sen tavoitteista. Käymme läpi sovelluksessa käytetyt työkalut, frameworkit ja kirjastot, joita käytettiin sovelluksessa ja miksi käytimme niitä. Käymme lisäksi läpi frameworkit, kirjastot ja työkalut, joita emme käyttäneet mutta olisimme voineet, ja miksi niitä ei käytetty. Selitämme myös sovelluksen arkkitetehtuurin ja toimintaperiaatteet. Lopussa on myös ohjeet miten sovellusta voi helposti jatkokehittää tarpeiden mukaan.

Avainsanat web-sovellus, angular, energiatehokkuus, Excel, sovelluskehitys

Sivut 142 sivua, joista liitteitä 113 sivua

HÄMEEN AMMATTIKORKEAKOULU
HÄME UNIVERSITY OF APPLIED SCIENCES

Degree Programme in Information Technology
Riihimäki

Author Tommi Koponen **Year** 2016

Subject Angular web-application based on Excel calculation tool

ABSTRACT

The purpose of this thesis was to convert as existing Excel calculation tool into a web application. The main goal was to make an easily maintainable and easy to develop further, web application that calculates energy efficiency number for buildings as accurately as the Excel tool. This thesis was requested by Environmental School of Finland Sykli.

I chose the topic because I had focused my studies on making web applications and I thought that porting a calculation focused Excel file into a web application would be suitable for my skills and would give me a lot more in-depth experience with development of web applications.

In this thesis, we go through what Energy efficiency is and its purpose. We explain a bit about Syklis EEnavi-project. We go through all the tools, frameworks and libraries used in the project and why we used them. We also go through the frameworks and tools which we could have used but decided against and why we didn't choose them. We also explain the architecture of the software and how it works. At the end, there are also instructions on how to further develop the program.

Keywords web application, angular, energy efficiency, Excel, software development

Pages 142 pages including appendices 113 pages

CONTENTS

Appendices

1 INTRODUCTION

Previously Environmental School of Finland Sykli had an Excel based tool to calculate buildings actual energy consumption. This Excel tool is used to calculate the energy efficiency of buildings and show how certain fixes affect the energy efficiency rating of the building.

The Excel tool is under continuous development and it is important that all the users are using the newest version of the tool. It was impossible to know which version of the program the users were using.

Sykli needed their Excel tool to be made into a web application to ensure that the users had the latest version of the program.

My job was to write the web application of based on the Excel tool. The goal of the web application was to be easily maintainable so that it would be easy to update all the formulas and develop new features.

2 ENERGY PERFORMANCE CERTIFICATES

Energy performance certificates show how energy efficient a building is and it may even contain some suggestions from professionals for improving the energy efficiency of the building. The energy performance certificate may show the actual energy use of the past year if available, it will always show the calculated energy use for the building and how much energy needs to be purchased. (Laki rakennuksen energiatodistuksesta 2013/50 § 9.). The energy efficiency number is calculated by dividing the calculated energy use by the total area (m^2) of the building (Energiatodistus – Mikä on energiatodistus).

2.1 Benefits

Energy performance certificates help in comparing the energy efficiency of different types of buildings. It encourages use of renewable energy in buildings. (Laki rakennuksen energiatodistuksesta 2013/50 § 1.). They help house buyers to see how energy efficient a building is before buying it, which helps them calculate the cost of energy needed to maintain habitable conditions in the building.

2.2 **Curation**

The energy performance certificates are curated heavily and there are certain criteria an individual must meet before they can issue energy performance certificates. They need to have an applicable degree on the field and need to pass an exam for energy performance certificate. After that, they are added to a list of registered professionals who can write official energy performance certificates. They also need to have proper tools for calculating the energy efficiency of buildings. (Laki rakennuksen energiatodistuksesta 2013/50 § 12.).

2.3 **Calculation tools**

A building that doesn't have cooling or has cooling only in some parts can use a month based calculation method. Any other building must use a dynamic calculation tool. The tool used for calculating the energy efficiency of the building must follow the D5 building regulations calculation methods. Dynamic calculation tool must be validated by EN (European Standard), CIBSE (Chartered Institution of Building Services Engineers) or ASHRAE (American Society of Heating, Refrigerating and Air-Conditioning Engineers) standardized test cases.
(Rakennusten energiatodistus ja sen E-luvun laskenta, Vuolle, s. 24).

3 **EENAVI PROJECT**

EEnavi is Syklis project to provide energy efficient renovation solutions by trained and neutral experts. EEnavi project aims to provide property owners and professionals a way to find suitable solutions that are economic, energy efficient, and increase the indoor comfort the building. The professionals will provide the customer multiple renovation solutions and their costs, which take into consideration the special features of the property and its structures. (Sykli - EEnavi).

3.1 **For companies**

EEnavi training is aimed for energy assessors and building inspectors or people with equivalent education. After the training the attendees will be able to do a comprehensive energy renovation charting that is based on the target buildings base data and an evaluation made on the grounds. The customer is given a report that will introduce them with at least three energy efficient renovation solutions. This service is the first step in serving customers with realistic and reasonable options in energy renovation. (Sykli – EEnavi yrityksille).

The EEnavi training will provide the attendee with improved knowledge in energy efficiency renovations and buildings lifecycles, access to the EEnavi calculator, which is designed to easily simulate different types of renovation plans for improving the energy efficiency of the building, and support from the other members in the EEnavi network. (Sykli – EEnavi yrityksille).

Picture 1 Sykli's illustration of their EEnavi training

3.2 For housing associations and small house owners

Trained EEnavi professionals will provide the energy renovation charting report and renovation plans to the small house owners and housing associations. They will also be given the cost of all the renovation options and how fast they will pay themselves back through saved energy. (Sykli – EEnavi pientaloasukkaat ja taloyhtiöt).

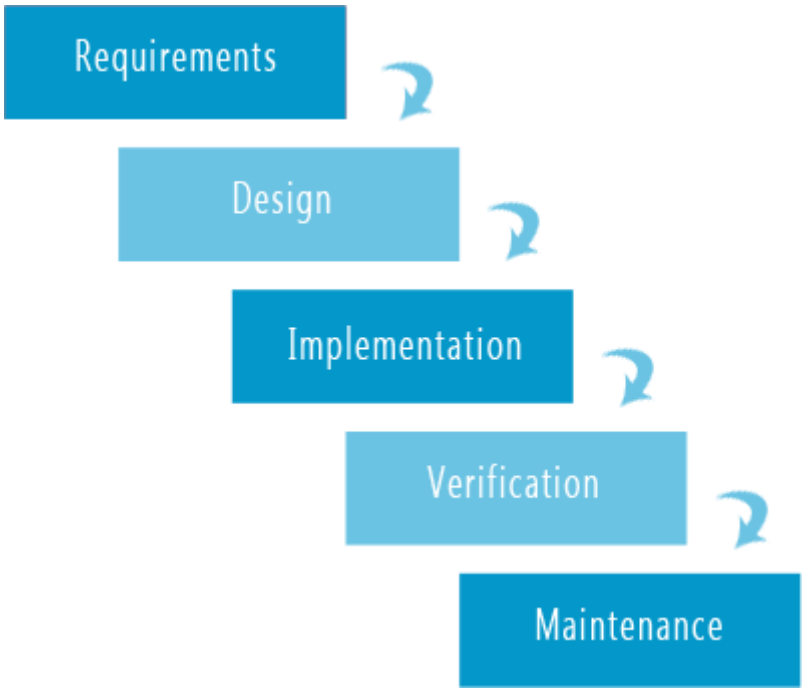
3.3 The calculation tool

The calculation tool is not used in official energy performance certificate calculations; rather it's used for calculating the energy efficiency number and to compare different renovation options effect on it. The energy efficiency number is comparable to the one in the official energy performance certificate. The main goal of the tool is to help visualize how

much certain renovation options improve the energy efficiency of the building and when the renovations pay themselves back through saved energy. The calculator takes into account more variables than official energy performance calculation tools, which need to follow strict rules for the calculations, and is able to give much more accurate numbers for the building. (Sykli – EEnavi pientaloasukkaat ja taloyhtiöt).

4 DEVELOPMENT

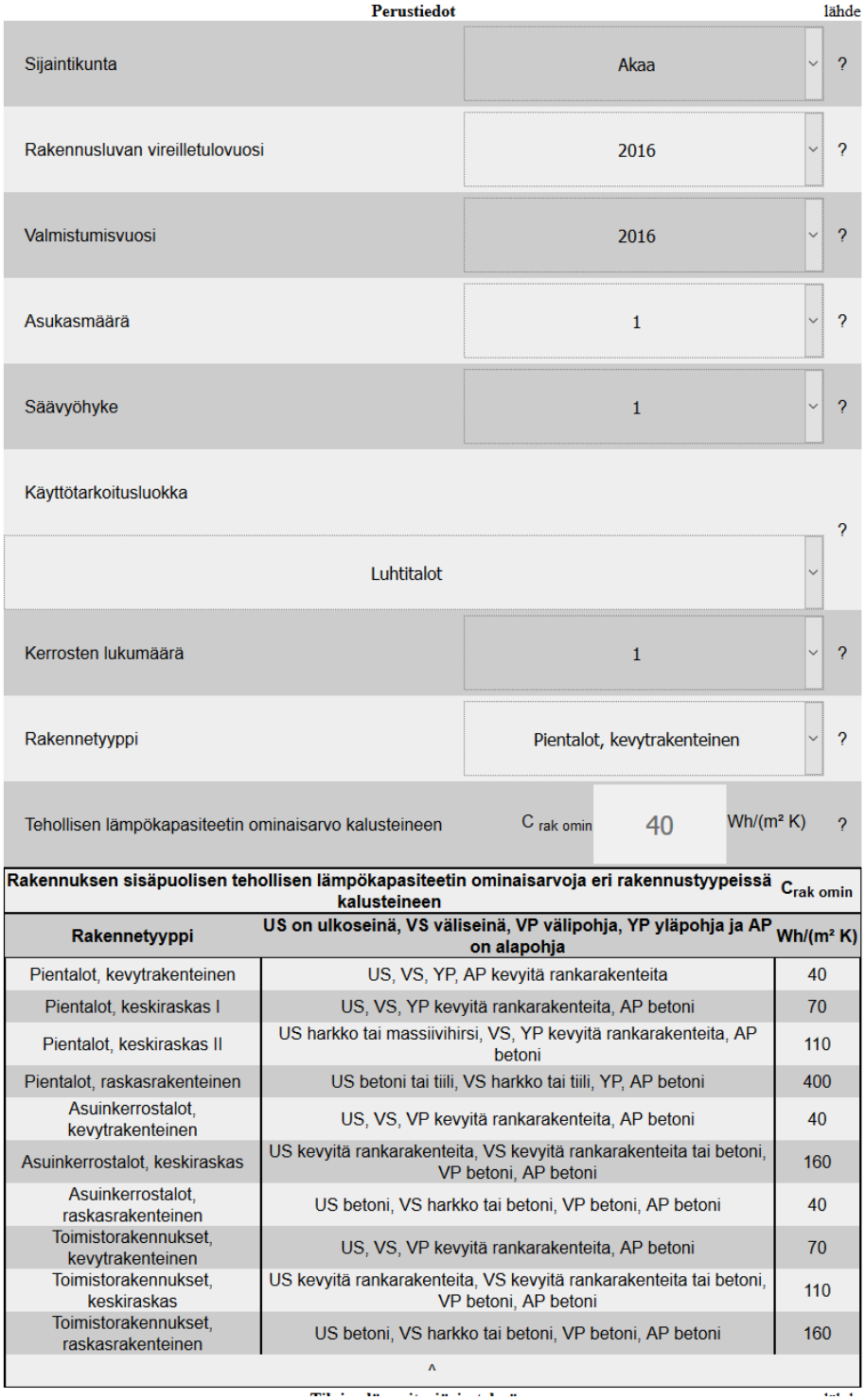
The development phase of the program consisted of initial planning phase, choosing frameworks suitable for the project, verifying requirements for the project, designing the software architecture, designing layout and look, and implementation. The development model that I loosely followed for this project was waterfall model. In waterfall model you go straightforward with the different phases of the development cycle. (Oagile 2014).



Picture 2 The waterfall model philosophy

4.1 Planning phase

The planning phase for this project consisted of looking through the provided Excel tool and having meetings with Sykli to realize their needs for the product. I also came up with a quick prototype/proof of concept, which I demonstrated to them and talked with them about its design.



Perustiedot			lähde
Sijaintikunta	Akaa		?
Rakennusluvan vireilletulovuosi	2016		?
Valmistumisvuosi	2016		?
Asukasmäärä	1		?
Säävyöhyke	1		?
Käyttötarkoitusluokka	Luhtitalot		?
Kerrosten lukumäärä	1		?
Rakennetyyppi	Pientalot, kevytrakenteinen		?
Tehollisen lämpökapasiteetin ominaisarvo kalusteineen	$C_{rak\ omin}$ 40	Wh/(m² K)	?

Rakennuksen sisäpuolisen tehollisen lämpökapasiteetin ominaisarvoja eri rakennustyypeissä kalusteineen		$C_{rak\ omin}$
Rakennetyyppi	**US on ulkoseinä, VS väliseinä, VP välipohja, YP yläpohja ja AP on alapohja**	**Wh/(m² K)**
Pientalot, kevytrakenteinen	US, VS, YP, AP kevyitä rankarakenteita	40
Pientalot, keskiraskas I	US, VS, YP kevyitä rankarakenteita, AP betoni	70
Pientalot, keskiraskas II	US harkko tai massiivihirsi, VS, YP kevyitä rankarakenteita, AP betoni	110
Pientalot, raskasrakenteinen	US betoni tai tiili, VS harkko tai tiili, YP, AP betoni	400
Asuinkerrostalot, kevytrakenteinen	US, VS, VP kevyitä rankarakenteita, AP betoni	40
Asuinkerrostalot, keskiraskas	US kevyitä rankarakenteita, VS kevyitä rankarakenteita tai betoni, VP betoni, AP betoni	160
Asuinkerrostalot, raskasrakenteinen	US betoni, VS harkko tai betoni, VP betoni, AP betoni	40
Toimistorakennukset, kevytrakenteinen	US, VS, VP kevyitä rankarakenteita, AP betoni	70
Toimistorakennukset, keskiraskas	US kevyitä rankarakenteita, VS kevyitä rankarakenteita tai betoni, VP betoni, AP betoni	110
Toimistorakennukset, raskasrakenteinen	US betoni, VS harkko tai betoni, VP betoni, AP betoni	160

^

Picture 3 Initial design prototype

4.2 Used frameworks, and tools

The program was written entirely with JavaScript. JavaScript is one of the most used programming languages used for web development.

4.2.1 Visual Studio Code

Visual Studio Code is my chosen text-editor for writing front-end and JavaScript code. It provides many useful features that make writing and debugging JavaScript code easy. It has a good built-in IntelliSense support that works especially well with node programs but unfortunately does not work properly with AngularJS and AMD (asynchronous module definition) with RequireJS. With Chrome debugger extension, you can attach the debugger to your running app on Chrome and debug with breakpoints directly on the text-editor. This provides a fast way to debug and fix certain errors in the code. The integrated terminal console makes it easy to run certain commands straight from the editor, such as compiling less files with lessc or running my CSV to JSON program, without having to open an external console. The built-in snippet support makes it easy to write a snippet and use it with the defined prefix while coding, it also supports setting tab-points and variables on the snippet to write some repetitive parts quickly. (Visual Studio Code).

Picture 4 Example snippet that I used to quickly add the defined directive code template for this project

The $n denote the variables on the snippet, for example here the $1 is used to mark the name of the directive and it's written on all the required places.

4.2.2 AngularJS

AngularJS is one of the best mvc (model view controller) frameworks for making single page applications. It's made and is being developed by Google. AngularJS offers great tools for creating dynamic htmls (hypertext markup language). It lets you create different logic, which is neatly separated from each other for each view thanks to Angulars directives and controllers. It has great data-binding support that lets you handle data between the view and controllers easily. With AngularJS, there is no need for extensive DOM (Document Object Model) manipulation, as AngularJS handles all that under the hood. (AngularJS).

4.2.3 Angular ui-router

Angular ui-router is an AngularJS module that helps with managing different states of the application, in other words the different view pages on the single page app. It provides an easy to use state provider that allows easy navigation between the different views on the app, allows for easy attaching of different controllers for each view, and allows easy modularization of templates. (AngularJS – ui router).

4.2.4 RequireJS

RequireJS is a file loader for JavaScript that simplifies and speeds up file loading. It helps with managing vast amount of different JavaScript files needed when building Angular apps. Its Text and JSON (JavaScript Object Notation) add-ons are a great help when loading a lot of JSON files. Its lazy loading makes the client download only all the files which are required as dependencies from the server instead of downloading everything at once. (RequireJS).

4.2.5 Git

Git is a version control tool. It's almost mandatory to have some kind of version control tool when there are multiple developers. Git is also useful when programming on multiple places. It's good to setup a base repository on your home computer then clone that repository to where ever you need it. This also provides multiple backups of the code. It's also great for when you have to do extensive refactoring on the code or adding new features which might break the code. You can just create a new branch where you can refactor the code etc. and then merge it back to the main branch when it works, or continue from the old branch if everything breaks without having to worry about manually reverting everything or maintaining multiple copies of the software. (Git).

4.2.6 ESLint

ESLint is a style checker for JavaScript. It's used to maintain consistent styling rules across the code. It helps with finding errors quickly as you code and to notice simple mistakes. It does static analysis on your code base and reports any conflict against the defined rules. There is a good extension for it on Visual Studio Code which runs ESLint as you type code and immediately reports any errors. This way it's really quick and easy to stay on top of any style mistakes you make from the start and helps in maintaining cleaner code. (ESLint).

4.2.7 Q

Q is a must have promise (promises are a concept of values that will be resolved later) helper library. It helps with managing the "callback hell" (callback functions calling callback functions and forming a pyramid when following proper indentation). Q brings a very useful "then" function that allows handling the value of the promise inside the then function, it is called when the promise resolves. It also brings very useful "deferred" function, which is a sort of mock-up promise that can be handled just like regular promises and waited upon and can be resolved manually. (Q).

4.2.8 Less

Less is a pre-processor for CSS. It extends CSS by providing variables, mixins and functions. The functions allow you to write complex rules, even with parameters, only once instead of having to write or copy-paste them for each rule. Variables are a great way to easily change commonly used values across the rules, such as row height or font size, and have them affect everywhere on your CSS rules. You can also nest your selectors to make the CSS more readable and maintainable. Less is compiled into a CSS file that can then be used normally like any other CSS file. (Less).

4.2.9 Express

Express is a lightweight server for Node.js. It's extremely useful for testing applications that normally require a web server to run. It's also useful for delivering showcase versions of web applications when you don't have a web server. (Express).

4.2.10 Jade

Jade is a simple template engine for writing HTML. It was used along with Express as it shipped with it. (Jade).

4.3 **Other considered frameworks and libraries**

4.3.1 jQuery

jQuery is probably one of the most used and useful JavaScript libraries. It provides many useful features that are easy to use; especially DOM manipulation with it is extremely powerful. Previously jQuery was almost mandatory for any web application because it handled all the browser discrepancies for the programmer. Another useful feature of jQuery is its animations support, which provides an easy way to do simple animations, such as sliding, for your website without having to dwindle around with CSS (Cascading Style Sheets) animations. (jQuery).

However In this project jQuery was not needed since there is no need for any DOM manipulation with Angular as it's one of Angulars best features to handle all the dynamic DOM manipulations. In addition, Angular already comes shipped with its own light jQuery (jqLite) which has all the most used features from jQuery. (AngularJS – Element).

4.3.2 React

React is, like Angular, an mvc type framework that allows making dynamic websites and is highly modular. React is much lower level framework than Angular and requires a lot more work to get the same done. The upside with React over Angular is that you can make only part of your web application with React making it possible to bring it into an older project. React is being developed by Facebook. (React).

4.3.3 TypeScript

TypeScript is a superset of JavaScript that brings types into JavaScript. It is great for larger applications as it brings more coordination between everything by bringing types for the usually typeless JavaScript. It helps with developing by helping to notice simple errors easier. Another great feature of TypeScript is that you can use the latest ECMAScript features. TypeScript files are compiled into regular JavaScript files, which can be run in any browser. (TypeScript).

TypeScript was not used because I looked into it when about 50% of the project was completed. I tried to incorporate TypeScript retroactively but it didn't prove to bring enough to justify rewriting almost everything to follow the TypeScript way, and incorporating it without using types would have been pointless.

4.3.4 Bootstrap

Bootstrap is one of the most used frameworks for responsive websites. It is mainly focused on building mobile websites. It also comes with Less and Sass (another CSS preprocessor like Less), which allow to write cleaner CSS. Bootstrap integrates into all parts of the website HTML, JavaScript, and CSS. It offers many useful features and prebuilt components, which further eases the burden from the developer. Bootstrap requires jQuery to work. (Bootstrap).

Bootstrap was not used in this project because mobile was not the main target of the application and bootstrap is cumbersome to maintain and update with its horrible class syntax required on every html element to make it work properly.

```
<div class="row">
  <div class="col-sm-5 col-md-6">.col-sm-5 .col-md-6</div>
  <div class="col-sm-5 col-sm-offset-2 col-md-6 col-md-offset-0">.col-sm-5 .col-sm-
offset-2 .col-md-6 .col-md-offset-0</div>
</div>

<div class="row">
  <div class="col-sm-6 col-md-5 col-lg-6">.col-sm-6 .col-md-5 .col-lg-6</div>
  <div class="col-sm-6 col-md-5 col-md-offset-2 col-lg-6 col-lg-offset-0">.col-sm-6
.col-md-5 .col-md-offset-2 .col-lg-6 .col-lg-offset-0</div>
</div>
```

Picture 5 Example of Bootstraps grid based class syntax, from their guide

4.3.5 D3.js

D3.js is a data visualization library. The name comes from Data-Driven Documents. It is an extremely powerful tool for making interactive charts and visualizations from set data. (D3js).

D3.js was going to be used for visualizing the data from different renovation plans calculated on the app. The visualization feature was dropped from the project towards the end so there wasn't any use for this library.

4.4 **Key design points of the code**

There were many design decisions that needed to be made for the projects code architecture.

4.4.1 Pure client side

There was no clear idea on which platform the program was going to be running, so I made the program 100% client side. A server is only needed to distribute the files to the user. This is a bad idea in practice since it opens the application to be copied extremely easily by anyone who has access to it. However, the app was constructed in a way that there are clear points which are easy to import to server-side and should be moved there before deployment.

4.4.2 DataFactory

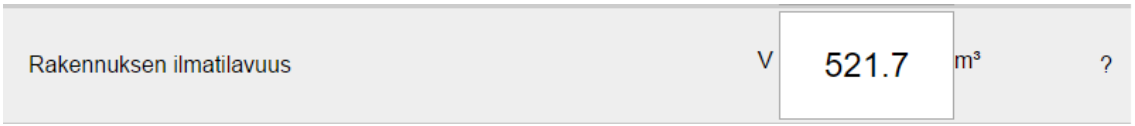
All the data collected from the different models on the program are stored on a centralized place. The dataFactory handles all the calculations that are required when any given value changes. The values on the objects inside the dataFactory are calculated with Angular watchers that handle the business logic of the program. The dataFactory passes on the values it holds when required giving us a simple centralized place to query for any values. By having all the business logic and data in a single place we provide an easy way to port this part to server side logic if needed.

4.4.3 Templates and models

Angular offers a great approach for writing less HTML when a lot of your front end is going to look pretty much the same. I made a base template for the different pages of the program and then again a template for each of the different type of input box that we needed on the program.

To support this template based model I made everything follow the same format, which allowed me to write object based models for each row on the page. By setting a common set of rules for the objects, I was able to make it easy to add new elements to the page.

```
rakennuksenIlmatilavuus: {
  name: 'rakennuksenIlmatilavuus',
  header: 'Rakennuksen ilmatilavuus',
  type: 'text',
  suffix: "m³",
  prefix: "V",
  bind: {
    name: "perussuureet",
    value: "rakennuksenIlmatilavuus"
  },
  source: 'ok2000-talossa vaipan alan lukuarvo = tilavuuden lukuarvo'
},
```



Picture 6 Example of a model object and how it looks when rendered

The header parameter defines the header of the row. Suffix and prefix define the suffixes and prefixes to be placed next to the input. Type defines the type of the input e.g. dropdown or text. The bind parameter defines which data we bind the input to on the dataFactory object. The source defines what shows on the popup window when hovering over the question mark. It's also possible to assign bunch of other parameters which define how the model works, for example you can define a disabled parameter which can provide a condition that disables the input or onChange to determine what happens when the value is changed etc.

4.4.4 JSON

There was a lot of predefined data in the Excel which was used in the calculations by certain user selected values. I decided to port this data in JSON format. JSON files are static text files made with strict syntax. These files can be parsed into JavaScript objects, which can be handled easily.

4.4.5 CSV-to-JSON tool

To help with moving all the tables from the Excel to JSON format I created a separate tool to transform a CSV (comma-separated value) file into a valid JSON file format. The program takes a CSV file and a schema file as a parameter and creates a JSON file from the CSV file in a format defined by the schema. The schema file is a JSON object with functions for its values and the data is formatted by these functions.

```
{
    "kunta": String,
    "saavyohyke": Number,
    "nimitys": String,
    "maakunta": String
}
```

Picture 7 A simple example of a schema file using build in String and Number constructors as functions

The object is called for each row in the CSV file to get a neatly formatted JSON file. The program also supports deeply nested JSON objects and complex formatting functions.

4.4.6 Final calculations

There was extreme amount of calculations that needed to be done to get the final energy efficiency numbers. To stay on schedule for the project I made a decision to make a JSON object format to handle all the calculation logic and its values, and then handle all the calculations in a single function that evaluates them to get the correct value.

4.4.7 Helper module

All the commonly needed functions that were needed by all the controllers or any function that were used in more than one place were moved to the helper module for easy maintaining and usability. For example the data binding function that was used to bind all the modules with the data in dataFactory and the function that polls and format data from the dataFactory.

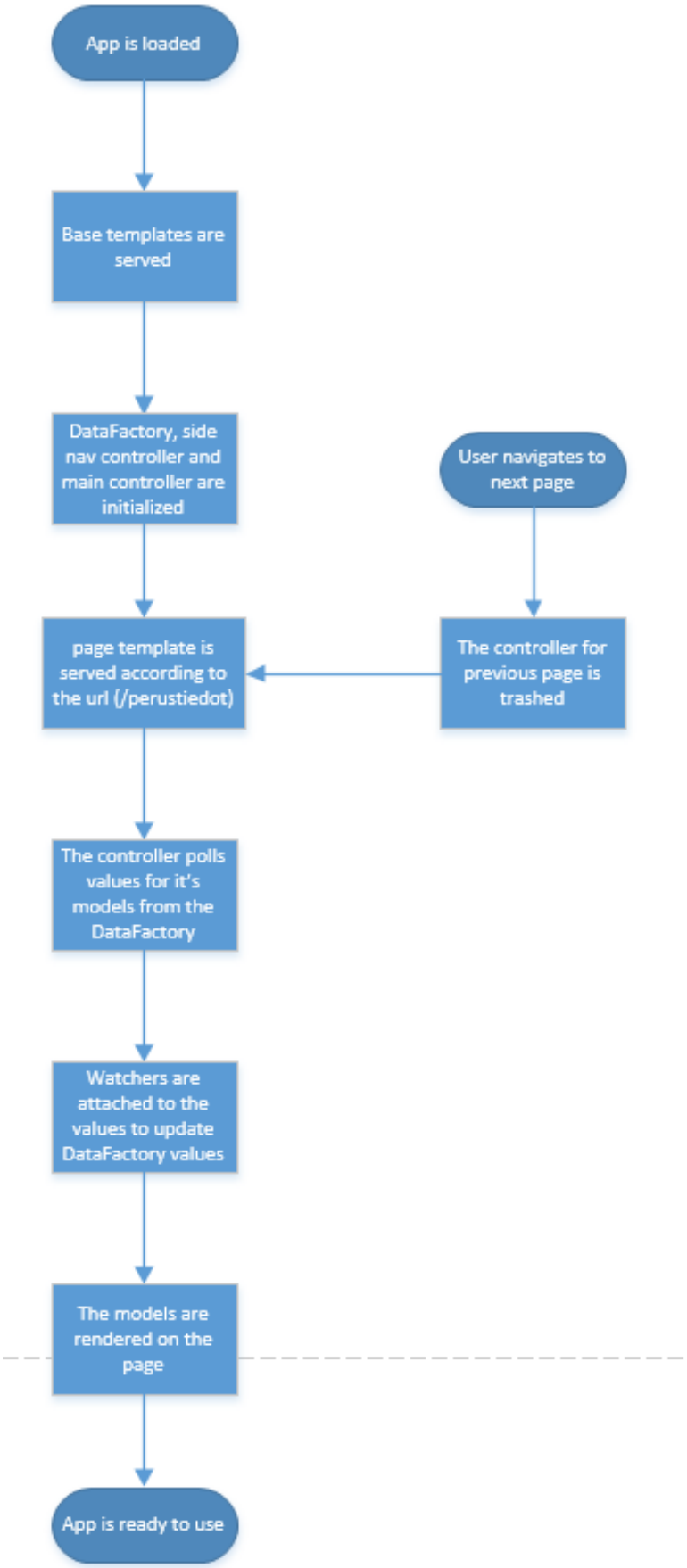
4.4.8 Testing

The program and all the calculations were tested manually to verify that all the calculations match with the values in the Excel. There is no test automation despite original plans to write some. There wasn't simply enough time to fit in writing tests for the program.

4.5 **Architecture**

4.5.1 Loading the app

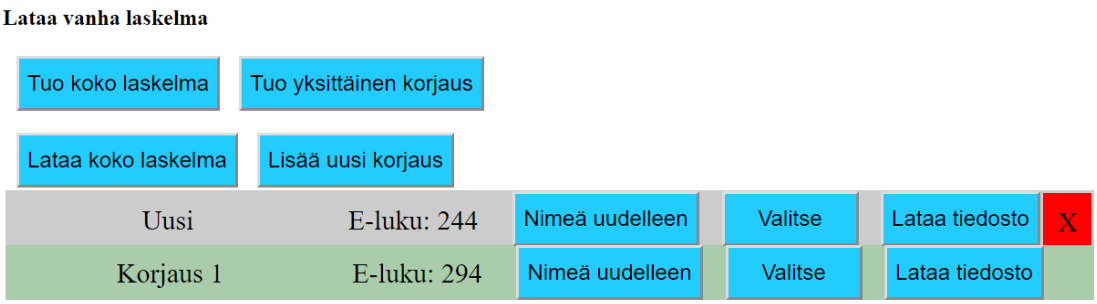
When a user navigates to the application, all the required files and the template of the page he is in (e.g. main page) are served. Angular initiates the DataFactory service and the main controller, side navigation bar controller and the controller of the page the user is on. The side navigation bar controller polls the page layout from the DataFactory and renders the side navigation bar. All the models from the controller of the page are rendered on the HTML page and shown to the user. After this the application is ready to be used. The final calculation is also triggered when the app is loaded.



Picture 8 Diagram of the application loading process

4.5.2 Main page

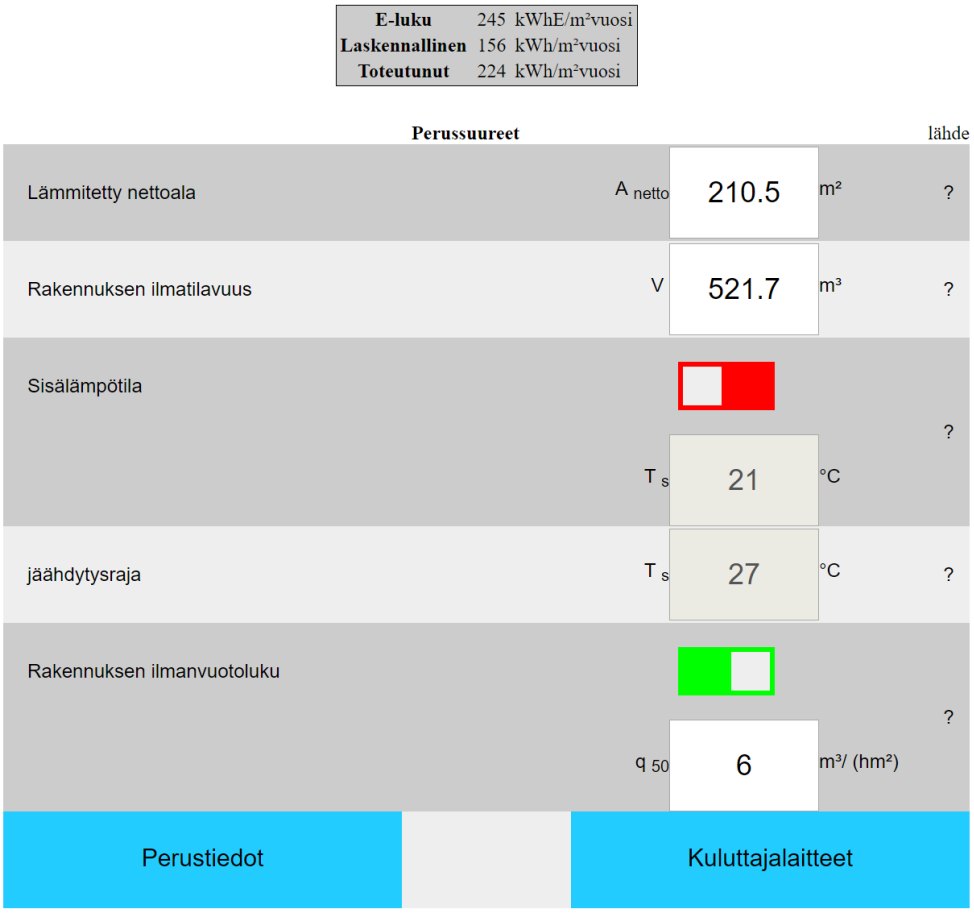
The main page uses the helper module to manage all files and exporting and downloading the files to be used. On the main page all the current editable files are shown and you can switch between them, rename them, save them, import new file lists or single files, and download the files or the file list. When a file is changed the final calculation for that file is triggered to verify the energy efficiency number.



Picture 9 The file list

4.5.3 Input page

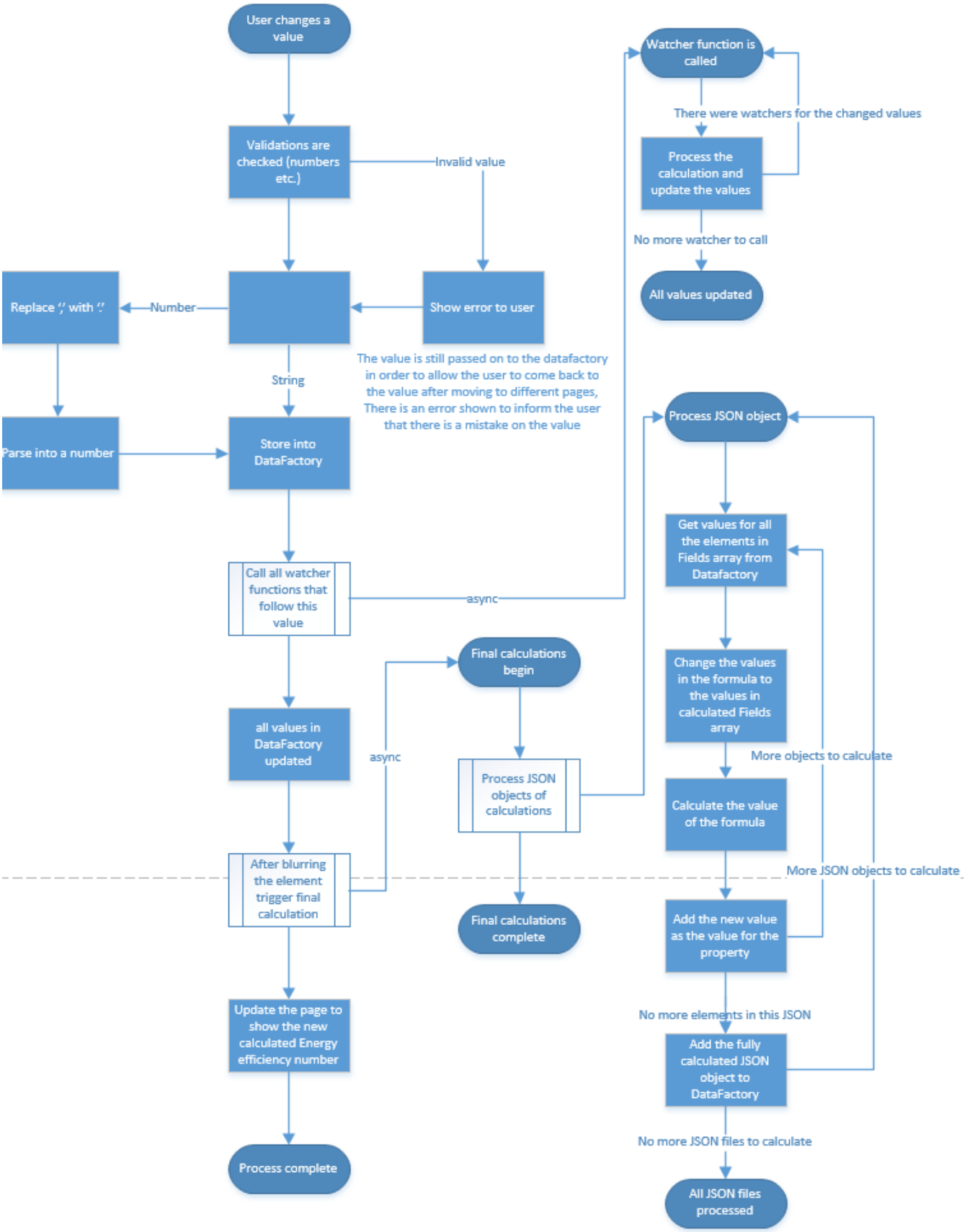
The rest of the pages are input pages where users can input values and they are used in the calculations. When user navigates to an input page the controller for that page is created and it polls data for all its models from the DataFactory according to the models bind properties. Then watchers are attached to all the models that have the bind property. The watchers handle the validation checking and keeping the properties in the DataFactory up to date.



Picture 10 Rendered Perussuureet input page

4.5.4 Changing input values

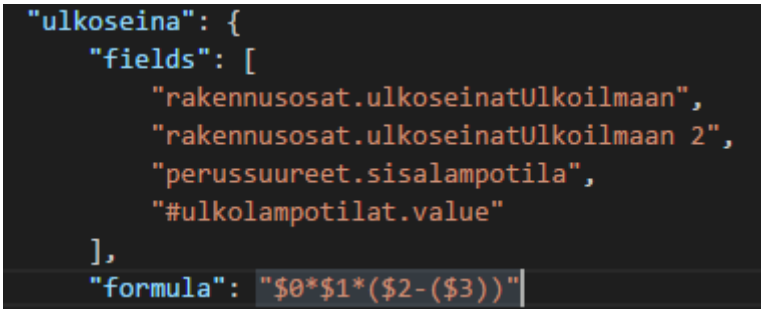
When user changes a value of an input field several things happen. The value on the model is updated and the watcher that was attached to it is triggered. The watcher checks the validity of the input and reports an error if the input didn't pass validation. Then the watcher replaces commas on number fields to dots to be able to parse them into actual number, if the field is a string nothing is done in this phase. The value is then passed on to the bound DataFactory property for storing. In the DataFactory all the watchers that are watching the value are triggered and their calculations are done and the values of their properties are updated and then their watchers trigger and so on until no more watcher are queued. If the input field is a free write text field then nothing happens until it's blurred, otherwise and after a text field is blurred the final calculations phase is triggered.



Picture 11 Background diagram of what happens when user changes value

4.5.5 Final calculation

This phase is responsible for doing all the extensive background calculations from the provided values to get the energy efficiency number of the building. When this phase is triggered it goes through the list of JSON objects and calculates each of them in order. The JSON object is a list of objects which either contain a value or an object with a fields array and formula properties. If the property is a value no calculation is done. In the field array all the values that are needed by the formula are defined and they correspond to data in the DataFactory or to data in the same JSON object when denoted with a hashtag (#). The formula property is a string which defines the formula for the calculation. In the formula, fields array values are referenced with $n notation where n is the index number of the desired value in the array.



Picture 12 Example of a calculatable object in JSON

The values referenced on the fields array are fetched via the helper module from Datafactury. Then fields array is replaced with the fetched values and the formula is completed with those values. Example $0 + $1 string would sum the first two values of the array. You can also do complex calculations with the formula such as calculating by values in specific indices in an array e.g. $0[$1+$2]. When all the values in the formula string are replaced the formula is evaluated with a function constructor. Evaluating strings is normally never recommended, but this is a solution which saved approximately 50 hours of time writing predefined functions for each, and this also makes it simpler to modify and write new formulas.

```
if (valuesObj.formula === "sum") {
    // Returns the sum of all the calculated values
    result = calculatedValues.reduce(function (value, previous) {
        return value + previous;
    }, 0);
}
else if (valuesObj.formula === "avg") {
    // Returns the average of all the calculated values
    result = calculatedValues.reduce(function (value, previous) {
        return value + previous;
    }, 0) / calculatedValues.length;
}
else {
    var finishedStr = valuesObj.formula.replace(/\$index/g, ind);
    for (var index = 0; index < calculatedValues.length; index++) {
        var repRegex = new RegExp("\\$" + index + "(?!\\d)", "g");
        // Replace the nth number on the string with nth value from the calculatedValues
        finishedStr = finishedStr.replace(repRegex, calculatedValues[index]);
    }
    var calcFn = new Function("return " + finishedStr);
    result = calcFn();
}
return result;
```

Picture 13 Calculating the values

4.6 **Problems**

Just like any software project, this didn't go through without problems.

4.6.1 Excel

The Excel sheet was huge and all the calculations were intertwined like spaghetti. And on top of that the Excel was practically undocumented so I had to go through all the cells one by one with the debugger to find which shell each part on the function corresponded to.

```
=JOS($G$171*$G$173*1,2*$G$163*($G$167-G342-$G$169-$G$159*($G$92-G342))*G344<0;0;JOS($H$72="Ei";
0;JOS($F$72="Ei";$G$171*$G$173*1,2*$G$163*($G$167-G342-$G$169)*G344;$G$171*$G$173*1,2*$G$163*(
$G$167-G342-$G$169-$G$159*($G$92-G342))*G344)))
```

Picture 14 Example of one of the worser functions on the Excel

There were many shells with worse or just as bad functions as this. There were even a couple of functions in the Excel that were incorrect and debugging those was horrible.

4.6.2 Specifications

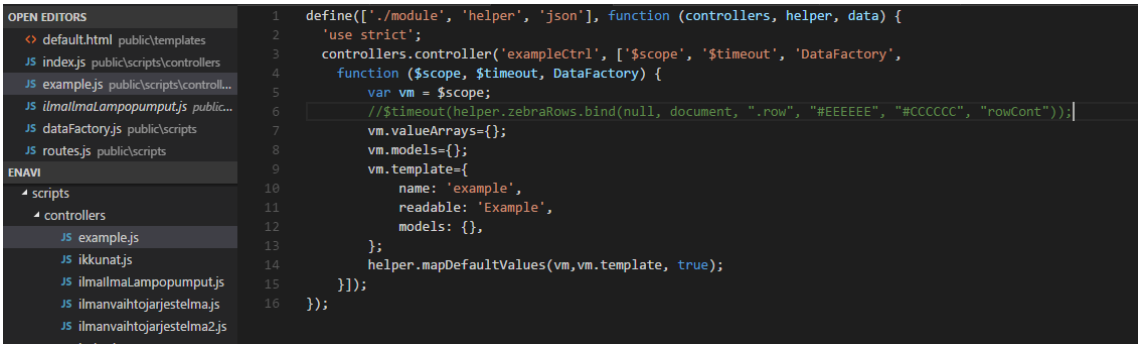
There were some problems with coming up with proper specifications for the application because Sykli is not specialized in IT-field. Because of this I had to work a bit harder on making the specifications based on the Excel-tool.

5 INSTRUCTIONS

Here are some instructions on how to make common changes to the application. These instructions are for high level programming with this application, low level isn't touched here.

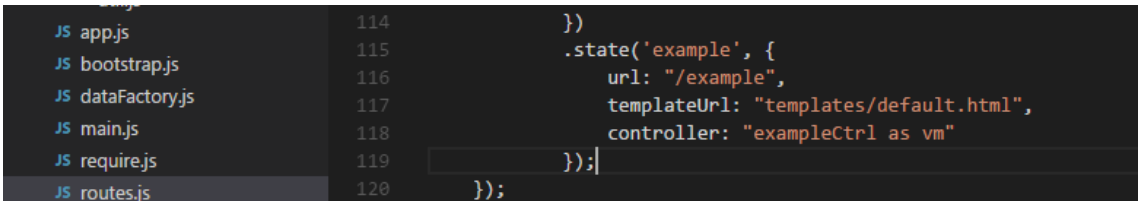
5.1 Adding new page

First, you need to create a controller for the new page in the controllers folder under scripts and add the controller to the index.js file on the same folder. You can look at other controllers for example for the controller base. There was a snippet for creating directives with visual studio code but the directives idea was ditched when we moved to routing with Angular Ui-router, and everything was converted to controllers instead. The zebra rows function is commented out until we add models to the page.



Picture 15 The controller should look like this

Then you need to add routing to be able to navigate to the page, write into the routes.js located in scripts folder. Here you will define what URL routes to this page, which template does this page use, and bind the new controller to this page.



Picture 16 The routing should look like this

Now it's possible to navigate to the new page through the provided URL, localhost:3000/#/example in this case. The page looks barren because we have no models to add to it yet.

E-luku	244 kWhE/m²vuosi
Laskennallinen	155 kWh/m²vuosi
Toteutunut	224 kWh/m²vuosi

Example lähde

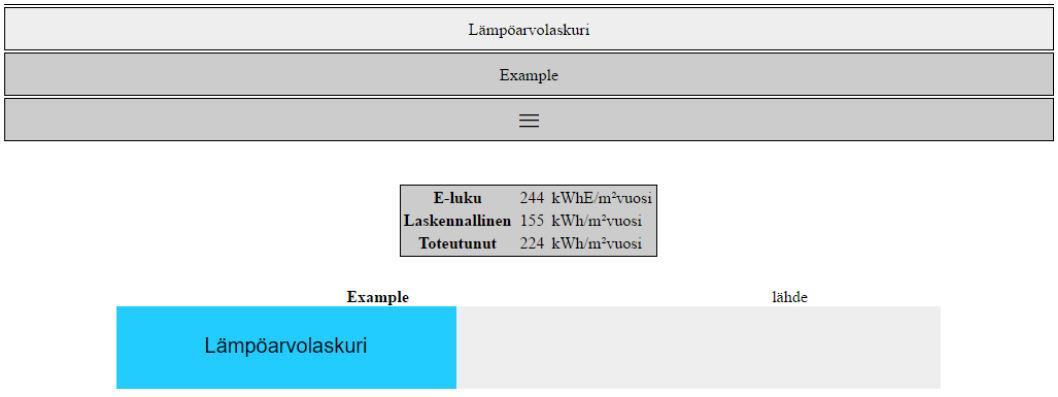
Picture 17 The page should look like this now

Next, we want to add this new page to the navigation bar. We need to add it to the pageLayout object in the dataFactory.js located in the scripts folder. The pageLayout object defines the ordering of the pages.

```
{
    id: "lampoarvolaskuri",
    display: "Lämpöarvolaskuri"
},
{
    id: "example",
    display: "Example"
}
```

Picture 18 Appended to the bottom of the pageLayout object

Now we can navigate to the page with the side navigation bar and on the footer (unless the page has hideOnFooter property set to true). The footer is also now automatically added to the bottom of this page.



Picture 19 Our example page should now look like this (on smaller screen, on bigger screen the navigation menu is on the side)

Finally, we also want to add a spot in the DataFactory. This is where we will store the values of the models of this page.

```
var example = {

};
return {
    pageLayout: pageLayout,
    perustiedot: perustiedot,
    tilojenLammitysjarjestelma: tilojenLammitysjarjestelma,
    kayttovesijarjestelma: kayttovesijarjestelma,
    ilmanvaihtojarjestelma: ilmanvaihtojarjestelma,
    kuukausiTalteenotto: kuukausiTalteenotto,
    perussuureet: perussuureet,
    lammitysjarjestelma: lammitysjarjestelma,
    kayttovesijarjestelma2: kayttovesijarjestelma2,
    ilmanvaihtojarjestelma2: ilmanvaihtojarjestelma2,
    ikkunat: ikkunat,
    rakennusosat: rakennusosat,
    kuluttajalaitteet: kuluttajalaitteet,
    lammonJakelunHaviot: lammonJakelunHaviot,
    jaahdytysjarjestelma: jaahdytysjarjestelma,
    lampopumppuLammitysmuotona: lampopumppuLammitysmuotona,
    lisalammitysjarjestelmia: lisalammitysjarjestelmia,
    ilmaIlmaLampopumput: ilmaIlmaLampopumput,
    toteutunutEnergiankulutus: toteutunutEnergiankulutus,
    varaavienTulisijojenPolttoaineet: varaavienTulisijojenPolttoaineet,
    lampoarvolaskuri: lampoarvolaskuri,
    example: example
};
```

Picture 20 Add a new empty object and add it to the return object of the DataFactory

That's all we need to do to add a new page to the app.

5.2 **Adding new models**

First, we want to add the models to the DataFactory under the object for the page we want them added to. We should also add any required watchers if we want the model to be calculated from some other values. We'll add two models for this example, the other pulls data from the number of inhabitants (asukasmäärä) under basic info (perustiedot) and multiplies it by the other models' value we add.

```
var example = {
    multiplier: 1,
    multiplied: null
};
```

Picture 21 The added properties for example page

The multipliers default value will be 1 and the multiplied will be calculated by a watcher function so we set it to null.

Next, we need to add a watcher for the multiplied value. Common convention for this project is to add a function where we define all the watchers for each of the objects properties, in this case exampleWatchers function was made. This function should be called directly after the definition of example object.

```
function exampleWatchers(){
    $rootScope.$watch(
        function () {
            return {
                multiplier: example.multiplier,
                inhabitants: perustiedot.asukasmaara
            };
        },
        function (values) {
            example.multiplied = values.multiplier * values.inhabitants;
        },
        true
    );
}
```

Picture 22 The added watcher function for example.multiplied

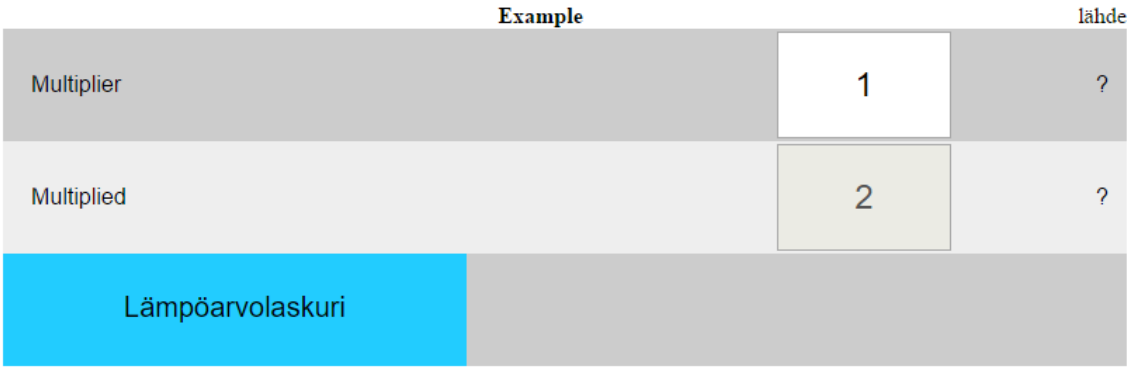
The first function of the watcher returns the values we want to watch with this watcher and the second handles the calculation. Now every time either of those values are changed the value of example.multiplied gets updated.

Now we want to add the models to the template of example page. We need to define their bind properties to match them with the object in DataFactory. We also want to add a disabled property to the multiplied model and set it to string “true” to make this field read-only.

```
vm.template={
    name: 'example',
    readable: 'Example',
    models: {
        multiplier: {
            name: 'multiplier',
            header: 'Multiplier',
            type: 'text',
            bind: {
                name: "example",
                value: "multiplier"
            },
            source: 'An example model'
        },
        multiplied: {
            name: 'multiplied',
            header: 'Multiplied',
            type: 'text',
            disabled: "true",
            bind: {
                name: "example",
                value: "multiplied"
            },
            source: 'An example model'
        }
    },
```

Picture 23 The added models

Now that we have the models added and bound properly they will be rendered on the example page. And they'll have their values set to the values in the DataFactory.



Picture 24 The rendered models on the example page

Now we notice that the value of multiplied doesn't actually change when we change the multiplier value. It does change on the DataFactory and

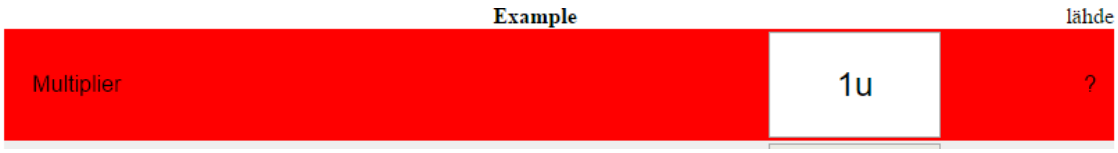
we can see the changed value if we navigate away and back to the example page. To fix this we need to add on onChange property to our multiplied model to poll for new value from DataFactory.

```
var pollOnChange = function(){
    $timeout(function(){
        vm.template.models.multiplied.model = helper.getValue("example.multiplied");
    });
};
vm.template={
    name: 'example',
    readable: 'Example',
    models: {
        multiplier: {
            name: 'multiplier',
            header: 'Multiplier',
            type: 'text',
            onChange: pollOnChange,
            bind: {
                name: "example",
                value: "multiplier"
            },
            source: 'An example model'
        },
```

Picture 25 Added onChange function

We need to update the models value of multiplied inside a $timeout function, this way it'll be queued to be updated after all the watchers are finished updating in DataFactory.

Multiplier model will automatically be assigned a validator to check that it's a number, we need to explicitly tell not to validate models.



Picture 26 Example of the validator

That's all we need to do to add new models for pages.

5.3 Updating final calculations

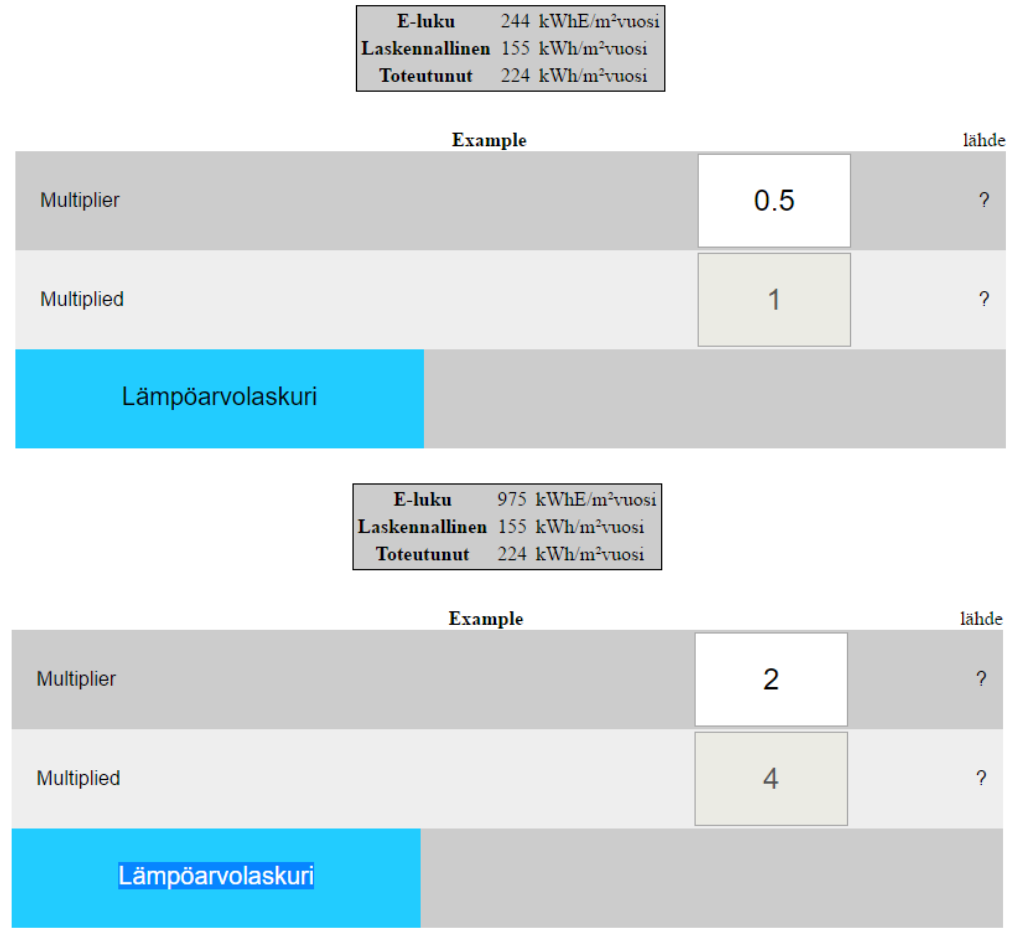
If we want our new example multiplied value to affect the final calculations, we need to use it somewhere on the final calculations JSONs. First we need to locate the JSON file we want to modify, for this example we'll just use the multiplied value to multiply the sum of energy efficiency number. All the JSON files for final calculations are found in laskenta folder, there are different subfolders for different phases in the final calculations.

We'll go to the eLuku sub folder and open the summa.json file. There we need to add our "example.multiplied" to the fields array in the "eLuku" property and then modify the formula to multiply the rest of the values.

```
"eLuku": {
    "fields": [
        "eLukuSahko.sahko.eArvo",
        "eLukuKaukolampo.kaukolampo.eArvo",
        "eLukuKaukojaahdytys.kaukojaahdytys.eArvo",
        "eLukuFossiilisetPolttoaineet.fossiilisetPolttoaineet.eArvo",
        "eLukuUusiutuvatPolttoaineet.uusiutuvatPolttoaineet.eArvo",
        "example.multiplied"
    ],
    "formula": "($0+$1+$2+$3+$4)*$5"
}
```

Picture 27 The new modified energy efficiency calculation

And that's it, now our new value will be used to multiply the actual energy efficiency number.



Picture 28 Our Multiplied value affecting the energy efficiency number

6 SUMMARY

The application successfully calculates the energy efficiency numbers for a house the same as the Excel, expect for few fields where the Excel calculation was wrong. It's possible to save and import old calculations from the application. All the pages are neatly formatted on their own page that can be easily navigated to through the side navigation bar. The calculations are done every time the user changes a value to make it simple for them to check small changes on the energy efficiency number.

Overall, the development of the program went well and I am satisfied with the result. During the production of the program, there were times when the workload seemed too much. The Excel was really complicated, and working with it was tedious.

From this project I learned a lot about developmental lifecycle and more about building Angular apps. I also learned about creating applications with modularity and ability to further develop it easily in mind.

The development of the application is supposed to continue after the thesis and all the code was written in a way to support easy modifications and additions to any parts of the code.

REFERENCES

Laki rakennuksen energiatodistuksesta 2013/50. Retrieved 17.10.2016 from http://www.finlex.fi/fi/laki/ajantasa/2013/20130050

Energiatodistus – Mikä on energiatodistus. Retrieved 17.10.2016 from http://energiatodistus.motiva.fi/mika-on-energiatodistus/

Vuolle M. (n.d.) Rakennusten energiatodistus ja sen E-luvun laskenta. Retrieved 17.10.2016 from http://energiatodistus.motiva.fi/energiatodistustenlaatijat/tapahtumat/et_vuolle_verkkoon_19032013.pdf

Sykli – EEnavi. Retrieved 18.10.2016 from http://www.sykli.fi/fi/hankkeet-ja-julkaisut/eenavi

Sykli – EEnavi yrityksille. Retrieved 18.10.2016 from http://www.sykli.fi/fi/hankkeet-ja-julkaisut/eenavi/yritykset

Sykli – EEnavi pientaloasukkaat ja taloyhtiöt. Retrieved 18.10.2016 from http://sykli.fi/fi/hankkeet-ja-julkaisut/eenavi/pientaloasukkaat-ja-taloyhtiot

Oagile. (2014) Waterfall Software Development Model. Blog publication 5 February 2014. Retrieved 19.10.2016 from http://www.oxagile.com/company/blog/the-waterfall-model/

Visual Studio Code. Retrieved 23.10.2016 from https://code.visualstudio.com/

AngularJS. Retrieved 20.10.2016 from https://angularjs.org/

AngularJS – ui router. Retrieved 20.10.2016 from https://github.com/angular-ui/ui-router/wiki

RequireJS. Retrieved 20.10.2016 from https://github.com/angular-ui/ui-router/wiki

Git. Retrieved 22.10.2016 from https://github.com/angular-ui/ui-router/wiki

ESLint. Retrieved 22.10.2016 from http://eslint.org/docs/about/

Q. Retrieved 21.10.2016 from http://documentup.com/kriskowal/q/

Less. Retrieved 21.10.2016 from http://lesscss.org/

Express. Retrieved 21.10.2016 from http://expressjs.com/

Jade. Retrieved 21.10.2016 from https://www.npmjs.com/package/jade

jQuery. Retrieved 22.10.2016 from http://jquery.com/

AngularJS – Element. Retrieved 22.10.2016 from https://docs.angularjs.org/api/ng/function/angular.element

React. Retrieved 22.10.2016 from https://facebook.github.io/react/

TypeScript. Retrieved 22.10.2016 from https://www.typescriptlang.org/

Bootstrap. Retrieved 22.10.2016 from http://getbootstrap.com/

D3js. Retrieved 22.10.2016 from https://d3js.org/

PICTURES

Appendix 1

Index page (index.jade)

```
extends layout

block content
  .banner
    h1
  sidenav
  span(id="tooltip")
    span(id="tooltipHeader")
      h4 Lähde
    span(id="tooltipText")
  .holder
    laskenta_directive
    div(ui-view)
    footer_directive
```

Appendix 2

Style sheet (style.less)

```
@rowHeight: 80px;
@defaultFont-size: 16px;

.center(){
  margin-left: auto;
  margin-right: auto;
}

.font(){
  font-size: @defaultFont-size;
  font-family: Arial,Verdana,"Times new roman";
}

.centerDiv(@height){

}

.centerText(@height: 10px,@font: @defaultFont-size){
  text-align: center;
  padding-top: (@height/2)-(@font)/2;
  height: (@height/2)+(@font);
  padding-bottom: -@font/2;
  font-size: @font;
  line-height: 0px;
}

.textFont(){
  font-size: 24px;
  text-align: center;
}

.holder {
  width: 800px;
  padding-top: 40px;
  .center;
}

.banner {
  width: 100%;
  background: red;
  margin-bottom: 0px;
  padding-bottom: 0px;
  .centerText(400px,50px);
}

table {
  width: 100%;
}

tr {
  text-align: center;
}

td {
  margin: 0;
  padding: 0;
}
```

```
.basicName {
  width: 100%;
  text-align: left;
  padding-left: 20px;
}

.info {
  width: 10%;
  &:hover {
    cursor: help;
  }
}

.double, .double + tr{
  .font;
  & .doubleDown, .doubleUp{
    text-align: left;
    padding-left: 20px;
    width: 60%;
  }
  & .doubleUp {
    margin-top: 100px;
  }
}

.doubleUp {

}

.textInputWide {
  height: 70px;
  width: 120px;
  .textFont;
}

.dropdownInput{
  border: 1px dotted grey;
  background-color: inherit;
  width: 100%;
  min-width: 300px;
  height: 70px;
  margin-top: 0px;
  text-align: center;
  font-size: @defaultFont-size;
  & option {
    height: 70px;
    background-color: white;
    .centerText(@rowHeight);
    &:not(:first-child){
      border-top: 1px dotted black;
    }
  }
}

.row{
  height: @rowHeight;
  .font;
}

.suffix {
  width: 100%;
  min-width: 90px;
  margin-left: 0px;
```

```
  padding-left: 0px;
  text-align: left;
}

.prefix {
  width: 100%;
  min-width: 120px;
  margin-right: 0px;
  padding-right: 0px;
  text-align: right;
}

.textInputFull {
  width: 100%;
  font-size: 20px;
  text-align: left;
}

.textInputHolder {
  margin: 0px;
  padding: 0px;
  width: 120px;
}

.textInput {
  height: 70px;
  width: 120px;
  .textFont;
}

.staticValue {
  max-width: 300px;
  min-width: 300px;
  text-align: center;
}

.window {
  width: 90px !important;
  margin: 0px;
  padding: 0px;
}

.ilmansuunta {
  border: 1px dotted grey;
  background-color: inherit;
  width: 140px;
  min-width: 140px;
  height: 70px;
  margin-top: 0px;
  text-align: center;
  font-size: @defaultFont-size;
  & option {
    height: 70px;
    background-color: white;
    .centerText(@rowHeight);
    &:not(:first-child){
      border-top: 1px dotted black;
    }
  } ;
}

.inputHolder {
  height: 70px;
  overflow-y: hidden;
```

```
}

#tooltip {
  position: absolute;
  display: none;
  height: 100px;
  text-align: center;
  width: 400px;
  z-index: 9999;
  opacity: 0.75;
  background-color: #FFFFFF;
  border-radius: 10px;
  border: 1px solid black;
}

#tooltipTet {
  opacity: 1;
}

.extraInfo {
  border: 1px solid black;
  text-align: left;
  font-size: 14px;
  & tr {
    height: 30px;
    & th {
      border-top: 1px solid black;
      border-bottom: 1px solid black;
    }
    & td {
      border-left: 1px solid black;
      border-right: 1px solid black;
      &:first-child {
        width: 30%;
      }
      &:nth-child(2){
        width: 60%;
      }
    }
  }
}

.table {
  display: block;
  &:not(first-child){
    /*padding-top: 50px;
    color: red;*/
  }
}

.fireplace {
  width: 80px !important;
}

.heatCalc {
  width: 100px !important;
}

.smallScr {
  float: none !important;
  width: 100% !important;
  padding: 0px;
  margin: 0px;
}
```

```
.sidenav {
  list-style-type: none;
  float: left;
  display: block;
  color: red;
  -moz-transition: height 10s ease;
  -webkit-transition: height 10s ease;
  -o-transition: height 10s ease;
  transition: height 10s ease;
  & li {
    width: 250px;
    height: 40px;
    .smallScr {
      width: 100% !important;
    }
    margin-bottom: 2px;
    border: 1px solid black;
  }
  & a {
    text-decoration: none;
    text-align: center;
    display: inline-block;
    color: black;
    padding-top: 12px;
    padding-bottom: -12px;
    width: 100%;
    height: 28px;
    &:hover {
      background-color: #CCCCCC;
    }
    &.active {
      background-color: #CCCCCC;
    }
  }
}
#hideSideNav {
  font-size: 36px;
  & a {
    padding-top: 0px;
    height: 40px;
  }
  &:hover {
    cursor: pointer;
    background-color: #CCCCCC;
  }
}
#showSideNav {
  font-size: 36px;
  text-align: center;
  display: block;
  margin: 0px;
  padding: 0px;
  width: 100%;
  height: 40px;
  background-color:  #EEEEEE;
  &.hamburger {
    padding-top: 20px;
    padding-bottom: -12px;
    height: 28px;
  }
  &:hover {
    cursor: pointer;
    background-color: #CCCCCC;
```

```
  }
}

.footer {
  width: 100%;
  text-align: center;
  font-family: Arial,Verdana,"Times new roman";
  font-size: 20px;
  background-color: #EEEEEE;
  & .footButton {
    height: @rowHeight;
    width: 330px;
    background-color: #22CCFF;
    height: 100%;
    display: block;
    & span {
      display: block;
      padding-top: 27px;
    }
    &:hover {
      background-color: #11BBCC;
      cursor: pointer;
    }
  }
  & #calculate {
    display: inline-block;
    padding-top: 15px;
    width: 140px;
    height: 65px;
    background-color:  #44AA44;
    &:hover{
      background-color:  #229922;
      cursor: pointer;
    }
    & #eLuku {
      display: inline-block;
      padding-top: 5px;
    }
  }
  & #prevButton {
    float: left;
  }
  & #nextButton {
    float: right;
  }
}

.switch {
  position: relative;
  display: inline-block;
  width: 80px;
  height: 40px;
  margin-top: 5px;
  &  input {
    display:none;
  }
}

.slider {
  position: absolute;
  cursor: pointer;
  top: 0;
  left: 0;
  right: 0;
```

```
  bottom: 0;
  background-color: #FF0000;
  -webkit-transition: .4s;
  transition: .4s;
  &:before {
    position: absolute;
    content: "";
    width: 32px;
    height: 32px;
    top: 4px;
    left: 4px;
    bottom: 4px;
    background-color: #EEEEEE;
    -webkit-transition: .4s;
    transition: .4s;
  }
}

input {
  &:checked {
    & + .slider{
      background-color: #00FF00;
    }
    & + .slider:before {
      -webkit-transform: translateX(40px);
      -ms-transform: translateX(40px);
      transform: translateX(40px);
    }
  }
  &:focus + .slider {
    box-shadow: 0 0 5px #00FF00;
  }
}

.laskentaTable{
  width: 250px;
  margin-left: auto;
  margin-right: auto;
  background-color: #CCCCCC;
  border: 1px solid black;
  margin-bottom: 30px;
  & tr {
    border: 1px solid black;
  }
}

.laskentaHeader {
  width: 100px;
  text-align: center;
}
.laskentaValue {
  width: 50px;
  text-align: center;
}
.laskentaSuffix {
  width: 100px;
  text-align: left !important;
}

.button-norm {
  height: 40px;
  min-width: 100px;
  font-size: 16px;
  background-color: #22CCFF;
```

```
    text-align: center;
    padding-top: 0px;
    margin-top: -2px;
    margin-left: 10px;
    &:hover {
      cursor: pointer;
      background-color: #11BBCC;
    }
  }

    .current {
      background-color: #AACCAA !important;
    }

  .fileList {
    margin-top: 0px;
    //border: 1px solid black;
    display: block;
    background-color: #CCCCCC;
    text-align: center;
    height: 40px;
    & .eLuku {
      width: 140px !important;
    }
    & .fileName {
      font-size: 20px;
      width: 280px;
    }
    & .buttonTd {
      width: 120px;
    }
    & .deleteTd {
      width: 40px;
      & .removeX {
        background-color: #FF0000;
        width: 100%;
        height: 30px;
        padding-top: 10px;
        font-size: 20px;
        display: block;
        &:hover{
          background-color: #AA3333;
          cursor: pointer;
        }
      }
    }
  }

  .errorRow {
    background-color: red !important;
  }
```

Appendix 3

Main requireJS entry point (Main.js)

```
require.config({

    // alias libraries paths
    paths: {
        'domReady': '../lib/requirejs-domready/domReady',
        'angular': '../lib/angular/angular',
        'uiRouter': '../lib/angular/angular-ui-router',
        'helper': './helpers/util',
        'text': '../lib/requirejs-text/text',
        'jsonr': '../lib/require/json',
        'json': '../JSON/jsons',
        'laskenta': '../laskenta/laskenta',
        'q': '../lib/q/q'
    },

    // angular does not support AMD out of the box, put it in a shim
    shim: {
        'angular': {
            exports: 'angular'
        },
        'uiRouter': {
            deps: ['angular']
        }
    },

    deps: [
        './bootstrap'
    ]
});
```

Appendix 4

Angular bootstrapping (bootstrap.js)

```
/**
 * bootstraps angular onto the window.document node
 * NOTE: the ng-app attribute should not be on the index.html when using
ng.bootstrap
 */
define([
    'require',
    'angular',
    'app',
    'routes',
    'dataFactory'
], function (require, ng) {
    'use strict';

    require(['domReady!'], function (document) {
        ng.bootstrap(document, ['app']);
    });
  }
);
```

Appendix 5

Main Angular entry point (app.js)

```
define([
    'angular',
    'helper',
    'uiRouter',
    './controllers/index',
    './directives/index'
], function (ng, helper) {
    'use strict';

    return ng.module('app', [
        'app.controllers',
        'app.directives',
        'ui.router'
    ]).run(function (DataFactory, $rootScope, $timeout) {
        helper.initFactory(DataFactory);
        $rootScope.$on("$stateChangeSuccess", function () {
            $timeout(function () {
                var currentPage = location.hash.match(/\/(.+)$/)[1];
                var currentActive = document.querySelector(".active");
                if (currentActive) {
                    currentActive.classList.remove("active");
                }
                document.querySelector("a[ui-sref~='" + currentPage +
"']").classList.add("active");
            });
        });
    });
});
```

Appendix 6

Routing (routes.js)

```
define(['./app', './controllers/index'], function (app) {
    'use strict';
    return app.config(function ($stateProvider, $urlRouterProvider) {
        //
        // For any unmatched url, redirect to /state1
        $urlRouterProvider.otherwise("/main");
        //
        // Now set up the states
        $stateProvider
            .state('main', {
                url: "/main",
                templateUrl: "templates/main.html",
                controller: "mainCtrl as vm"
            })
            .state('perustiedot', {
                url: "/perustiedot",
                templateUrl: "templates/perustiedot.html",
                controller: "perustiedotCtrl as vm"
            })
            .state('tilojenLammitysjarjestelma', {
                url: "/tilojenLammitysjarjestelma",
                templateUrl: "templates/default.html",
                controller: "tilojenLammitysjarjestelmaCtrl as vm"
            })
            .state('kayttovesijarjestelma', {
                url: "/kayttovesijarjestelma",
                templateUrl: "templates/default.html",
                controller: "kayttovesijarjestelmaCtrl as vm"
            })
            .state('ilmanvaihtojarjestelma', {
                url: "/ilmanvaihtojarjestelma",
                templateUrl: "templates/default.html",
                controller: "ilmanvaihtojarjestelmaCtrl as vm"
            })
            .state('kuukausiTalteenotto', {
                url: "/kuukausiTalteenotto",
                templateUrl: "templates/kuukausi-talteenotto.html",
                controller: "kuukausiTalteenottoCtrl as vm"
            })
            .state('perussuureet', {
                url: "/perussuureet",
                templateUrl: "templates/default.html",
                controller: "perussuureetCtrl as vm"
            })
            .state('lammitysjarjestelma', {
                url: "/lammitysjarjestelma",
                templateUrl: "templates/default.html",
                controller: "lammitysjarjestelmaCtrl as vm"
            })
            .state('kayttovesijarjestelma2', {
                url: "/kayttovesijarjestelma2",
                templateUrl: "templates/default.html",
                controller: "kayttovesijarjestelma2Ctrl as vm"
            })
            .state('ilmanvaihtojarjestelma2', {
                url: "/ilmanvaihtojarjestelma2",
                templateUrl: "templates/default.html",
                controller: "ilmanvaihtojarjestelma2Ctrl as vm"
            })
            .state('ikkunat', {
```

```
            url: "/ikkunat",
            templateUrl: "templates/ikkunat.html",
            controller: "ikkunatCtrl as vm"
        })
        .state('rakennusosat', {
            url: "/rakennusosat",
            templateUrl: "templates/rakennusosat.html",
            controller: "rakennusosatCtrl as vm"
        })
        .state('kuluttajalaitteet', {
            url: "/kuluttajalaitteet",
            templateUrl: "templates/default.html",
            controller: "kuluttajalaitteetCtrl as vm"
        })
        .state('lammonJakelunHaviot', {
            url: "/lammonJakelunHaviot",
            templateUrl: "templates/default.html",
            controller: "lammonJakelunHaviotCtrl as vm"
        })
        .state('jaahdytysjarjestelma', {
            url: "/jaahdytysjarjestelma",
            templateUrl: "templates/default.html",
            controller: "jaahdytysjarjestelmaCtrl as vm"
        })
        .state('lampopumppuLammitysmuotona', {
            url: "/lampopumppuLammitysmuotona",
            templateUrl: "templates/default.html",
            controller: "lampopumppuLammitysmuotonaCtrl as vm"
        })
        .state('lisalammitysjarjestelmia', {
            url: "/lisalammitysjarjestelmia",
            templateUrl: "templates/default.html",
            controller: "lisalammitysjarjestelmiaCtrl as vm"
        })
        .state('ilmaIlmaLampopumput', {
            url: "/ilmaIlmaLampopumput",
            templateUrl: "templates/ilmaIlmaLampopumput.html",
            controller: "ilmaIlmaLampopumputCtrl as vm"
        })
        .state('toteutunutEnergiankulutus', {
            url: "/toteutunutEnergiankulutus",
            templateUrl: "templates/toteutunutEnergiankulutus.html",
            controller: "toteutunutEnergiankulutusCtrl as vm"
        })
        .state('varaavienTulisijojenPolttoaineet', {
            url: "/varaavienTulisijojenPolttoaineet",
            templateUrl: "templates/varaavienTulisijojenPolttoaineet.html",
            controller: "varaavienTulisijojenPolttoaineetCtrl as vm"
        })
        .state('lampoarvolaskuri', {
            url: "/lampoarvolaskuri",
            templateUrl: "templates/lampoarvolaskuri.html",
            controller: "lampoarvolaskuriCtrl as vm"
        })
        .state('example', {
            url: "/example",
            templateUrl: "templates/default.html",
            controller: "exampleCtrl as vm"
        });
    });
});
```

Appendix 7

Main helper file Module (util.js)

```
define(["q"], function (q) {
  'use strict';

  var Helpers = function () {
    var data;
    var files = {
      files: []
    };
    var curFile;
    var dataPromise = q.defer();

    this.initFactory = function (dataFactory) {
      if (dataFactory) {
        data = dataFactory;
        var file = {
          name: "Uusi",
          value: data
        };
        files.files.push(file);
        curFile = file;
        dataPromise.resolve();
      }
    };

    this.addFile = function (file) {
      files.files.push(file);
    };

    this.changeFiles = function (fileList) {
      files = fileList;
      curFile = files.files[0];
      this.setValues(curFile, true);
    };

    this.renameFile = function (file, name) {
      file.name = name;
    };

    this.getFiles = function () {
      return files;
    };

    this.getCurFile = function () {
      return curFile;
    };

    this.getData = function () {
      return data;
    };

    this.removeFile = function (file) {
      files.files.splice(files.files.indexOf(file), 1);
      return files;
    };

    this.setValues = function (newFile, isNew) {
      if (isNew !== true) {
        files.files[files.files.indexOf(curFile)].value =
JSON.parse(JSON.stringify(data));
      }
```

```
      curFile = newFile;
      var keys = Object.keys(newFile.value);
      for (var key of keys) {
        var deepKeys = Object.keys(newFile.value[key]);
        for (var dKey of deepKeys) {
          if (data[key]) {
            data[key][dKey] = newFile.value[key][dKey];
          }
          else {
            data[key] = {};
            data[key][dKey] = newFile.value[key][dKey];
          }
        }
      }
      curFile.value = data;
    };

    this.save = function (file) {
      var filename = file ? file.name + ".json" : 'Laskelma.json';
      var blob = new Blob([JSON.stringify(file ? file : files, null, 2)], { type:
'text/json' });
      if (window.navigator && window.navigator.msSaveOrOpenBlob)
        window.navigator.msSaveOrOpenBlob(blob, filename);
      else {
        var a = document.createElement("a");
        a.download = filename;
        a.href = window.URL.createObjectURL(blob);
        a.dataset.downloadurl = ['text/json', a.download, a.href].join(':');
        a.click();
      }
    };

    this.addToModels = function (scope, name) {
      if (data) {
        data[name] = scope;
      }
      else {
        dataPromise.promise.then(function () {
          data[name] = scope;
        });
      }
    };

    this.getValue = function (model, ignoreModel, customModelStr) {
      if (/\s\d$/.test(model)) {
        var modelIndex = model.match(/\d$/)[0];
        model = model.replace(/\s\d$/, modelIndex);
      }
      if (/\st$/.test(model)) {
        ignoreModel = true;
        model = model.replace(/\st$/, "");
      }
      model = model.split(".");
      var modelValue = data[model[0]];
      for (var key of model.slice(1)) {
        modelValue = modelValue[key];
      }
      if (typeof modelValue === "string" && /^\d/.test(modelValue)) {
        modelValue = Number(modelValue.replace(/\,/g, "."));
      }
      return modelValue;
    };

    this.arrayInRange = function (length, startInt) {
```

```
      return Array.apply(null, Array(length)).map(
        function (el, index) {
          return index + (startInt | 0);
        });
    };

    this.zebraRows = function (element, selector, styleOdd, styleEven,
siblingClass) {
      var matches = element.querySelectorAll(selector);
      for (var index = 0; index < matches.length; index++) {
        var el = matches[index];
        if (index % 2 == 0 && styleEven) {
          el.style["background-color"] = styleEven;
          if (el.nextElementSibling &&
el.nextElementSibling.classList.contains(siblingClass)) {
            el.nextElementSibling.style["background-color"] = styleEven;
          }
        }
        else if (index % 2 == 1 && styleOdd) {
          el.style["background-color"] = styleOdd;
          if (el.nextElementSibling &&
el.nextElementSibling.classList.contains(siblingClass)) {
            el.nextElementSibling.style["background-color"] = styleOdd;
          }
        }
      }
    };

    this.mapDefaultValues = function (vm, template, d) {
      var keys = Object.keys(template.models);
      keys.forEach(function (key, index) {
        var currentModel = vm.template.models[key];
        if (d) {
          if (currentModel.bind) {
            currentModel.model =
data[currentModel.bind.name][currentModel.bind.value];
            vm.$watch("template.models." + key + ".model", function (value) {
              if (typeof value === "string" &&
/^\d+?(?:[\.\,]\d{1,})?$/.test(value)) {
                data[currentModel.bind.name][currentModel.bind.value] =
Number(value.replace(/\,/g, "."));
                currentModel.error = false;
              }
              else if (currentModel.noValidate === true || typeof value ===
"number" ||
                !/text/.test(currentModel.type)|| value === "") {
                data[currentModel.bind.name][currentModel.bind.value] = value;
                currentModel.error = false;
              }
              else {
                data[currentModel.bind.name][currentModel.bind.value] = value;
                currentModel.error = true;
                //setTimeout(vm.$digest, 10);
              }
            }, true);
          }
          if (currentModel.bind2) {
            currentModel.model2 =
data[currentModel.bind2.name][currentModel.bind2.value];
            vm.$watch("template.models." + key + ".model2", function (value) {
              if (value !== undefined) {
                if (typeof value === "string" &&
/^\d+?(?:[\.\,]\d{1,})?$/.test(value)) {
```

```
                  data[currentModel.bind2.name][currentModel.bind2.value] =
Number(value.replace(/\,/g, "."));
                  currentModel.error = false;
                }
                else if (currentModel.noValidate2 === true || typeof value ===
"number" ||
                  !/text/.test(currentModel.type) || value === "") {
                  data[currentModel.bind2.name][currentModel.bind2.value] =
value;
                  currentModel.error = false;
                }
                else {
                  data[currentModel.bind2.name][currentModel.bind2.value] =
value;
                  currentModel.error = true;
                  //setTimeout(vm.$digest, 10);
                }
              }
            }, true);
          }
        }
      });
    };

    this.filterObject = function (object, propertyName, value) {
      return object.filter(el => el[propertyName] == value)[0];
    };

    this.showTooltip = function (event, html) {
      var tooltip = document.querySelector("#tooltip");
      var tooltipText = tooltip.querySelector("#tooltipText");
      tooltipText.innerHTML = html;
      tooltip.style.display = "block";
      tooltip.style.top = (((120) - 100) + window.scrollY) + "px";
      tooltip.style.left = (((window.innerWidth / 2) - 200) + window.scrollX) +
"px";
    };

    this.hideTooltip = function () {
      var tooltip = document.querySelector("#tooltip");
      var tooltipText = tooltip.querySelector("#tooltipText");
      tooltipText.innerHTML = "";
      tooltip.style.display = "none";
    };
  };

  return new Helpers();

});
```

Appendix 8

DataFactory Factory (dataFactory.js)

```
define(['app', 'json', 'helper'], function (app, data) {
   'use strict';
   return app.factory('DataFactory', function ($rootScope) {
      var pageLayout = [
         {
            id: "main",
            display: "Etusivu",
            hideOnFooter: true
         },
         {
            id: "perustiedot",
            display: "Perustiedot"
         },
         {
            id: "perussuureet",
            display: "Perussuureet"
         },
         {
            id: "kuluttajalaitteet",
            display: "Kuluttajalaitteet"
         },
         {
            id: "rakennusosat",
            display: "Rakennusosat"
         },
         {
            id: "ikkunat",
            display: "Ikkunat"
         },
         {
            id: "tilojenLammitysjarjestelma",
            display: "Tilojen lämmitysjärjestelmä"
         },
         {
            id: "lammitysjarjestelma",
            display: "Lämmitysjärjestelmä"
         },
         {
            id: "ilmaIlmaLampopumput",
            display: "Ilma-ilma Lämpöpumput"
         },
         {
            id: "lampopumppuLammitysmuotona",
            display: "Lämpöpumppu Lämmitysmuotona"
```

```
  },
  {
    id: "lisalammitysjarjestelmia",
    display: "Lisälämmitysjärjestelmiä"
  },
  {
    id: "lammonJakelunHaviot",
    display: "Lämmönjakelun häviöt"
  },
  {
    id: "jaahdytysjarjestelma",
    display: "Jäähdytysjärjestelmä"
  },
  {
    id: "ilmanvaihtojarjestelma",
    display: "Ilmanvaihtojarjestelmä"
  },
  {
    id: "ilmanvaihtojarjestelma2",
    display: "Ilmanvaihtojarjestelma 2"
  },
  {
    id: "kuukausiTalteenotto",
    display: "Kuukausi talteenotto"
  },
  {
    id: "kayttovesijarjestelma",
    display: "Käyttövesijärjestelmä"
  },
  {
    id: "kayttovesijarjestelma2",
    display: "Käyttövesijärjestelmä 2"
  },
  {
    id: "varaavienTulisijojenPolttoaineet",
    display: "Varaavien tulisijojen polttoaineet"
  },
  {
    id: "lampoarvolaskuri",
    display: "Lämpöarvolaskuri"
  },
  {
    id: "example",
    display: "Example"
  }
];
var perustiedot = {
  tilaaja: "",
```

```
        yritys: "",
        katuosoite: "",
        postinumero: "",
        postitoimipaikka: "",
        sijaintikunta: data.counties[2],
        rakennusluvanVuosi: 2015,
        valmistumisvuosi: 2016,
        asukasmaara: 2,
        saavyohyke: data.counties[2].saavyohyke,
        kayttotarkoitusluokka: data.houseTypes[5],
        kerrokset: 2,
        rakennetyyppi: data.structureTypes[4],
        lampokapasiteetti: data.structureTypes[4].value
    };
    perustiedotWatchers();
    var tilojenLammitysjarjestelma = {
        lammitysjarjestelma: data.heatingTypes[1],
        lammonjakojarjestelma: data.heatDivisionSystems[8],
        tulisijat: 1,
        tulisijojenKaytto: data.fireplaceUse[0],
        tulisijojenKaytto2: data.fireplaceUse[0].value
    };
    var kayttovesijarjestelma = {
        lampimanKayttovedenLammitysjarjestelma: data.warmWaterHeatingSystem[0],
        vedenVaraaja: data.waterHeater[8],
        vedenVaraaja2: data.waterHeater[8].varaajanLampohavio,
        vedenKierto: false,
        kierronLammityslaitteet: data.circulationHeatingSystems[1],
        kiertoputkienEristys: data.kiertoPutkienEristys[4],
        vedenJakelujarjestelma: data.vedenJakelujarjestelma[3],
        vedenJakelujarjestelma2: null
    };
    kayttovesiWatchers();
    var ilmanvaihtojarjestelma = {
        ilmanvaihtojarjestelma: "Koneellinen tulopoisto",
        ilmanvaihtokoneidenMaara: 1,
        ilmanvaihtoLammontalteenotto: true,
        LTOpoiskytkenta: true,
        tuloilmanJalkilammitys: true,
        tuloilmanJalkilammitysLampolahde: "Sähkövastus",
        LTOSuhde: 45,
        lammontalteenotonLammonvaihdin:
data.lammontalteenotonLammonvaihdin[5],
        lammontalteenotonLammonvaihdin2:
data.lammontalteenotonLammonvaihdin[5].value,
        JateilmanAlinLampotila: 5
    };
    var kuukausiTalteenotto = {
```

```
    tammikuu: true,
    tammikuu2: true,
    helmikuu: true,
    helmikuu2: true,
    maaliskuu: true,
    maaliskuu2: true,
    huhtikuu: true,
    huhtikuu2: true,
    toukokuu: true,
    toukokuu2: true,
    kesakuu: true,
    kesakuu2: true,
    heinakuu: false,
    heinakuu2: false,
    elokuu: false,
    elokuu2: false,
    syyskuu: true,
    syyskuu2: true,
    lokakuu: true,
    lokakuu2: true,
    marraskuu: true,
    marraskuu2: true,
    joulukuu: true,
    joulukuu2: true,
};
var perussuureet = {
    lammitettyNettoala: 210.5,
    rakennuksenIlmatilavuus: 521.7,
    sisalampotila: null,
    sisalampotilaIgnore: false,
    jaahdytysraja: null,
    rakennuksenIlmanvuoto: 6.0,
    ilmanvuotoIgnore: true
};
perussuureetWatchers();
var lammitysjarjestelma = {
    lammonjakojarjestelmanVuosihyoty: null,
    lammonjakojarjestelmanVuosihyoty2: false,
    lammonjakojarjestelmanApulaitteidenKulutus: null,
    lammonjakojarjestelmanApulaitteidenKulutus2: false,
    lammitysenergianTuottoTilatHyoty: null,
    lammitysenergianTuottoTilatHyoty2: false,
    tilojenLammontuottoApulaitteidenKulutus: null,
    tilojenLammontuottoApulaitteidenKulutus2: false,
    lammitysenergianTuottoVesiHyoty: null,
    lammitysenergianTuottoVesiHyoty2: false,
    kayttovesiLammontuottoKulutus: null,
    kayttovesiLammontuottoKulutus2: false,
```

```
    varaavaTulisijaHyoty: 0.6
};
lammitysjarjestelmaWatchers();
var kayttovesijarjestelma2 = {
    lampimankayttovedenLammitysenergianTarve: null,
    lampimankayttovedenLammitysenergianTarveYla: null,
    lampimankayttovedenLammitysenergianTarveELuku: null,
    lampimankayttovedenOminaiskulutus: null,
    lampimankayttovedenVarastoinninHavio: null,
    lampimankayttovedenJakeluHyoty: null,
    lampimankayttovedenKierronHavio: null
};
kayttovesi2Watchers();
var ilmanvaihtojarjestelma2 = {
    ilmanvaihtoTarkistus: false,
    ilmanvaihtokerroin: 0.4,
    ilmanvaihtoLTOpoistohyoty: 45,
    ilmanvaihtoPoistoilmavirta: null,
    ilmanvaihtoTuloilmavirta: null,
    ilmanvaihtoSPFLuku: 2.5,
    ilmanvaihtoKorvausilmavirta: null,
    tuloilmaSisaanpuhalluslampotila: 17,
    tuloilmaLammonNousu: 0.5,
    ilmanvaihtolaitoksenVuorokausiSuhde: 1.0,
    ilmanvaihtolaitoksenViikkoSuhde: 1.0,
    ilmanvaihtoLammityspatteriSuhde: 1.0
};
ilmanvaihto2Watchers();
var ikkunat = {
    ilmansuunnat: {
        koillinen: "koillinen",
        kaakko: "kaakko",
        lounas: "lounas",
        luode: "luode"
    },
    pintaAla: {
        koillinen: 21.1,
        kaakko: 7.95,
        lounas: 10.90,
        luode: 11.30
    },
    uArvo: {
        koillinen: 2.1,
        kaakko: 2.1,
        lounas: 2.1,
        luode: 2.1
    },
    auringonSateily: {
```

```
        koillinen: 0.55,
        kaakko: 0.55,
        lounas: 0.55,
        luode: 0.55
    },
    kehakerroin: {
        koillinen: 0.75,
        kaakko: 0.75,
        lounas: 0.75,
        luode: 0.75
    },
    verhokerroin: {
        koillinen: 1,
        kaakko: 1,
        lounas: 1,
        luode: 1
    },
    ymparistokerroin: {
        koillinen: 15,
        kaakko: 15,
        lounas: 15,
        luode: 15
    },
    ylapuolinenVarjostus: {
        koillinen: 0,
        kaakko: 0,
        lounas: 0,
        luode: 0
    },
    sivuvarjostus: {
        koillinen: 0,
        kaakko: 0,
        lounas: 0,
        luode: 0
    }
};
var rakennusosat = {
    ulkoseinatUlkoilmaan: 220,
    ulkoseinatUlkoilmaan2: 0.24,
    ulkoseinatUlkoilmaanValue: null,
    ylapohja: 120.5,
    ylapohja2: 0.18,
    ylapohjaValue: null,
    alapohjaMaataVasten: 120.5,
    alapohjaMaataVasten2: 0.21,
    alapohjaMaataVastenValue: null,
    alapohjaTuulettuva: 0,
    alapohjaTuulettuva2: 0,
```

```
    alapohjaTuulettuvaValue: null,
    alapohjaYhteensa: null,
    alapohjaYhteensa2: null,
    alapohjaYhteensaValue: null,
    ikkunat: null,
    ikkunat2: null,
    ikkunatValue: null,
    ovet: 9.4,
    ovet2: 0.7,
    ovetValue: null,
};
rakennusosatWatchers();
var kuluttajalaitteet = {
    rakennuksenPaivittainenKayttoaikasuhde: null,
    rakennuksenViikottainenKayttoaikasuhde: null,
    kuluttajalaitteidenOminaisteho: null,
    kuluttajalaitteidenKayttoaste: null,
    valaistuksenOminaisteho: null,
    valaistuksenKayttoaste: null,
    lampokuormaIhmisista: null
};
kuluttajalaitteetWatchers();
var lammonJakelunHaviot = {
    menoPaluuPutketPituus: 0,
    olosuhteet: data.olosuhteet[0],
    lammonJakelujarjestelmanLampohavioKylmaTila: null,
    lammonJakelujarjestelmanVarastointiLampohavio: 0
};
lammonJakelunHaviotWatchers();
var jaahdytysjarjestelma = {
    energiamuoto: data.lampopumpunLaji[0],
    lampotyyppi: null,
    jaahdytyksenNettotarve: 5000,
    kokonaishyotysuhde: 0.8
};
jaahdytysWatchers();
var lampopumppuLammitysmuotona = {
    lampopumpunLaji: "Maalämpöpumppu (MLP)",
    lampopumpunTuottamaOsuus: 88,
    lampopumpunKausisuorityskykykerroinTilat: 3.1,
    lampopumpunKausisuorityskykykerroinVesi: 2.3,
    lampopumpunNimellisteho: 10,
    lampopumpunNimellistehoDisabled: null
};
lampopumppuLammitysWatchers();
var lisalammitysjarjestelmia = {
    suuntauskerroin: 1,
    pintaAla: 10,
```

```
    energiatuotto: 156,
    pumppujenTeho: null,
    pumpunKayttoaika: 2000,
    aurinkojarjestelmaEnergia: 0,
    tuulivoimala: 0
};
lisalammitysWatchers();
var ilmaIlmaLampopumput = {
    pumppu1: 2.8,
    pumppu12: 0,
    pumppu2: 2.8,
    pumppu22: 0,
    pumppu3: 2.8,
    pumppu32: 0,
};
var toteutunutEnergiankulutus = {
    seurantavuosi: 2015,
    sahko: 42000,
    fossiiliset: 0,
    uusiutuvat: 5100
};
var varaavienTulisijojenPolttoaineet = {
    polttopuu: 2000,
    polttopuuTarvittava: null,
    polttopuuOstoenergia: null,
    polttopuuTuotto: null,
    polttopuuOstoenergia2: null,
    polttopuu2: 1000,
    pilkkeetHavu: 2000,
    pilkkeetHavuTarvittava: null,
    pilkkeetHavuOstoenergia: null,
    pilkkeetHavuTuotto: null,
    pilkkeetHavuOstoenergia2: null,
    pilkkeetHavu2: 4.0,
    pilkkeetKoivu: 2000,
    pilkkeetKoivuTarvittava: null,
    pilkkeetKoivuOstoenergia: null,
    pilkkeetKoivuTuotto: null,
    pilkkeetKoivuOstoenergia2: null,
    pilkkeetKoivu2: 3.0,
    puupelletit: 2000,
    puupelletitTarvittava: null,
    puupelletitOstoenergia: null,
    puupelletitTuotto: null,
    puupelletitOstoenergia2: null,
    puupelletit2: 800,
    polttohake: 2000,
    polttohakeTarvittava: null,
```

```
    polttohakeOstoenergia: null,
    polttohakeTuotto: null,
    polttohakeOstoenergia2: null,
    polttohake2: 4.0,
    kivihiili: 2000,
    kivihiiliTarvittava: null,
    kivihiiliOstoenergia: null,
    kivihiiliTuotto: null,
    kivihiiliOstoenergia2: null,
    kivihiili2: 500,
    palaturve: 2000,
    palaturveTarvittava: null,
    palaturveOstoenergia: null,
    palaturveTuotto: null,
    palaturveOstoenergia2: null,
    palaturve2: 1000,
    puubriketit: 2000,
    puubriketitTarvittava: null,
    puubriketitOstoenergia: null,
    puubriketitTuotto: null,
    puubriketitOstoenergia2: null,
    puubriketit2: 800
};
varaavienTulisijojenWatchers();
var lampoarvolaskuri = {
    raskasOljy: 11400,
    raskasOljy2: 1000,
    raskasOljyMaara: null,
    raskasOljyMaara2: null,
    kevytOljy: 29363,
    kevytOljy2: 7000,
    kevytOljyMaara: null,
    kevytOljyMaara2: null,
    maakaasu: 20000,
    maakaasu2: 2000,
    maakaasuMaara: null,
    maakaasuMaara2: null,
    polttopuu: 2500,
    polttopuu2: 1500,
    polttopuuMaara: null,
    polttopuuMaara2: null,
    pilkkeetHavu: 5200,
    pilkkeetHavu2: 4.0,
    pilkkeetHavuMaara: null,
    pilkkeetHavuMaara2: null,
    pilkkeetKoivu: 3400,
    pilkkeetKoivu2: 2.0,
    pilkkeetKoivuMaara: null,
```

```
    pilkkeetKoivuMaara2: null,
    puupelletit: 7050,
    puupelletit2: 1500,
    puupelletitMaara: null,
    puupelletitMaara2: null,
    polttohake: 2700,
    polttohake2: 3.0,
    polttohakeMaara: null,
    polttohakeMaara2: null,
    kivihiili: 2000,
    kivihiili2: 500,
    kivihiiliMaara: null,
    kivihiiliMaara2: null,
    palaturve: 2640,
    palaturve2: 800,
    palaturveMaara: null,
    palaturveMaara2: null,
    puubriketit: 7200,
    puubriketit2: 1500,
    puubriketitMaara: null,
    puubriketitMaara2: null
};
lampoarvoWatchers();
var example = {
    multiplier: 1,
    multiplied: null
};
exampleWatchers();

return {
    pageLayout: pageLayout,
    perustiedot: perustiedot,
    tilojenLammitysjarjestelma: tilojenLammitysjarjestelma,
    kayttovesijarjestelma: kayttovesijarjestelma,
    ilmanvaihtojarjestelma: ilmanvaihtojarjestelma,
    kuukausiTalteenotto: kuukausiTalteenotto,
    perussuureet: perussuureet,
    lammitysjarjestelma: lammitysjarjestelma,
    kayttovesijarjestelma2: kayttovesijarjestelma2,
    ilmanvaihtojarjestelma2: ilmanvaihtojarjestelma2,
    ikkunat: ikkunat,
    rakennusosat: rakennusosat,
    kuluttajalaitteet: kuluttajalaitteet,
    lammonJakelunHaviot: lammonJakelunHaviot,
    jaahdytysjarjestelma: jaahdytysjarjestelma,
    lampopumppuLammitysmuotona: lampopumppuLammitysmuotona,
    lisalammitysjarjestelmia: lisalammitysjarjestelmia,
    ilmaIlmaLampopumput: ilmaIlmaLampopumput,
```

```
        toteutunutEnergiankulutus: toteutunutEnergiankulutus,
        varaavienTulisijojenPolttoaineet: varaavienTulisijojenPolttoaineet,
        lampoarvolaskuri: lampoarvolaskuri,
        example: example
    };
    /* Functional calculations redacted */
    function exampleWatchers(){
        $rootScope.$watch(
            function () {
                return {
                    multiplier: example.multiplier,
                    inhabitants: perustiedot.asukasmaara
                };
            },
            function (values) {
                example.multiplied = values.multiplier * values.inhabitants;
            },
            true
        );
    }
  });
});
```

Appendix 9

JSON file parser and distributor Module (jsons.js)

```
define(['jsonr!../JSON/rakennetyypit.json', 'jsonr!../JSON/kunnat.json',
  'jsonr!../JSON/kayttotarkoitusluokat.json',
'jsonr!../JSON/lammitusmuodot.json',
  'jsonr!../JSON/lammonjakojarjestelmat.json',
'jsonr!../JSON/varaajanarvot.json',
  'jsonr!../JSON/lampimankayttovedenlammitysjarjestelma.json',
  'jsonr!../JSON/kierronLammityslaitteet.json',
'jsonr!../JSON/kiertoPutkienEristys.json',
  'jsonr!../JSON/vedenJakelujarjestelma.json',
'jsonr!../JSON/lammontalteenotonLammonvaihdin.json',
  'jsonr!../JSON/olosuhteet.json', 'jsonr!../JSON/lampopumpunLaji.json',
  'jsonr!../JSON/polttoaineidenLampoarvot.json'],
  function (structureTypes, counties, houseTypes, heatingTypes,
heatDivisionSystems, waterHeater,
    warmWaterHeatingSystem, circulationHeatingSystems, kiertoPutkienEristys,
polttoaineidenLampoarvot) {
    var keys = ["structureTypes", "counties", "houseTypes", "heatingTypes",
"heatDivisionSystems",
      "waterHeater", "warmWaterHeatingSystem", "circulationHeatingSystems",
"kiertoPutkienEristys",
      "vedenJakelujarjestelma", "lammontalteenotonLammonvaihdin", "olosuhteet",
"lampopumpunLaji",
      "polttoaineidenLampoarvot"];
    var jsonObjects = {};
    for (let i = 0; i < keys.length; i++) {
      jsonObjects[keys[i]] = arguments[i];
    }
    jsonObjects.fireplaceUse = [
      { name: "Standardikäyttö", value: 2000 },
      { name: "Oma Valinta" }
    ];
    return jsonObjects;
  });
```

Appendix 10

Default template HTML (default.html)

```
<div id="{{template.name}}" class="inputTable">
   <table cellspacing="0" cellpadding="0">
    <thead>
     <tr>
      <th colspan="4">
      {{template.readable}}
      </th>
      <td>
       lähde
      </td>
     </tr>
    </thead>
    <tbody id="repeatRows" ng-repeat="key in template.models" ng-
switch="key.type" tables>
    </tbody>
   </table>
</div>
```

Appendix 11

Main html to write all the rows to a table (tables.html)

```
<tr class="row {{key.error?'errorRow':''}}" ng-switch-when="select">
  <td class="basicName">
    {{key.header}}
  </td>
  <td colspan="3">
    <span class="inputHolder">
      <select class="dropdownInput" id="{{key.name}}" ng-model="key.model"
        ng-disabled={{key.disabled}} ng-mouseover="key.hover($event)"
        ng-options={{key.options}} ng-change="key.onChange();
$root.$broadcast('calculate')"
        ng-click="key.onClick()" ng-hide="key.hide"
        ng-mouseleave="key.leave()">
      </select>
    </span>
  </td>
  <td class="info" ng-mouseleave="hideTooltip()" ng-
mousemove="showTooltip($event,key.source)">
    ?
  </td>
</tr>
<tr class="row {{key.error?'errorRow':''}}" ng-switch-when="selectWide">
  <td class="basicName" colspan="4">
    {{key.header}}
  </td>
  <td rowspan="2" class="info" ng-mouseleave="hideTooltip()" ng-
mousemove="showTooltip($event,key.source)">
    ?
  </td>
</tr>
<tr class="rowCont  {{key.error?'errorRow':''}}" ng-switch-when="selectWide">
  <td colspan="4">{{key.prefix}}
    <at>{{key.prefixSub}}</at>
    <select class="dropdownInput" id="{{key.name}}" ng-model="key.model" ng-
disabled={{key.disabled}} ng-mouseover="key.hover()"
      ng-options={{key.options}} ng-change="key.onChange();
$root.$broadcast('calculate')" ng-click="key.onClick()" ng-hide="key.hide" ng-
mouseleave="key.leave()">
    </select>{{key.suffix}}
    <at>{{key.suffixSub}}</at>
  </td>
</tr>
<tr class="row {{key.error?'errorRow':''}}" ng-switch-when="text">
  <td class="basicName">
    {{key.header}}
  </td>
  <td class="prefix">{{key.prefix}}
    <at>{{key.prefixSub}}</at>
  </td>
  <td class="textInputHolder"><input ng-blur="$root.$broadcast('calculate')"
type="text" class="textInput" id="{{key.name}}" ng-disabled={{key.disabled}} ng-
mouseover="key.hover($event)" ng-change="key.onChange()"
      ng-model="key.model" ng-click="key.onClick()" ng-hide="key.hide" ng-
mouseleave="key.leave()"></td>
  <td class="suffix">{{key.suffix}}
    <at>{{key.suffixSub}}</at>
  </td>
  <td class="info" ng-mouseleave="hideTooltip()" ng-
mousemove="showTooltip($event,key.source)">
    ?
  </td>
```

```
</tr>
<tr class="row double {{key.error?'errorRow':''}}" ng-switch-when="select+text">
  <td rowspan="2" class="basicName">
    {{key.header}}
  </td>
  <td colspan="3">
    <span class="inputHolder">
    <select class="dropdownInput" id="{{key.name}}" ng-model="key.model"
      ng-disabled={{key.disabled}} ng-mouseover="key.hover($event)"
      ng-options={{key.options}} ng-change="key.onChange();
$root.$broadcast('calculate')"
      ng-click="key.onClick()" ng-hide="key.hide"
      ng-mouseleave="key.leave()">
    </select>
    </span>
  </td>
  <td rowspan="2" class="info" ng-mouseleave="hideTooltip()" ng-
mousemove="showTooltip($event,key.source)">
    ?
  </td>
</tr>
<tr class="rowCont {{key.error?'errorRow':''}}" ng-switch-when="select+text">
  <td class="prefix">{{key.prefix2}}
    <at>{{key.prefixSub2}}</at>
  </td>
  <td class="textInputHolder"><input ng-blur="$root.$broadcast('calculate')"
type="text" class="textInput" id="{{key.name2}}" ng-disabled={{key.disabled2}}
ng-mouseover="key.hover2($event)"
      ng-change="key.onChange2()" ng-model="key.model2" ng-click="key.onClick2()"
ng-hide="key.hide2" ng-mouseleave="key.leave2()"></td>
  <td class="suffix">{{key.suffix2}}
    <at>{{key.suffixSub2}}</at>
  </td>
</tr>
<tr class="row {{key.error?'errorRow':''}}" ng-switch-when="switch">
  <td colspan="1" class="basicName">
    {{key.header}}
  </td>
  <td colspan="3">
    <span class="inputHolder">
   <label class="switch">
      <input type="checkbox" id="{{key.name}}" ng-model="key.model" ng-
disabled={{key.disabled}}
      ng-mouseover="key.hover()" ng-change="key.onChange()"
      ng-click="key.onClick(); $root.$broadcast('calculate')" ng-hide="key.hide"
ng-mouseleave="key.leave()">
      <span class="slider"></span>
    </label>
    </span>
  </td>
  <td class="info" ng-mouseleave="hideTooltip()" ng-
mousemove="showTooltip($event,key.source)">
    ?
  </td>
</tr>
<tr class="row double  {{key.error?'errorRow':''}}" ng-switch-when="switch+text">
  <td class="basicName">
    {{key.header}}
  </td>
  <td colspan="3">
    <span class="inputHolder">
    </at><label class="switch">
      <input type="checkbox" id="omaValinta{{key.name2}}" ng-model="key.model2"
ng-disabled={{key.disabled2}}
```

```
      ng-mouseover="key.hover2()" ng-change="key.onChange2()"
      ng-click="key.onClick2(); $root.$broadcast('calculate')" ng-
hide="key.hide2" ng-mouseleave="key.leave2()">
      <span class="slider"></span>
    </label>
    </span>
  </td>
  <td rowspan="2" class="info" ng-mouseleave="hideTooltip()" ng-
mousemove="showTooltip($event,key.source)">
    ?
  </td>
</tr>
<tr class="rowCont  {{key.error?'errorRow':''}}" ng-switch-when="switch+text">
  <td class="basicName">{{key.header2}}</td>
  <td class="prefix">{{key.prefix}}
    <at>{{key.prefixSub}}</at>
  </td>
  <td class="textInputHolder"><input ng-blur="$root.$broadcast('calculate')"
type="text" class="textInput" id="{{key.name}}" ng-disabled={{key.disabled}} ng-
mouseover="key.hover($event)"
      ng-change="key.onChange()" ng-model="key.model" ng-click="key.onClick()"
ng-hide="key.hide" ng-mouseleave="key.leave()"></td>
  <td class="suffix">{{key.suffix}}
    <at>{{key.suffixSub}}</at>
  </td>
</tr>
<tr class="row {{key.error?'errorRow':''}}" ng-switch-when="separator">
  <td colspan="9">{{key.header}}</td>
</tr>
<tr class="row {{key.error?'errorRow':''}}" ng-switch-when="value">
  <td colspan="1" class="basicName">{{key.header}}</td>
  <td colspan="3" class="staticValue">{{key.model}}</td>
  <td></td>
</tr>
<tr class="row {{key.error?'errorRow':''}}" ng-switch-when="textWide">
  <td class="basicName">
    {{key.header}}
  </td>
  <td colspan="3" class="textInputHolderWide"><input ng-
blur="$root.$broadcast('calculate')" type="text" class="textInputFull"
id="{{key.name}}" ng-disabled={{key.disabled}} ng-mouseover="key.hover($event)"
ng-change="key.onChange()"
      ng-model="key.model" ng-click="key.onClick()" ng-hide="key.hide" ng-
mouseleave="key.leave()"></td>
  <td class="info" ng-mouseleave="hideTooltip()" ng-
mousemove="showTooltip($event,key.source)">
    ?
  </td>
</tr>
```

Appendix 12

Footer HTML (footer.html)

```
<div class="footer row" ng-show="showFooter">
    <span class="footButton" id="prevButton" ng-click='previous()' ng-
style="{'visibility': hasPrevious?'visible':'hidden'}">
        <span>{{previousPage}}</span>
    </span>
    <span class="footButton" id="nextButton" ng-click='next()' ng-
style="{'visibility': hasNext?'visible':'hidden'}">
        <span>{{nextPage}}</span>
    </span>
</div>
```

Appendix 13

Front page HTML (main.html)

```
<h4>Lataa vanha laskelma</h4>
    <input type="button" value="Tuo koko laskelma" class="button-norm"
onclick="document.querySelector('#filereadCalc').click()">
    <input type="file" id="filereadCalc" style="display:none" id="fileUpload2"
fileread="datas" isdata="'true'" accept="*.json">
    <input type="button" value="Tuo yksittäinen korjaus" class="button-norm"
onclick="document.querySelector('#filereadFix').click()">
    <input type="file" id="filereadFix" style="display:none" id="fileUpload"
fileread="uploadedFile" accept="*.json"><br><br>
<input type="button" value="Lataa koko laskelma" class="button-norm" ng-
click="save()"></span>
<input type="button" value="Lisää uusi korjaus" class="button-norm" ng-
click="addFix()"></span>
<table>
    <tr class="fileList {{currentFile===file?'current':''}}" ng-repeat="file in
datas.files">
        <td class="fileName">{{file.name}}</td>
        <td class="fileName eLuku">E-luku:
{{Math.ceil(file.value.eLukuSumma.summa.eLuku)}}</td>
        <td class="buttonTd"><input type="button" value="Nimeä uudelleen"
class="button-norm" ng-click="renameFile(file)"></td>
        <td class="buttonTd"><input type="button" value="Valitse" class="button-
norm" ng-click="swapFile(file)"></td>
        <td class="buttonTd"><input type="button" value="Lataa tiedosto"
class="button-norm" ng-click="save(file)"></td>
        <td class="deleteTd"><span ng-if="currentFile!==file" class="removeX" ng-
click="removeFile(file)">X</span></td>
    </tr>
</table>
```

Appendix 14

Calculation component HTML (laskenta.html)

```
<table class="laskentaTable">
    <tr ng-repeat="luku in calcs">
        <th class="laskentaHeader">{{luku.header}}</th>
        <td class="laskentaValue">{{luku.value}}</td>
        <td class="laskentaSuffix">{{luku.suffix}}</td>
    </tr>
</table>
```

Appendix 15

Side navigation bar HTML (sidenavbar.html)

```
<ul class="sidenav {{smallScr?'smallScr':''}}" ng-show="showSideNav">
  <li  ng-repeat="page in pages track by page.id"
class="{{smallScr?'smallScr':''}} {{page.hasErrors===true?'errorRow':''}}">
    <a ui-sref="{{page.id}}" ng-click='click($event)'>{{page.display}}</a>
  </li>
  <li ng-show="smallScr" id="hideSideNav" class="{{smallScr?'smallScr':''}}">
    <a ng-click="toggleSideNav()">≡</a>
  </li>
</ul>
<div id="showSideNav" ng-show="!showSideNav" ng-click="toggleSideNav()">
  <span class="hamburger">≡</span>
</div>
```

Appendix 16

IlmaIlmaLampapumput HTML (ilmaIlmaLampopumput.html)

```
<table cellspacing="0" cellpadding="0">
  <thead>
    <tr>
      <th>Ilma-ilma lämpöpumput tilojen lämmityksessä.</th>
      <th colspan="2">SPF-luku</th>
      <th colspan="2">Tuotto tiloihin</th>
      <th>Lähde</th>
    </tr>
  </thead>
  <tr class="row {{key.error?'errorRow':''}}" id="repeatRows" ng-repeat="key in
template.models">
    <td class="basicName">{{key.header}}</td>
    <td>
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput" ng-model="key.model" ng-disabled={{key.disabled}} ng-
mouseover="key.hover()"
        ng-change="key.onChange()" ng-click="key.onClick()" ng-hide="key.hide"
ng-mouseleave="key.leave()">
    </td>
    <td class="suffix">kW/(m3/s)</td>
    <td>
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput" ng-model="key.model2" ng-disabled={{key.disabled2}} ng-
mouseover="key.hover2()"
        ng-change="key.onChange()" ng-click="key.onClick2()" ng-hide="key.hide2"
ng-mouseleave="key.leave2()">
    </td>
    <td class="suffix">kWh/a</td>
    <td class="info" ng-mouseleave="hideTooltip()" ng-
mousemove="showTooltip($event,key.source)">
      ?
    </td>
  </tr>
  </tbody>
</table>
```

Appendix 17

Kuukausi-talteenotto HTML (kuukausi-talteenotto.html)

```
<table cellspacing="0" cellpadding="0">
  <thead>
    <tr>
      <th colspan="2">
        Ilmanvaihdon lämmöntalteenoton ja jälkilämmityksen kuukausiaikataulu
      </th>
      <th>Lämmöntalteenotto</th>
      <th>Jälkilämmitys</th>
    </tr>
  </thead>
  <tr class="row {{key.error?'errorRow':''}}" id="repeatRows" ng-repeat="key in
template.models">
    <td colspan="2">{{key.header}}</td>
    <td>
      <label class="switch">
          <input ng-blur="$root.$broadcast('calculate')"type="checkbox"
id="{{key.name}}" ng-model="key.model" ng-disabled={{key.disabled}} ng-
mouseover="key.hover()" ng-change="key.onChange()"
          ng-click="key.onClick()" ng-hide="key.hide" ng-
mouseleave="key.leave()">
          <span class="slider"></span>
        </label>
    </td>
    <td>
      <label class="switch">
        <input ng-blur="$root.$broadcast('calculate')"type="checkbox"
id="{{key.name}}" ng-model="key.model2" ng-disabled={{key.disabled2}} ng-
mouseover="key.hover2()" ng-change="key.onChange2()"
          ng-click="key.onClick2()" ng-hide="key.hide2" ng-
mouseleave="key.leave2()">
          <span class="slider"></span>
        </label>
    </td>
  </tr>
  </tbody>
  <!--<tr class="row" ng-show="!template.extraInfo.visible" ng-
click="template.extraInfo.show()"><td colspan="3">V</td></tr>
    <tr class="row" ng-show="false"><!--this is here to keep the zebra rows
correct-></tr>
    </table>
    <table  cellspacing="0" cellpadding="0" ng-show="template.extraInfo.visible"
class="extraInfo">
    <tbody>
    </tbody>-->
</table>
```

Appendix 18

Lampoarvolaskuri HTML (lampoarvolaskuri.html)

```
<table cellspacing="0" cellpadding="0">
  <thead>
    <tr>
      <th colspan="9"Lämpöarvolaskuri</th>
    </tr>
    <tr>
      <th></th>
      <th class="heatCalc">Määrä</th>
      <th class="heatCalc">Määrä</th>
      <th class="heatCalc"></th>
      <th class="heatCalc">Määrä</th>
      <th class="heatCalc"></th>
      <th class="heatCalc">Määrä</th>
    </tr>
    <tr>
      <th>Polttoaine</th>
      <th>kWh</th>
      <th></th>
      <th>Yksikkö</th>
      <th></th>
      <th>Yksikkö</th>
      <th>kWh</th>
    </tr>
  </thead>
  <tr class="row {{key.error?'errorRow':''}}" id="repeatRows" ng-repeat="key in
template.models">
    <td class="basicName">{{key.header}}</td>
    <td class="heatCalc">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput heatCalc" ng-model="key.model" ng-disabled={{key.disabled}} ng-
mouseover="key.hover()"
        ng-change="key.onChange()" ng-click="key.onClick()" ng-hide="key.hide"
ng-mouseleave="key.leave()">
    </td>
    <td class="heatCalc">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput heatCalc" ng-model="key.maara" disabled ng-
mouseover="key.hover2()"
        ng-change="key.onChange()" ng-click="key.onClick2()" ng-hide="key.hide2"
ng-mouseleave="key.leave2()">
    </td>
    <td class="suffix">{{key.suffix}}</td>
    <td class="heatCalc">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput heatCalc" ng-model="key.model2" ng-disabled={{key.disabled}} ng-
mouseover="key.hover()"
        ng-change="key.onChange()" ng-click="key.onClick()" ng-hide="key.hide"
ng-mouseleave="key.leave()">
    </td>
    <td class="suffix">{{key.suffix2}}</td>
    <td class="heatCalc">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput heatCalc" ng-model="key.maara2" disabled ng-
mouseover="key.hover()"
        ng-change="key.onChange()" ng-click="key.onClick()" ng-hide="key.hide"
ng-mouseleave="key.leave()">
    </td>
  </tr>
  </tbody>
</table>
```

Appendix 19

Perustiedot HTML (perustiedot.html)

```
<table id="perustiedot" class="inputTable" cellspacing="0" cellpadding="0">
  <thead>
   <tr>
    <th colspan="4">
    Perustiedot
    </th>
    <td>
     lähde
    </td>
   </tr>
  </thead>
  <tbody id="repeatRows" ng-repeat="key in template.models" ng-switch="key.type"
tables>
  </tbody>
  <tr class="row" ng-show="!template.extraInfo.visible" ng-
click="template.extraInfo.show()"><td colspan="5">V</td></tr>
  <tr class="row" ng-show="false"><!--this is here to keep the zebra rows
correct--></tr>
</table>
  <table cellspacing="0" cellpadding="0" ng-show="template.extraInfo.visible"
class="extraInfo">
<tbody>
  <tr class="row"><th colspan="5">Rakennuksen sisäpuolisen tehollisen
lämpökapasiteetin ominaisarvoja eri rakennustyypeissä
kalusteineen</th><th>C<sub>rak omin<sub></th></tr>
  <tr class="row"><th>Rakennetyyppi</th><th colspan="4">US on ulkoseinä, VS
väliseinä, VP välipohja, YP yläpohja ja AP on alapohja</th><th>Wh/(m²
K)</th></tr>
  <tr class="row"><td>Pientalot, kevytrakenteinen</td><td colspan="4">US, VS, YP,
AP kevyitä rankarakenteita</td><td>40</td></tr>
  <tr class="row"><td>Pientalot, keskiraskas I</td><td colspan="4">US, VS, YP
kevyitä rankarakenteita, AP betoni</td><td>70</td></tr>
  <tr class="row"><td>Pientalot, keskiraskas II</td><td colspan="4">US harkko tai
massiivihirsi, VS, YP kevyitä rankarakenteita, AP betoni</td><td>110</td></tr>
  <tr class="row"><td>Pientalot, raskasrakenteinen</td><td colspan="4">US betoni
tai tiili, VS harkko tai tiili, YP, AP betoni</td><td>400</td></tr>
  <tr class="row"><td>Asuinkerrostalot, kevytrakenteinen</td><td colspan="4">US,
VS, VP kevyitä rankarakenteita, AP betoni</td><td>40</td></tr>
  <tr class="row"><td>Asuinkerrostalot, keskiraskas</td><td colspan="4">US
kevyitä rankarakenteita, VS kevyitä rankarakenteita tai betoni, VP betoni, AP
betoni</td><td>160</td></tr>
  <tr class="row"><td>Asuinkerrostalot, raskasrakenteinen</td><td colspan="4">US
betoni, VS harkko tai betoni, VP betoni, AP betoni</td><td>40</td></tr>
  <tr class="row"><td>Toimistorakennukset, kevytrakenteinen</td><td
colspan="4">US, VS, VP kevyitä rankarakenteita, AP betoni</td><td>70</td></tr>
  <tr class="row"><td>Toimistorakennukset, keskiraskas</td><td colspan="4">US
kevyitä rankarakenteita, VS kevyitä rankarakenteita tai betoni, VP betoni, AP
betoni</td><td>110</td></tr>
  <tr class="row"><td>Toimistorakennukset, raskasrakenteinen</td><td
colspan="4">US betoni, VS harkko tai betoni, VP betoni, AP
betoni</td><td>160</td></tr>
  <tr class="row" ng-click="template.extraInfo.hide()"><td
colspan="6">^</td></tr>
</tbody>
</table>
```

Appendix 20

Rakennusosat HTML (rakennusosat.html)

```
<table cellspacing="0" cellpadding="0">
  <thead>
    <tr>
      <th>Rakennusosat</th>
      <th>A m²</th>
      <th>U W/(m² K)</th>
      <th>T<sub>u</sub>, laskennan käyttämä ulkolämpötila °C</th>
      <th>UA W/K</th>
      <th>Lähde</th>
    </tr>
  </thead>
  <tr class="row {{key.error?'errorRow':''}}" id="repeatRows" ng-repeat="key in
template.models">
    <td>{{key.header}}</td>
    <td>
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput" ng-model="key.model" ng-disabled={{key.disabled}} ng-
mouseover="key.hover()" ng-change="key.onChange()"
          ng-click="key.onClick()" ng-hide="key.hide" ng-
mouseleave="key.leave()">
    </td>
    <td>
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput" ng-model="key.model2" ng-disabled={{key.disabled2}} ng-
mouseover="key.hover2()" ng-change="key.onChange()"
          ng-click="key.onClick2()" ng-hide="key.hide2" ng-
mouseleave="key.leave2()">
    </td>
    <td>{{key.tila}}</td>
    <td>
      <input ng-blur="$root.$broadcast('calculate')"type="text" class="textInput"
disabled ng-model="key.value" ng-init="key.onChange()">
    </td>
   <td class="info" ng-mouseleave="hideTooltip()" ng-
mousemove="showTooltip($event,key.source)">
    ?
   </td>
  </tr>
  </tbody>
</table>
```

Appendix 21

VaraavienTulisijojenPolttoaineet HTML (varaavienTulisijojenPolttoaineet.html)

```
<table cellspacing="0" cellpadding="0">
  <thead>
    <tr>
      <th colspan="9">Varaavien tulisijojen polttoaineen laskenta</th>
    </tr>
    <tr>
      <th></th>
      <th class="fireplace">Lämmitys-energian tarve</th>
      <th class="fireplace">Tarvittava polttoaineen määrä</th>
      <th class="fireplace"></th>
      <th class="fireplace">Osto-energian kulutus</th>
      <th class="fireplace">Käytettävän polttoaineen määrä</th>
      <th class="fireplace"></th>
      <th class="fireplace">Lämmitys-energian tuotto</th>
      <th class="fireplace">Osto-energian kulutus</th>
    </tr>
    <tr>
      <th>Polttoaine</th>
      <th>kWh</th>
      <th></th>
      <th>Yksikkö</th>
      <th>kWh</th>
      <th></th>
      <th>Yksikkö</th>
      <th>kWh</th>
      <th>kWh</th>
    </tr>
  </thead>
  <tr class="row {{key.error?'errorRow':''}}" id="repeatRows" ng-repeat="key in
template.models">
    <td class="basicName">{{key.header}}</td>
    <td class="fireplace">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput fireplace" ng-model="key.model" ng-disabled={{key.disabled}} ng-
mouseover="key.hover()"
        ng-change="key.onChange()" ng-click="key.onClick()" ng-hide="key.hide"
ng-mouseleave="key.leave()">
    </td>
    <td class="fireplace">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput fireplace" ng-model="key.tarvittava" disabled ng-
mouseover="key.hover2()"
        ng-change="key.onChange()" ng-click="key.onClick2()" ng-hide="key.hide2"
ng-mouseleave="key.leave2()">
    </td>
    <td class="fireplace">{{key.suffix}}</td>
    <td class="fireplace">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput fireplace" ng-model="key.ostoenergia" disabled ng-
mouseover="key.hover()"
        ng-change="key.onChange()" ng-click="key.onClick()" ng-hide="key.hide"
ng-mouseleave="key.leave()">
    </td>
    <td class="fireplace">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput fireplace" ng-model="key.model2" ng-disabled={{key.disabled}}
ng-mouseover="key.hover()"
        ng-change="key.onChange()" ng-click="key.onClick()" ng-hide="key.hide"
ng-mouseleave="key.leave()">
    </td>
```

```
    <td class="fireplace">{{key.suffix2}}</td>
    <td class="fireplace">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput fireplace" ng-model="key.tuotto" disabled ng-
mouseover="key.hover()"
        ng-change="key.onChange()" ng-click="key.onClick()" ng-hide="key.hide"
ng-mouseleave="key.leave()">
    </td>
    <td class="fireplace">
      <input ng-blur="$root.$broadcast('calculate')"type="text" id="{{key.name}}"
class="textInput fireplace" ng-model="key.ostoenergia2" disabled ng-
mouseover="key.hover()"
        ng-change="key.onChange()" ng-click="key.onClick()" ng-hide="key.hide"
ng-mouseleave="key.leave()">
    </td>
  </tr>
  </tbody>
</table>
```

Appendix 22

Directives index (index.js)

```
define([
    './table-template',
    './sidenav',
    './laskenta',
    './footer',
    './filereader'
], function () {});
```

Appendix 23

Filereader Directive (filereader.js)

```
define(['./module', 'helper'], function (directives, helper) {
    'use strict';
    directives.directive("fileread", ["$timeout", "$rootScope", function
($timeout, $rootScope) {
        return {
            scope: {
                fileread: "=",
                isdata: "<"
            },
            link: function ($scope, element, attrs) {
                element.bind("change", function (changeEvent) {
                    var reader = new FileReader();
                    reader.onload = function (evt) {
                        $scope.$apply(function () {
                            if ($scope.isdata !== "true") {
                                $scope.fileread = {
                                    result: evt.target.result,
                                    name: changeEvent.target.files[0].name
                                };
                            }
                            else{
                                $scope.fileread = JSON.parse(evt.target.result);
                                helper.changeFiles($scope.fileread);
                                $rootScope.$broadcast("calculate");
                            }
                            element[0].value = "";
                        });
                    };
                    reader.readAsText(changeEvent.target.files[0]);
                });
            }
        };
    }]);
});
```

Appendix 24

Footer Directive (footer.js)

```
define(['./module', 'helper', 'laskenta'], function (directives, helper,
laskenta) {
    'use strict';
    directives.directive('footerDirective', ["$timeout", "DataFactory", function
($timeout, DataFactory) {
        return {
            restrict: "AE",
            templateUrl: "templates/footer.html",
            controller: function ($scope, $timeout, $rootScope, $state) {
                var vm = $scope;
                var pages = DataFactory.pageLayout.filter(function (page) {
                    return page.hideOnFooter !== true;
                });
                vm.eLuku;
                vm.showFooter;
                var currentPage;
                vm.nextPage = "next";
                vm.previousPage = "prev";
                $rootScope.$on("$stateChangeSuccess", function () {
                    $timeout(function () {
                        currentPage = pages.filter(function (obj) {
                            return obj.id === location.hash.match(/\/(.+)$/)[1];
                        })[0];
                        if (pages.indexOf(currentPage) === -1) {
                            vm.showFooter = false;
                        }
                        else {
                            vm.showFooter = true;
                        }
                        vm.hasNext = pages.indexOf(currentPage) < pages.length -
1;
                        vm.hasPrevious = pages.indexOf(currentPage) > 0;
                        if (vm.hasNext) {
                            vm.nextPage = pages[pages.indexOf(currentPage) +
1].display;
                        }
                        if (vm.hasPrevious) {
                            vm.previousPage = pages[pages.indexOf(currentPage) -
1].display;
                        }
                        $rootScope.$broadcast("calculate");
                    });
                });
                vm.next = function () {
                    $state.go(pages[pages.indexOf(currentPage) + 1].id);
                };
                vm.previous = function () {
                    $state.go(pages[pages.indexOf(currentPage) - 1].id);
                };
            }
        };
    }]);
});
```

Appendix 25

Side navigation bar Directive (sidenav.js)

```
define(['./module', 'helper'], function (directives, helper) {
  'use strict';
  directives.directive('sidenav', ["$timeout", "$window", "DataFactory",
"$rootScope",
   function ($timeout, $window, DataFactory, $rootScope) {
    return {
      restrict: "AE",
      templateUrl: "templates/sidenavbar.html",
      controller: function ($scope, $timeout, $window) {
        var vm = $scope;
        vm.showSideNav = true;
        vm.toggleSideNav = function () {
          vm.showSideNav = !vm.showSideNav;
        };
        $rootScope.$watch(
          function () {
            return DataFactory.pageLayout;
          },
          function (layout) {
            vm.pages = layout;
          },
          true
        );
        vm.click = function () {
          if (vm.smallScr) {
            vm.showSideNav = false;
          }
        };
        var checkSize = function () {
          if ($window.innerWidth < 1420 && $window.innerWidth > 1150) {
            document.querySelector(".holder").style["margin-left"] = "300px";
            vm.smallScr = false;
            vm.showSideNav = true;
          }
          else if ($window.innerWidth <= 1150) {
            vm.showSideNav = false;
            vm.smallScr = true;
            document.querySelector(".holder").style["margin-left"] = "auto";
          }
          else {
            document.querySelector(".holder").style["margin-left"] = "auto";
            vm.smallScr = false;
            vm.showSideNav = true;
          }
        };
        checkSize();
        angular.element($window).bind('resize', function () {
          checkSize();
          $scope.$digest();
        });
        $timeout(helper.zebraRows.bind(null, document, "li", "#EEEEEE",
"#EEEEEE", "rowCont"));
      }
    };
  }]);
});
```

Appendix 26

Calculations Directive (laskenta.js)

```
define(['./module', 'laskenta', 'helper'], function (directives, laskenta,
helper) {
    'use strict';
    directives.directive('laskentaDirective', ['$timeout', "DataFactory",
"$rootScope",
        function ($timeout, DataFactory, $rootScope) {
            return {
                restrict: 'AE',
                templateUrl: "templates/laskenta.html",
                scope: true,
                controller: controller
            };
        }
    ]);
    var controller = function ($scope, $timeout, DataFactory, $rootScope) {
        var vm = $scope;
        vm.eLuku;
        vm.calcs = [
            {
                header: "E-luku",
                value: 0,
                suffix: "kWhE/m²vuosi"
            },
            {
                header: "Laskennallinen",
                value: 0,
                suffix: "kWh/m²vuosi"
            },
            {
                header: "Toteutunut",
                value: 0,
                suffix: "kWh/m²vuosi"
            }
        ];
        vm.$on("calculate", function (values) {
            laskenta.handleJSONs().then(function () {
                vm.calcs[0].value =
Math.ceil(DataFactory.eLukuSumma.summa.eLuku);
                vm.calcs[1].value =
Math.ceil(DataFactory.eLukuSumma.summa.kulutusAla);
                vm.calcs[2].value = Math.ceil(
                    (
                        DataFactory.toteutunutEnergiankulutus.sahko +
                        DataFactory.toteutunutEnergiankulutus.fossiiliset +
                        DataFactory.toteutunutEnergiankulutus.uusiutuvat
                    ) / DataFactory.perussuureet.lammitettyNettoala
                );
                $rootScope.$digest();
            });
        });
    };
});
```

Appendix 27

Controllers index (index.js)

```
define([
    './mainController',
    './perustiedot',
    './tilojen-lammitysjarjestelma',
    './kayttovesijarjestelma',
    './ilmanvaihtojarjestelma',
    './kuukausi-talteenotto',
    './perussuureet',
    './lammitysjarjestelma',
    './kayttovesijarjestelma2',
    './ilmanvaihtojarjestelma2',
    './ikkunat',
    './rakennusosat',
    './kuluttajalaitteet',
    './lammonJakelunHaviot',
    './jaahdytysjarjestelma',
    './lampopumppuLammitysmuotona',
    './lisalammitysjarjestelmia',
    './ilmaIlmaLampopumput',
    './toteunutEnergiankulutus',
    './varaavienTulisijojenPolttoaineet',
    './lampoarvolaskuri',
    './main',
    './example'
], function () {});
```

Appendix 28

Example controller (from instructions, example.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
    'use strict';
    controllers.controller('exampleCtrl', ['$scope', '$timeout', 'DataFactory',
        function ($scope, $timeout, DataFactory) {
            var vm = $scope;
            //$timeout(helper.zebraRows.bind(null, document, ".row", "#EEEEEE",
"#CCCCCC", "rowCont"));
            vm.valueArrays = {};
            vm.models = {};
            var pollOnChange = function () {
                $timeout(function () {
                    vm.template.models.multiplied.model =
helper.getValue("example.multiplied");
                });
            };
            vm.template = {
                name: 'example',
                readable: 'Example',
                models: {
                    multiplier: {
                        name: 'multiplier',
                        header: 'Multiplier',
                        type: 'text',
                        onChange: pollOnChange,
                        bind: {
                            name: "example",
                            value: "multiplier"
                        },
                        source: 'An example model'
                    },
                    multiplied: {
                        name: 'multiplied',
                        header: 'Multiplied',
                        type: 'text',
                        disabled: "true",
                        bind: {
                            name: "example",
                            value: "multiplied"
                        },
                        source: 'An example model'
                    }
                },
            };
            helper.mapDefaultValues(vm, vm.template, true);
        }]);
});
```

Appendix 29

Ikkunat controller (ikkunat.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('ikkunatCtrl', ['$scope', '$timeout', 'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;
      $timeout(helper.zebraRows.bind(null, document, ".row", "#EEEEEE",
"#CCCCCC", "rowCont"));
      vm.valueArrays = {
        ilmansuunnat: [
          [
            "pohjoinen", "koillinen"
          ],
          [
            "ita", "kaakko"
          ],
          [
            "etela", "lounas"
          ],
          [
            "lansi", "luode"
          ]
        ]
      };
      vm.models = {};
      vm.template = {
        name: 'ikkunat',
        readable: "Ikkunat",
        models: {
          ilmansuunnat: {
            name: "ilmansuunnat",
            header: "ilmansuunnat",
            type: "ilmansuunnat",
            source: "Havainnointi paikanpäällä",
            koillinen: DataFactory.ikkunat.ilmansuunnat.koillinen,
            kaakko: DataFactory.ikkunat.ilmansuunnat.kaakko,
            lounas: DataFactory.ikkunat.ilmansuunnat.lounas,
            luode: DataFactory.ikkunat.ilmansuunnat.luode,
          },
          pintaAla: {
            name: 'pintaAla',
            header: 'Pinta-ala (puite- ja karmirakenteineen)',
            type: 'text',
            koillinen: DataFactory.ikkunat.pintaAla.koillinen,
            kaakko: DataFactory.ikkunat.pintaAla.kaakko,
            lounas: DataFactory.ikkunat.pintaAla.lounas,
            luode: DataFactory.ikkunat.pintaAla.luode,
            prefix: "A",
            prefixSub: "ikk",
            suffix: "m²",
            source: 'Havainnointi paikanpäällä'
          },
          uArvo: {
            name: 'uArvo',
            header: 'Keskimääräinen U-arvo ilmansuunnittain',
            type: 'text',
            koillinen: DataFactory.ikkunat.uArvo.koillinen,
            kaakko: DataFactory.ikkunat.uArvo.kaakko,
            lounas: DataFactory.ikkunat.uArvo.lounas,
            luode: DataFactory.ikkunat.uArvo.luode,
            prefix: "U",
```

```
            suffix: "W/(m² K)",
            source: 'Havainnointi paikanpäällä'
          },
          auringonSateily: {
            name: 'auringonSateily',
            header: 'Auringon säteilyn kokonaisläpäisykerroin',
            type: 'text',
            koillinen: DataFactory.ikkunat.auringonSateily.koillinen,
            kaakko: DataFactory.ikkunat.auringonSateily.kaakko,
            lounas: DataFactory.ikkunat.auringonSateily.lounas,
            luode: DataFactory.ikkunat.auringonSateily.luode,
            prefix: "g",
            source: 'valmistajan ilmoittama arvo'
          },
          kehakerroin: {
            name: 'kehakerroin',
            header: 'Kehäkerroin',
            type: 'text',
            koillinen: DataFactory.ikkunat.kehakerroin.koillinen,
            kaakko: DataFactory.ikkunat.kehakerroin.kaakko,
            lounas: DataFactory.ikkunat.kehakerroin.lounas,
            luode: DataFactory.ikkunat.kehakerroin.luode,
            prefix: "F",
            prefixSub: "kehä",
            source: 'D5/2012 kohta 5.3.4 oletusarvo'
          },
          verhokerroin: {
            name: 'verhokerroin',
            header: 'Verhokerroin',
            type: 'text',
            koillinen: DataFactory.ikkunat.verhokerroin.koillinen,
            kaakko: DataFactory.ikkunat.verhokerroin.kaakko,
            lounas: DataFactory.ikkunat.verhokerroin.lounas,
            luode: DataFactory.ikkunat.verhokerroin.luode,
            prefix: "F",
            prefixSub: "verho",
            source: 'havainnointi paikanpäällä: ei valon esteitä päiväsaikaan'
          },
          separator: {
            header: "Ulkopuolisen varjostusten kulmat Apukuvan 1. merkinnöin. Jos
ei varjostusta, kulma on 0°",
            type: 'separator'
          },
          ymparistokerroin: {
            name: 'ymparistokerroin',
            header: 'Ympäristökertoimen kulma',
            type: 'text',
            koillinen: DataFactory.ikkunat.ymparistokerroin.koillinen,
            kaakko: DataFactory.ikkunat.ymparistokerroin.kaakko,
            lounas: DataFactory.ikkunat.ymparistokerroin.lounas,
            luode: DataFactory.ikkunat.ymparistokerroin.luode,
            prefix: "φ",
            suffix: "°",
            source: 'havainnointi paikanpäällä: ei yläpuolista varjostusta'
          },
          ylapuolinenVarjostus: {
            name: 'ylapuolinenVarjostus',
            header: 'Yläpuolisen varjostuksen kulma',
            type: 'text',
            koillinen: DataFactory.ikkunat.ylapuolinenVarjostus.koillinen,
            kaakko: DataFactory.ikkunat.ylapuolinenVarjostus.kaakko,
            lounas: DataFactory.ikkunat.ylapuolinenVarjostus.lounas,
            luode: DataFactory.ikkunat.ylapuolinenVarjostus.luode,
            prefix: "α",
```

```
            suffix: "°",
            source: 'havainnointi paikanpäällä: ei sivuvarjostusta'
          },
          sivuvarjostus: {
            name: 'sivuvarjostus',
            header: 'Sivuvarjostuksen kulma',
            type: 'text',
            koillinen: DataFactory.ikkunat.sivuvarjostus.koillinen,
            kaakko: DataFactory.ikkunat.sivuvarjostus.kaakko,
            lounas: DataFactory.ikkunat.sivuvarjostus.lounas,
            luode: DataFactory.ikkunat.sivuvarjostus.luode,
            prefix: "β",
            suffix: "°",
            source: 'havainnointi paikanpäällä D5/20112 taul. 5.3 s. 32'
          },
        },
      };
      var keys = Object.keys(vm.template.models);
      keys.forEach(function (key, index) {
        var directions = ["koillinen", "kaakko", "lounas", "luode"];
        directions.forEach(function (dir, index) {
          if (vm.template.models[key].type !== "separator") {
            vm.$watch("template.models." + key + "."+dir, function (value) {
              if (typeof value === "string" &&
/^\d+?(?:[\.\,]\d{1,})?$/.test(value)) {
                DataFactory[vm.template.name][key][dir] =
Number(value.replace(/\,/g, "."));
                vm.template.models[key].error = false;
              }
              else if (vm.template.models[key].noValidate === true || typeof
value === "number" ||
                !/text/.test(vm.template.models[key].type)) {
                DataFactory[vm.template.name][key][dir] = value;
                vm.template.models[key].error = false;
              }
              else {
                vm.template.models[key].error = true;
                DataFactory[vm.template.name][key][dir] = value;
              }
            }, true);
          }
        });
      });
      helper.mapDefaultValues(vm, vm.template);
    }]);
});
```

Appendix 30

ilmaIlmaLampopumput Controller (ilmaIlmaLampopumput.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('ilmaIlmaLampopumputCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;
      $timeout(helper.zebraRows.bind(null, document, ".row", "#EEEEEE", "#CCCCCC",
"rowCont"));
      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'ilmaIlmaLampopumput',
        readable: "Ilma-ilma lämpöpumput tilojen lämmityksessä",
        models: {
          pumppu1: {
            name: 'pumppu1',
            header: 'Pumppu 1',
            type: 'text',
            bind: {
              name: "ilmaIlmaLampopumput",
              value: "pumppu1"
            },
            bind2: {
              name: "ilmaIlmaLampopumput",
              value: "pumppu12"
            },
            source: 'Huomioi E-laskennassa katso Ym 176/2013 s.16 enimmäisarvot.'
          },
          pumppu2: {
            name: 'pumppu2',
            header: 'Pumppu 2',
            type: 'text',
            bind: {
              name: "ilmaIlmaLampopumput",
              value: "pumppu2"
            },
            bind2: {
              name: "ilmaIlmaLampopumput",
              value: "pumppu22"
            },
            source: 'Huomioi E-laskennassa katso Ym 176/2013 s.16 enimmäisarvot.'
          },
          pumppu3: {
            name: 'pumppu3',
            header: 'Pumppu 3',
            type: 'text',
            bind: {
              name: "ilmaIlmaLampopumput",
              value: "pumppu3"
            },
            bind2: {
              name: "ilmaIlmaLampopumput",
              value: "pumppu32"
            },
            source: 'Huomioi E-laskennassa katso Ym 176/2013 s.16 enimmäisarvot.'
          }
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 31

Ilmanvaihtojarjestelma Controller (ilmanvaihtojarjestelma.js)

```
define(["./module", "helper", "json"], function (controllers, helper, data) {
  "use strict";

  var ilmanvaihtoKoneMaxMaara = 3;

  controllers.controller("ilmanvaihtojarjestelmaCtrl", ["$scope", "$timeout",
"DataFactory",
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;


      var lammonTalteenotonLammonvaihdinChange = function () {
        if (vm.template.models.lammontalteenotonLammonvaihdin.model.value) {
          vm.template.models.lammontalteenotonLammonvaihdin.model2 =
            vm.template.models.lammontalteenotonLammonvaihdin.model.value;
        }
      };

      vm.valueArrays = {
        ilmanvaihtojarjestelma: ["Koneellinen tulopoisto", "Painovoimainen",
"Koneellinen poisto"],
        ilmanvaihtokoneidenMaara: helper.arrayInRange(ilmanvaihtoKoneMaxMaara +
1),
        ilmanvaihtoLammontalteenotto: ["Kyllä", "Ei"],
        LTOpoiskytkenta: ["Kyllä", "Ei"],
        tuloilmanJalkilammitys: ["Kyllä", "Ei"],
        tuloilmanJalkilammitysLampolahde: ["Sähkövastus", ""],
        lammontalteenotonLammonvaihdin: data.lammontalteenotonLammonvaihdin
      };
      vm.models = {};
      vm.template = {
        name: "ilmanvaihtojarjestelma",
        readable: "Ilmanvaihtojärjestelmä",
        models: {
          ilmanvaihtojarjestelma: {
            name: 'ilmanvaihtojarjestelma',
            header: 'Ilmanvaihtojärjestelmä',
            type: 'select',
            bind: {
              name: "ilmanvaihtojarjestelma",
              value: "ilmanvaihtojarjestelma"
            },
            options: 'ilmanvaihto as ilmanvaihto for ilmanvaihto in
valueArrays.ilmanvaihtojarjestelma',
            source: 'havainnointi paikanpäällä'
          },
          ilmanvaihtokoneidenMaara: {
            name: 'ilmanvaihtokoneidenMaara',
            header: 'Ilmanvaihtokoneiden lukumäärä',
            type: 'select',
            bind: {
              name: "ilmanvaihtojarjestelma",
              value: "ilmanvaihtokoneidenMaara"
            },
            options: 'amount as amount for amount in
valueArrays.ilmanvaihtokoneidenMaara',
            source: 'havainnointi paikanpäällä'
          },
          ilmanvaihtoLammontalteenotto: {
            name: 'ilmanvaihtoLammontalteenotto',
```

```
            header: 'ilmanvaihdon lämmöntalteenotto',
            type: 'switch',
            bind: {
              name: "ilmanvaihtojarjestelma",
              value: "ilmanvaihtoLammontalteenotto"
            },
            options: 'bool as bool for bool in
valueArrays.ilmanvaihtoLammontalteenotto',
            source: 'havainnointi paikanpäällä'
          },
          LTOpoiskytkenta: {
            name: 'LTOpoiskytkenta',
            header: 'LTO:n poiskytkentä asetusarvon ylittyessä',
            type: 'switch',
            bind: {
              name: "ilmanvaihtojarjestelma",
              value: "LTOpoiskytkenta"
            },
            options: 'bool as bool for bool in valueArrays.LTOpoiskytkenta',
            source: 'havainnointi paikanpäällä'
          },
          tuloilmanJalkilammitys: {
            name: 'tuloilmanJalkilammitys',
            header: 'Tuloilman jälkilämmitys',
            type: 'switch',
            bind: {
              name: "ilmanvaihtojarjestelma",
              value: "tuloilmanJalkilammitys"
            },
            options: 'bool as bool for bool in
valueArrays.tuloilmanJalkilammitys',
            source: 'havainnointi paikanpäällä'
          },
          tuloilmanJalkilammitysLampolahde: {
            name: 'tuloilmanJalkilammitysLampolahde',
            header: 'Tuloilman jälkilämmityksen lämmönlähde',
            type: 'select',
            bind: {
              name: "ilmanvaihtojarjestelma",
              value: "tuloilmanJalkilammitysLampolahde"
            },
            options: 'source as source for source in
valueArrays.tuloilmanJalkilammitysLampolahde',
            source: 'havainnointi paikanpäällä'
          },
          LTOSuhde: {
            name: 'LTOSuhde',
            header: 'LTO lämpötilasuhde (tulo- ja poistoilmavirrat yhtä suuria)',
            type: 'text',
            suffix: "%",
            bind: {
              name: "ilmanvaihtojarjestelma",
              value: "LTOSuhde"
            },
            source: 'havainnointi paikanpäällä, valmistajan ilmoittama arvo'
          },
          lammontalteenotonLammonvaihdin: {
            name: 'lammontalteenotonLammonvaihdin',
            name2: 'lammontalteenotonLammonvaihdinText',
            header: 'Lämmöntalteenoton lämmönvaihdintyyppi',
            type: 'select+text',
            disabled2:
"template.models.lammontalteenotonLammonvaihdin.model.value",
            onChange: lammonTalteenotonLammonvaihdinChange,
```

```
            bind: {
              name: "ilmanvaihtojarjestelma",
              value: "lammontalteenotonLammonvaihdin"
            },
            bind2: {
              name: "ilmanvaihtojarjestelma",
              value: "lammontalteenotonLammonvaihdin2"
            },
            options: 'tyyppi as tyyppi.name for tyyppi in
valueArrays.lammontalteenotonLammonvaihdin ' +
            'track by tyyppi.name',
            source: 'havainnointi paikanpäällä'
          },
          JateilmanAlinLampotila: {
            name: 'JateilmanAlinLampotila',
            header: 'Jäteilman alin mahdollinen  lämpötila, °C',
            type: 'text',
            suffix: '°C',
            bind: {
              name: "ilmanvaihtojarjestelma",
              value: "JateilmanAlinLampotila"
            },
            source: 'havainnointi paikanpäällä, valmistajan ilmoittama arvo
(D5/2012 kohta 3.4.1) s.22'
          },
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);

    }]);
});
```

Appendix 32

Ilmanvaihtojarjestelma 2 Controller (ilmanvaihtojarjestelma2.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('ilmanvaihtojarjestelma2Ctrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;

      var changeCheck = function () {
        $timeout(function () {
          vm.template.models.ilmanvaihtoTuloilmavirta.model =
            helper.getValue("ilmanvaihtojarjestelma2.ilmanvaihtoTuloilmavirta");
          vm.template.models.ilmanvaihtoKorvausilmavirta.model =

helper.getValue("ilmanvaihtojarjestelma2.ilmanvaihtoKorvausilmavirta");
        });
      };
      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'ilmanvaihtojarjestelma2',
        readable: "Ilmanvaihtojärjestelmä",
        models: {
          ilmanvaihtoTarkistus: {
            name: 'ilmanvaihtoTarkistus',
            header: 'Laskenta ilmanvaihtokertoimen mukaan (E-lukulaskennassa aina
ei)',
            type: 'switch',
            onChange: changeCheck,
            bind: {
              name: "ilmanvaihtojarjestelma2",
              value: "ilmanvaihtoTarkistus"
            },
            source: ''
          },
          ilmanvaihtokerroin: {
            name: 'ilmanvaihtokerroin',
            header: 'Ilmanvaihtokerroin',
            type: 'text',
            suffix: "1/h",
            onChange: changeCheck,
            bind: {
              name: "ilmanvaihtojarjestelma2",
              value: "ilmanvaihtokerroin"
            },
            source: ''
          },
          ilmanvaihtoLTOpoistohyoty: {
            name: 'ilmanvaihtoLTOpoistohyoty',
            header: 'Ilmanvaihtokoneen LTO:n poistoilman vuosihyötysuhde',
            type: 'text',
            bind: {
              name: "ilmanvaihtojarjestelma2",
              value: "ilmanvaihtoLTOpoistohyoty"
            },
            prefix: "η",
            prefixSub: "a,ivkone",
            suffix: "%",
            source: 'Lämpötilasuhteesta YM Monisteen 122 mukaisesti laskettuna'
          },
          // note the watchers for this are added below,
```

```
            ilmanvaihtoPoistoilmavirta: {
              name: 'ilmanvaihtoPoistoilmavirta',
              header: 'Ilmanvaihdon poistoilmavirta',
              type: 'text',
              bind: {
                name: "ilmanvaihtojarjestelma2",
                value: "ilmanvaihtoPoistoilmavirta"
              },
              onChange: changeCheck,
              disabled: "true",
              prefix: "q",
              prefixSub: "v,poisto",
              suffix: "m³/s",
              source: 'D3/2012 taulukko 2 s.18. E-lukulaskennassa tulo- ja
poistoilmavirrat yhtä suuria'
            },
            // note the watchers for this are added below,
            ilmanvaihtoTuloilmavirta: {
              name: 'ilmanvaihtoTuloilmavirta',
              header: 'Ilmanvaihdon tuloilmavirta',
              type: 'text',
              bind: {
                name: "ilmanvaihtojarjestelma2",
                value: "ilmanvaihtoTuloilmavirta"
              },
              onChange: changeCheck,
              disabled: "true",
              prefix: "q",
              prefixSub: "v,tulo",
              suffix: "m³/s",
              source: 'D3/2012 taulukko 2 s.18'
            },
            ilmanvaihtoSPFLuku: {
              name: 'ilmanvaihtoSPFLuku',
              header: 'Ilmanvaihdon SFP-luku eli ominaissähköteho',
              type: 'text',
              bind: {
                name: "ilmanvaihtojarjestelma2",
                value: "ilmanvaihtoSPFLuku"
              },
              suffix: "kW/(m³/s)",
              source: 'Valmistajan ilmoittama arvo YM as. 176/2013, L 1, taul. 3'
            },
            ilmanvaihtoKorvausilmavirta: {
              name: 'ilmanvaihtoKorvausilmavirta',
              header: 'Ilmanvaihdon korvausilmavirta',
              type: 'text',
              bind: {
                name: "ilmanvaihtojarjestelma2",
                value: "ilmanvaihtoKorvausilmavirta"
              },
              onChange: changeCheck,
              disabled: "true",
              prefix: "'q",
              prefixSub: "v,korvausilma",
              suffix: "m³/s",
              source: ''
            },
            tuloilmaSisaanpuhalluslampotila: {
              name: 'tuloilmaSisaanpuhalluslampotila',
              header: 'Tuloilman sisäänpuhalluslämpötila',
              type: 'text',
              bind: {
                name: "ilmanvaihtojarjestelma2",
```

```
            value: "tuloilmaSisaanpuhalluslampotila"
          },
          prefix: "T",
          prefixSub: "sp",
          suffix: "°C",
          source: 'Havainnointi paikanpäällä'
        },
        tuloilmaLammonNousu: {
          name: 'tuloilmaLammonNousu',
          header: 'Lämpötilan nousu tuloilmapuhaltimessa (E-laskennassa 0.5)',
          type: 'text',
          bind: {
            name: "ilmanvaihtojarjestelma2",
            value: "tuloilmaLammonNousu"
          },
          prefix: "ΔT",
          prefixSub: "puhallin",
          suffix: "°C",
          source: 'E-laskennassa 0,5 D5/2012 luku 3.4 (oletusarvo)'
        },
        ilmanvaihtolaitoksenVuorokausiSuhde: {
          name: 'ilmanvaihtolaitoksenVuorokausiSuhde',
          header: 'Ilmanvaihtolaitoksen vuorokautinen käyntiaikasuhde h/(24 h)
(E-laskennassa 1.0)',
          type: 'text',
          bind: {
            name: "ilmanvaihtojarjestelma2",
            value: "ilmanvaihtolaitoksenVuorokausiSuhde"
          },
          prefix: "t",
          prefixSub: "d",
          source: 'E-laskennassa 1,0 D3/2012 luku 3.3 taulukko 3'
        },
        ilmanvaihtolaitoksenViikkoSuhde: {
          name: 'ilmanvaihtolaitoksenViikkoSuhde',
          header: 'Ilmanvaihtolaitoksen viikoittainen käyntiaikasuhde vrk/(7
vrk) (E-laskennassa 1.0)',
          type: 'text',
          bind: {
            name: "ilmanvaihtojarjestelma2",
            value: "ilmanvaihtolaitoksenViikkoSuhde"
          },
          prefix: "t",
          prefixSub: "v",
          source: 'E-laskennassa 1,0 D3/2012 luku 3.3 taulukko 3'
        },
        ilmanvaihtoLammityspatteriSuhde: {
          name: 'ilmanvaihtoLammityspatteriSuhde',
          header: 'Ilmanvaihdon lämmityspatterin lämmöntuoton hyötysuhde (E-
laskennassa oletusarvo 1.0)',
          type: 'text',
          bind: {
            name: "ilmanvaihtojarjestelma2",
            value: "ilmanvaihtoLammityspatteriSuhde"
          },
          prefix: "η",
          prefixSub: "tuotto,iv",
          source: 'Ym energiatodistusoppaan 2013 Liite 5.11.2013 Energiatod.
laadintaesim. OK-talo v. 2000 s.40'
        },
      },
    };
    vm.$watch(
      function () {
```

```
          return {
            a: vm.template.models.ilmanvaihtoTuloilmavirta.model,
            b: vm.template.models.ilmanvaihtoKorvausilmavirta.model
          };
        },
        function () {
          vm.template.models.ilmanvaihtoPoistoilmavirta.model =

helper.getValue("ilmanvaihtojarjestelma2.ilmanvaihtoPoistoilmavirta");
        },
        true
      );


      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 33

Jaahdytysjarjestelma Controller (jaahdytysjarjestelma.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('jaahdytysjarjestelmaCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;

      var changeFn = function () {
        $timeout(function () {
          vm.template.models.lampotyyppi.model =
helper.getValue("jaahdytysjarjestelma.lampotyyppi");
        });
      };

      vm.valueArrays = {
        lampopumpunLaji: data.lampopumpunLaji
      };
      vm.models = {};
      vm.template = {
        name: 'jaahdytysjarjestelma',
        readable: "Jäähdytysjärjestelmä",
        models: {
          info: {
            header: "Täytä vain, jos jäähdytys on yksittäisissä pienissä tiloissa
tai jos sekä lämmitys-" +
            " että jäähdytysenergian nettotarpeet on laskettu dynaamisella
menetelmällä",
            type: "separator"
          },
          energiamuoto: {
            name: 'energiamuoto',
            header: 'Energiamuoto',
            type: 'select',
            bind: {
              name: "jaahdytysjarjestelma",
              value: "energiamuoto"
            },
            onChange: changeFn,
            options: 'pumppu as pumppu.name for pumppu in
valueArrays.lampopumpunLaji track by pumppu.name',
            source: ''
          },
          lampotyyppi: {
            name: 'lampotyyppi',
            header: 'Energiamuodon tyyppi',
            type: 'value',
            bind: {
              name: "jaahdytysjarjestelma",
              value: "lampotyyppi"
            },
            source: ''
          },
          jaahdytyksenNettotarve: {
            name: 'jaahdytyksenNettotarve',
            header: 'Jäähdytyksen nettotarve',
            type: 'text',
            suffix: "kWh/a",
            bind: {
              name: "jaahdytysjarjestelma",
              value: "jaahdytyksenNettotarve"
```

```
            },
            source: ''
          },
          kokonaishyotysuhde: {
            name: 'kokonaishyotysuhde',
            header: 'Kokonaishyötysuhde',
            type: 'text',
            bind: {
              name: "jaahdytysjarjestelma",
              value: "kokonaishyotysuhde"
            },
            source: ''
          }
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 34

Kayttovesijarjestelma Controller (kayttovesijarjestelma.js)

```
define(["./module", "helper", "json"], function (controllers, helper, data) {
  "use strict";

  controllers.controller("kayttovesijarjestelmaCtrl", ["$scope", "$timeout",
"DataFactory",
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;


      var vedenJakeluWatcher = function (newVal) {
        for (var model of vm.valueArrays.vedenJakelujarjestelma.slice(1)) {
          var val = helper.getValue(model.Value);
          model.Name = model.Name.replace(/\d+[\.\,]?\d*$/, val);
        }
        vm.template.models.vedenJakelujarjestelma.model2 =
          helper.getValue(vm.template.models.vedenJakelujarjestelma.model.Value);
      };

      var vedenVaraajaChange = function () {
        if
(vm.template.models.vedenVaraaja.model.VaraajanTilavuusEristePaksuusVuotuinenLamp
ohavio
          != "Oma valinta") {
          vm.template.models.vedenVaraaja.model2 =
            vm.template.models.vedenVaraaja.model.varaajanLampohavio;
        }
      };

      var vedenJakelujarjestelmaChange = function () {
        if (vm.template.models.vedenJakelujarjestelma.model.Value) {
          vedenJakeluWatcher();
        }
      };

      vm.valueArrays = {
        vedenLammitysjarjestelma: data.warmWaterHeatingSystem,
        vedenVaraaja: data.waterHeater,
        vedenKierto: ["On", "Ei ole"],
        kierronLammityslaitteet: data.circulationHeatingSystems,
        kiertoputkienEristys: data.kiertoPutkienEristys,
        vedenJakelujarjestelma: data.vedenJakelujarjestelma
      };
      vm.models = {};
      vm.template = {
        name: "kayttovesijarjestelma",
        readable: "Käyttövesijärjestelmä",
        models: {
          lampimanKayttovedenLammitysjarjestelma: {
            name: "lampimanKayttovedenLammitysjarjestelma",
            header: "Lämpimän käyttöveden lämmitysjärjestelmä",
            type: "select",
            bind: {
              name: "kayttovesijarjestelma",
              value: "lampimanKayttovedenLammitysjarjestelma"
            },
            options: "city as city.name for city in
valueArrays.vedenLammitysjarjestelma track by city.name",
            source: "havainnointi paikanpäällä"
          },
          vedenVaraaja: {
```

```
            name: "vedenVaraaja",
            name2: "vedenVaraajaText",
            header: "Lämpimän käyttöveden varaaja",
            type: "select+text",
            onChange: vedenVaraajaChange,
            suffix2: "kWh/a",
            disabled2: "template.models.vedenVaraaja.model.varaajanLampohavio",
            bind: {
              name: "kayttovesijarjestelma",
              value: "vedenVaraaja"
            },
            bind2: {
              name: "kayttovesijarjestelma",
              value: "vedenVaraaja2"
            },
            options: "city as
city.VaraajanTilavuusEristePaksuusVuotuinenLampohavio " +
            "for city in valueArrays.vedenVaraaja track by
city.VaraajanTilavuusEristePaksuusVuotuinenLampohavio",
            source: "havainnointi paikanpäällä"
          },
          vedenKierto: {
            name: "vedenKierto",
            header: "Lämpimän käyttöveden kierto",
            type: "switch",
            bind: {
              name: "kayttovesijarjestelma",
              value: "vedenKierto"
            },
            options: "kierto as kierto for kierto in valueArrays.vedenKierto",
            source: "Havainnointi paikanpäällä"
          },
          kierronLammityslaitteet: {
            name: 'kierronLammityslaitteet',
            header: 'kierron lämmityslaitteet',
            type: 'select',
            bind: {
              name: "kayttovesijarjestelma",
              value: "kierronLammityslaitteet"
            },
            options: 'kierto as kierto.KierronLammitysLaitteet for ' +
            'kierto in valueArrays.kierronLammityslaitteet',
            source: 'YM asetus 176/2013 liite 1  Taul. 5 s.10-11'
          },
          kiertoputkienEristys: {
            name: 'kiertoputkienEristys',
            header: 'Kiertoputkien eristys',
            type: 'select',
            bind: {
              name: "kayttovesijarjestelma",
              value: "kiertoputkienEristys"
            },
            options: 'kierto as kierto.KiertojohdonEristys for kierto in
valueArrays.kiertoputkienEristys ' +
            'track by kierto.kiertojohdonEristys',
            source: 'havainnointi paikanpäällä'
          },
          vedenJakelujarjestelma: {
            name: "vedenJakelujarjestelma",
            name2: "vedenJakelujarjestelmaText",
            header: "Lämpimän käyttöveden siirto: rakenne ja hyötysuhde",
            type: "select+text",
            onChange: vedenJakelujarjestelmaChange,
            disabled2: "template.models.vedenJakelujarjestelma.model.Value",
```

```
            bind: {
              name: "kayttovesijarjestelma",
              value: "vedenJakelujarjestelma"
            },
            bind2: {
              name: "kayttovesijarjestelma",
              value: "vedenJakelujarjestelma2"
            },
            options: "city as city.Name " +
            "for city in valueArrays.vedenJakelujarjestelma track by city.Name",
            source: "havainnointi paikanpäällä"
          }
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);
      vedenJakelujarjestelmaChange();
    }]);
});
```

Appendix 35

Kayttovesijarjestelma 2 Controller (kayttovesijarjestelma2.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('kayttovesijarjestelma2Ctrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;

      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'kayttovesijarjestelma2',
        readable: "Käyttövesijärjestelmä",
        models: {
          lampimankayttovedenLammitysenergianTarve: {
            name: 'lampimankayttovedenLammitysenergianTarve',
            header: 'Lämpimän käyttöveden lämmitysenergian nettotarve',
            type: 'text',
            disabled: "true",
            suffix: "kWh/(m² a)",
            bind: {
              name: "kayttovesijarjestelma2",
              value: "lampimankayttovedenLammitysenergianTarve"
            },
            source: 'D3/2012 taulukko 5 s.21'
          },
          lampimankayttovedenLammitysenergianTarveYla: {
            name: 'lampimankayttovedenLammitysenergianTarveYla',
            header: 'Lämpimän käyttöveden lämmitysenergian  nettotarpeen
yläraja',
            type: 'text',
            disabled: "true",
            suffix: "kWh/a",
            bind: {
              name: "kayttovesijarjestelma2",
              value: "lampimankayttovedenLammitysenergianTarveYla"
            },
            source: 'D3/2012 taulukko 5 s.21'
          },
          lampimankayttovedenLammitysenergianTarveELuku: {
            name: 'lampimankayttovedenLammitysenergianTarveELuku',
            header: 'Lämpimän käyttöveden lämmitysenergian nettotarve E-
lukulaskennassa',
            type: 'text',
            disabled: "true",
            suffix: "kWh/(m² a)",
            bind: {
              name: "kayttovesijarjestelma2",
              value: "lampimankayttovedenLammitysenergianTarveELuku"
            },
            source: 'D3/2012 taulukko 5 s.21 + YM asetus 5/13 s.5 E-lukulaskennan
standardikäyttö'
          },
          lampimankayttovedenOminaiskulutus: {
            name: 'lampimankayttovedenOminaiskulutus',
            header: 'Lämpimän käyttöveden ominaiskulutus',
            type: 'text',
            disabled: "true",
            suffix: "dm³/(m² a)",
            bind: {
              name: "kayttovesijarjestelma2",
```

```
              value: "lampimankayttovedenOminaiskulutus"
            },
            source: 'D3/2012 taulukko 5 s.21 + YM asetus 5/13 s.5 E-lukulaskennan
standardikäyttö'
          },
          lampimankayttovedenVarastoinninHavio: {
            name: 'lampimankayttovedenVarastoinninHavio',
            header: 'Lämpimän käyttöveden varastoinnin vuotuinen lämpöhäviö',
            type: 'text',
            disabled: "true",
            prefix: "Q",
            prefixSub: "lkv,varastointi",
            suffix: "kWh/a",
            bind: {
              name: "kayttovesijarjestelma2",
              value: "lampimankayttovedenVarastoinninHavio"
            },
            source: 'YM asetus 176/2013, liite 1, taulukko 8 s.11'
          },
          lampimankayttovedenJakeluHyoty: {
            name: 'lampimankayttovedenJakeluHyoty',
            header: 'Lämpimän käyttöveden jakelun (siirron) hyötysuhde',
            type: 'text',
            disabled: "true",
            prefix: "η",
            prefixSub: "lkv,siirto",
            bind: {
              name: "kayttovesijarjestelma2",
              value: "lampimankayttovedenJakeluHyoty"
            },
            source: 'YM asetus 176/2013, liite 1, taulukko 5 s.10 erillinen
pientalo: ei kiertoa, suojaputkessa'
          },
          lampimankayttovedenKierronHavio: {
            name: 'lampimankayttovedenKierronHavio',
            header: 'Lämpimän käyttöveden kierron lämpöhäviö',
            type: 'text',
            disabled: "true",
            prefix: "Q",
            prefixSub: "lkv,kierto",
            suffix: "kWh/a",
            bind: {
              name: "kayttovesijarjestelma2",
              value: "lampimankayttovedenKierronHavio"
            },
            source: 'ei kiertojohtoa, D5 6.3.3 s.42'
          }
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 36

Kuluttajalaitteet Controller (kuluttajalaitteet.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('kuluttajalaitteetCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;
      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'kuluttajalaitteet',
        readable: "Kuluttajalaitteet",
        models: {
          rakennuksenPaivittainenKayttoaikasuhde: {
            name: 'rakennuksenPaivittainenKayttoaikasuhde',
            header: 'Rakennuksen päivittäinen käyttöaikasuhde h/(24 h)',
            type: 'text',
            disabled: "true",
            bind: {
              name: "kuluttajalaitteet",
              value: "rakennuksenPaivittainenKayttoaikasuhde"
            },
            source: 'D3/2012 luku 3.3 taulukko 3'
          },
          rakennuksenViikottainenKayttoaikasuhde: {
            name: 'rakennuksenViikottainenKayttoaikasuhde',
            header: 'Rakennuksen viikottainen käyttöaikasuhde vrk/(7 vrk)',
            type: 'text',
            disabled: "true",
            bind: {
              name: "kuluttajalaitteet",
              value: "rakennuksenViikottainenKayttoaikasuhde"
            },
            source: 'D3/2012 luku 3.3 taulukko 3'
          },
          kuluttajalaitteidenOminaisteho: {
            name: 'kuluttajalaitteidenOminaisteho',
            header: 'Kuluttajalaitteiden ominaisteho',
            type: 'text',
            disabled: "true",
            prefix: "W/m²",
            bind: {
              name: "kuluttajalaitteet",
              value: "kuluttajalaitteidenOminaisteho"
            },
            source: 'D3/2012 luku 3.3 taulukko 3'
          },
          kuluttajalaitteidenKayttoaste: {
            name: 'kuluttajalaitteidenKayttoaste',
            header: 'Kuluttajalaitteiden Käyttöaste',
            type: 'text',
            disabled: "true",
            bind: {
              name: "kuluttajalaitteet",
              value: "kuluttajalaitteidenKayttoaste"
            },
            source: 'D3/2012 luku 3.3 taulukko 3'
          },
          valaistuksenOminaisteho: {
            name: 'valaistuksenOminaisteho',
            header: 'Valaistuksen ominaisteho',
```

```
            type: 'text',
            disabled: "true",
            prefix: "W/m²",
            bind: {
              name: "kuluttajalaitteet",
              value: "valaistuksenOminaisteho"
            },
            source: 'D3/2012 luku 3.3 taulukko 3'
          },
          valaistuksenKayttoaste: {
            name: 'valaistuksenKayttoaste',
            header: 'Valaistuksen Käyttöaste',
            type: 'text',
            disabled: "true",
            bind: {
              name: "kuluttajalaitteet",
              value: "valaistuksenKayttoaste"
            },
            source: 'D3/2012 luku 3.3 taulukko 3'
          },
          lampokuormaIhmisista: {
            name: 'lampokuormaIhmisista',
            header: 'Lämpökuorma ihmisistä',
            type: 'text',
            disabled: "true",
            prefix: "W/m²",
            bind: {
              name: "kuluttajalaitteet",
              value: "lampokuormaIhmisista"
            },
            source: 'D3/2012 luku 3.3 taulukko 3'
          },
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 37

Kuukausi-talteenotto Controller (kuukausi-talteenotto.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('kuukausiTalteenottoCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;
      vm.valueArrays = {};
      $timeout(helper.zebraRows.bind(null, document, ".row", "#EEEEEE",
"#CCCCCC", "rowCont"));
      var monthChange = function () {
        this.model2 = this.model;
      };
      vm.models = {};
      vm.template = {
        name: 'kuukausiTalteenotto',
        readable: "Ilmanvaihdon lämmöntalteenoton ja jälkilämmityksen
kuukausiaikataulu",
        models: {
          tammikuu: {
            name: 'tammikuu',
            header: 'Tammikuu',
            type: 'slider',
            bind: {
              name: "kuukausiTalteenotto",
              value: "tammikuu"
            },
            model2: DataFactory.kuukausiTalteenotto.tammikuu2
          },
          helmikuu: {
            name: 'helmikuu',
            header: 'Helmikuu',
            type: 'slider',
            bind: {
              name: "kuukausiTalteenotto",
              value: "helmikuu"
            },
            model2: DataFactory.kuukausiTalteenotto.helmikuu2
          },
          maaliskuu: {
            name: 'maaliskuu',
            header: 'Maaliskuu',
            type: 'slider',
            bind: {
              name: "kuukausiTalteenotto",
              value: "maaliskuu"
            },
            model2: DataFactory.kuukausiTalteenotto.maaliskuu2
          },
          huhtikuu: {
            name: 'huhtikuu',
            header: 'Huhtikuu',
            type: 'slider',
            bind: {
              name: "kuukausiTalteenotto",
              value: "huhtikuu"
            },
            model2: DataFactory.kuukausiTalteenotto.huhtikuu2
          },
          toukokuu: {
            name: 'toukokuu',
```

```
    header: 'Toukokuu',
    type: 'slider',
    bind: {
      name: "kuukausiTalteenotto",
      value: "toukokuu"
    },
    model2: DataFactory.kuukausiTalteenotto.toukokuu2
  },
  kesakuu: {
    name: 'kesakuu',
    header: 'Kesäkuu',
    type: 'slider',
    bind: {
      name: "kuukausiTalteenotto",
      value: "kesakuu"
    },
    model2: DataFactory.kuukausiTalteenotto.kesakuu2
  },
  heinakuu: {
    name: 'heinakuu',
    header: 'Heinäkuu',
    type: 'slider',
    bind: {
      name: "kuukausiTalteenotto",
      value: "heinakuu"
    },
    model2: DataFactory.kuukausiTalteenotto.heinakuu2
  },
  elokuu: {
    name: 'elokuu',
    header: 'Elokuu',
    type: 'slider',
    bind: {
      name: "kuukausiTalteenotto",
      value: "elokuu"
    },
    model2: DataFactory.kuukausiTalteenotto.elokuu2
  },
  syyskuu: {
    name: 'syyskuu',
    header: 'Syyskuu',
    type: 'slider',
    bind: {
      name: "kuukausiTalteenotto",
      value: "syyskuu"
    },
    model2: DataFactory.kuukausiTalteenotto.syyskuu2
  },
  lokakuu: {
    name: 'lokakuu',
    header: 'Lokakuu',
    type: 'slider',
    bind: {
      name: "kuukausiTalteenotto",
      value: "lokakuu"
    },
    model2: DataFactory.kuukausiTalteenotto.lokakuu2
  },
  marraskuu: {
    name: 'marraskuu',
    header: 'Marraskuu',
    type: 'slider',
    bind: {
      name: "kuukausiTalteenotto",
```

```
              value: "marraskuu"
            },
            model2: DataFactory.kuukausiTalteenotto.marraskuu2
          },
          joulukuu: {
            name: 'joulukuu',
            header: 'Joulukuu',
            type: 'slider',
            bind: {
              name: "kuukausiTalteenotto",
              value: "joulukuu"
            },
            model2: DataFactory.kuukausiTalteenotto.joulukuu2
          }
        },
      };
      Object.keys(vm.template.models).forEach(function (key) {
        vm.template.models[key].onChange =
monthChange.bind(vm.template.models[key]);
      });
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 38

Lammitysjarjestelma Controller (lammitysjarjestelma.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('lammitysjarjestelmaCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;

      var watchHandler = function (value) {
        if (this.model2 === false) {
          // need to wait for change in dataFactory first
          $timeout(function () { this.model =
helper.getValue("lammitysjarjestelma." + this.name); }.bind(this));
        }
      };

      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'lammitysjarjestelma',
        readable: 'Lämmitysjärjestelmä',
        models: {
          lammonjakojarjestelmanVuosihyoty: {
            name: 'lammonjakojarjestelmanVuosihyoty',
            header: 'Käytä omavalintaa',
            header2: 'Lämmönjakojärjestelmän vuosihyötysuhde',
            type: 'switch+text',
            onChange2: watchHandler,
            disabled: "template.models.lammonjakojarjestelmanVuosihyoty.model2
=== false",
            prefix: "η",
            noValidate2: true,
            prefixSub: "lämmitys,tilat",
            bind: {
              name: "lammitysjarjestelma",
              value: "lammonjakojarjestelmanVuosihyoty"
            },
            bind2: {
              name: "lammitysjarjestelma",
              value: "lammonjakojarjestelmanVuosihyoty2"
            },
            source: 'YM asetus 176/2013, liite 1, taulukko 9:
sähköpatterilämmitys s.12'
          },
          lammonjakojarjestelmanApulaitteidenKulutus: {
            name: 'lammonjakojarjestelmanApulaitteidenKulutus',
            header2: 'Lämmön jakelujärjestelmänapulaitteiden sähkönkulutus',
            header: "Käytä omavalintaa",
            type: 'switch+text',
            noValidate2: true,
            onChange2: watchHandler,
            disabled:
"template.models.lammonjakojarjestelmanApulaitteidenKulutus.model2 === false",
            prefix: "e",
            prefixSub: "tilat",
            suffix: "kWh/(m²a)",
            bind: {
              name: "lammitysjarjestelma",
              value: "lammonjakojarjestelmanApulaitteidenKulutus"
            },
            bind2: {
```

```
              name: "lammitysjarjestelma",
              value: "lammonjakojarjestelmanApulaitteidenKulutus2"
            },
            source: 'YM asetus 176/2013, liite 1, taulukko 9:
sähköpatterilämmitys s.12'
          },
          lammitysenergianTuottoTilatHyoty: {
            name: 'lammitysenergianTuottoTilatHyoty',
            header2: 'Lämmitysenergian tuoton hyötysuhde tilojen lämmityksessä',
            header: "Käytä omavalintaa",
            type: 'switch+text',
            noValidate2: true,
            onChange2: watchHandler,
            disabled: "template.models.lammitysenergianTuottoTilatHyoty.model2
=== false",
            prefix: "η",
            prefixSub: "tuotto, tilat",
            suffix: "",
            bind: {
              name: "lammitysjarjestelma",
              value: "lammitysenergianTuottoTilatHyoty"
            },
            bind2: {
              name: "lammitysjarjestelma",
              value: "lammitysenergianTuottoTilatHyoty2"
            },
            source: 'YM asetus 176/2013, liite 1, taulukko 10:
sähköpatterilämmitys s.12'
          },
          tilojenLammontuottoApulaitteidenKulutus: {
            name: 'tilojenLammontuottoApulaitteidenKulutus',
            header: "Käytä omavalintaa",
            header2: 'Tilojen lämmöntuottojärjestelmän apulaitteiden
sähkönkulutus',
            type: 'switch+text',
            onChange2: watchHandler,
            noValidate2: true,
            disabled:
"template.models.tilojenLammontuottoApulaitteidenKulutus.model2 === false",
            prefix: "e",
            prefixSub: "tuotto, tilat",
            suffix: "kWh/(m²a)",
            bind: {
              name: "lammitysjarjestelma",
              value: "tilojenLammontuottoApulaitteidenKulutus"
            },
            bind2: {
              name: "lammitysjarjestelma",
              value: "tilojenLammontuottoApulaitteidenKulutus2"
            },
            source: 'YM asetus 176/2013, liite 1, taulukko 10:
sähköpatterilämmitys s.12'
          },
          lammitysenergianTuottoVesiHyoty: {
            name: 'lammitysenergianTuottoVesiHyoty',
            header: "Käytä omavalintaa",
            header2: 'Lämmitysenergian tuoton hyötysuhde käyttöveden
lämmityksessä',
            type: 'switch+text',
            onChange2: watchHandler,
            noValidate2: true,
            disabled: "template.models.lammitysenergianTuottoVesiHyoty.model2 ===
false",
            prefix: "η",
```

```
            prefixSub: "tuotto, lkv",
            bind: {
              name: "lammitysjarjestelma",
              value: "lammitysenergianTuottoVesiHyoty"
            },
            bind2: {
              name: "lammitysjarjestelma",
              value: "lammitysenergianTuottoVesiHyoty2"
            },
            source: 'Käyttövesivaraajan häviöt lasketaan erikseen'
          },
          kayttovesiLammontuottoKulutus: {
            name: 'kayttovesiLammontuottoKulutus',
            header: "Käytä omavalintaa",
            header2: 'Käyttöveden lämmöntuotantojärjestelmän apulaitteiden
sähkönkulutus',
            type: 'switch+text',
            disabled: "template.models.kayttovesiLammontuottoKulutus.model2 ===
false",
            onChange2: watchHandler,
            noValidate2: true,
            prefix: "e",
            prefixSub: "tuotto, lkv",
            suffix: 'kWh/(m²a)',
            bind: {
              name: "lammitysjarjestelma",
              value: "kayttovesiLammontuottoKulutus"
            },
            bind2: {
              name: "lammitysjarjestelma",
              value: "kayttovesiLammontuottoKulutus2"
            },
            source: 'Ei apulaitteita'
          },
          varaavaTulisijaHyoty: {
            name: 'varaavaTulisijaHyoty',
            header: 'Varaavan tulisijan kokonaishyötysuhde',
            type: 'text',
            prefix: "η",
            prefixSub: "tulisija",
            bind: {
              name: "lammitysjarjestelma",
              value: "varaavaTulisijaHyoty"
            },
            bind2: {
              name: "lammitysjarjestelma",
              value: "varaavaTulisijaHyoty2"
            },
            source: 'YM asetus 176/2013, liite 1, kohta 2.3.1 s.15 oletus 0,60
jos ei parempaa tietoa'
          }
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 39

LammonJakelunHaviot Controller (lammonJakelunHaviot.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('lammonJakelunHaviotCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;

      var changeFn = function () {
        $timeout(function () {
          vm.template.models.lammonJakelujarjestelmanLampohavioKylmaTila.model =

helper.getValue("lammonJakelunHaviot.lammonJakelujarjestelmanLampohavioKylmaTila"
);
        });
      };
      vm.valueArrays = {
        olosuhteet: data.olosuhteet
      };
      vm.models = {};
      vm.template = {
        name: 'lammonJakelunHaviot',
        readable: 'Lämmön jakelujärjestelmän häviöt lämmittämättömiin tiloihin',
        models: {
          menoPaluuPutketPituus: {
            name: 'menoPaluuPutketPituus',
            header: 'Meno- ja paluuputkien yhteenlaskettu pituus kylmässä
tilassa',
            type: 'text',
            onChange: changeFn,
            prefix: "L",
            suffix: "m",
            bind: {
              name: "lammonJakelunHaviot",
              value: "menoPaluuPutketPituus"
            },
            source: ''
          },
          olosuhteet: {
            name: 'olosuhteet',
            header: 'Olosuhteet',
            type: 'select',
            onChange: changeFn,
            bind: {
              name: "lammonJakelunHaviot",
              value: "olosuhteet"
            },
            options: 'olosuhde as olosuhde.name for olosuhde in
valueArrays.olosuhteet ' +
            'track by olosuhde.name',
            source: ''
          },
          lammonJakelujarjestelmanLampohavioKylmaTila: {
            name: 'lammonJakelujarjestelmanLampohavioKylmaTila',
            header: 'Lämmön jakelujärjestelmän lämpöhäviö kylmään tilaan',
            type: 'text',
            disabled: "true",
            prefix: "Q",
            prefixSub: "jakelu,ulos",
            suffix: "kWh/a",
            bind: {
```

```
            name: "lammonJakelunHaviot",
            value: "lammonJakelujarjestelmanLampohavioKylmaTila"
          },
          source: 'D5 taul. 6.1 s.39'
        },
        lammonJakelujarjestelmanVarastointiLampohavio: {
          name: 'lammonJakelujarjestelmanVarastointiLampohavio',
          header: 'Lämmön jakelujärjestelmän varastoinnin lämpöhäviö',
          type: 'text',
          prefix: "Q",
          prefixSub: "varastointi,ulos",
          suffix: "kWh/a",
          bind: {
            name: "lammonJakelunHaviot",
            value: "lammonJakelujarjestelmanVarastointiLampohavio"
          },
          source: ''
        },
      },
    };
    helper.mapDefaultValues(vm, vm.template, true);
  }]);
});
```

Appendix 40

Lampoarvolaskuri Controller (lampoarvolaskuri.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('lampoarvolaskuriCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;
      $timeout(helper.zebraRows.bind(null, document, ".row", "#EEEEEE",
"#CCCCCC", "rowCont"));
      var changeFn = function () {
        $timeout(function () {
          this.maara = helper.getValue("lampoarvolaskuri." + this.name +
"Maara");
          this.maara2 = helper.getValue("lampoarvolaskuri." + this.name +
"Maara2");
        }.bind(this));
      };
      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'lampoarvolaskuri',
        models: {
          raskasOljy: {
            name: 'raskasOljy',
            header: 'Raskas polttoöljy',
            type: 'text',
            suffix: "kg",
            suffix2: "kg",
            bind: {
              name: "lampoarvolaskuri",
              value: "raskasOljy"
            },
            bind2: {
              name: "lampoarvolaskuri",
              value: "raskasOljy2"
            },
            maara: DataFactory.lampoarvolaskuri.raskasOljyMaara,
            maara2: DataFactory.lampoarvolaskuri.raskasOljyMaara2
          },
          kevytOljy: {
            name: 'kevytOljy',
            header: 'Kevyt polttoöljy',
            type: 'text',
            suffix: "litraa",
            suffix2: "litraa",
            bind: {
              name: "lampoarvolaskuri",
              value: "kevytOljy"
            },
            bind2: {
              name: "lampoarvolaskuri",
              value: "kevytOljy2"
            },
            maara: DataFactory.lampoarvolaskuri.kevytOljyMaara,
            maara2: DataFactory.lampoarvolaskuri.kevytOljyMaara2
          },
          maakaasu: {
            name: 'maakaasu',
            header: 'Maakaasu',
            type: 'text',
            suffix: "m³n",
```

```
    suffix2: "m³n",
    bind: {
      name: "lampoarvolaskuri",
      value: "maakaasu"
    },
    bind2: {
      name: "lampoarvolaskuri",
      value: "maakaasu2"
    },
    maara: DataFactory.lampoarvolaskuri.maakaasuMaara,
    maara2: DataFactory.lampoarvolaskuri.maakaasuMaara2
  },
  polttopuu: {
    name: 'polttopuu',
    header: 'Polttopuu yleensä (pilkkeet)',
    type: 'text',
    suffix: "kg",
    suffix2: "kg",
    bind: {
      name: "lampoarvolaskuri",
      value: "polttopuu"
    },
    bind2: {
      name: "lampoarvolaskuri",
      value: "polttopuu2"
    },
    maara: DataFactory.lampoarvolaskuri.polttopuuMaara,
    maara2: DataFactory.lampoarvolaskuri.polttopuuMaara2
  },
  pilkkeetHavu: {
    name: 'pilkkeetHavu',
    header: 'Pilkkeet (havu- ja sekapuu)',
    type: 'text',
    suffix: "pino-m³",
    suffix2: "pino-m³",
    bind: {
      name: "lampoarvolaskuri",
      value: "pilkkeetHavu"
    },
    bind2: {
      name: "lampoarvolaskuri",
      value: "pilkkeetHavu2"
    },
    maara: DataFactory.lampoarvolaskuri.pilkkeetHavuMaara,
    maara2: DataFactory.lampoarvolaskuri.pilkkeetHavuMaara2
  },
  pilkkeetKoivu: {
    name: 'pilkkeetKoivu',
    header: 'Pilkkeet (koivu)',
    type: 'text',
    suffix: "pino-m³",
    suffix2: "pino-m³",
    bind: {
      name: "lampoarvolaskuri",
      value: "pilkkeetKoivu"
    },
    bind2: {
      name: "lampoarvolaskuri",
      value: "pilkkeetKoivu2"
    },
    maara: DataFactory.lampoarvolaskuri.pilkkeetKoivuMaara,
    maara2: DataFactory.lampoarvolaskuri.pilkkeetKoivuMaara2
  },
  puupelletit: {
```

```
    name: 'puupelletit',
    header: 'Puupelletit',
    type: 'text',
    suffix: "kg",
    suffix2: "kg",
    bind: {
      name: "lampoarvolaskuri",
      value: "puupelletit"
    },
    bind2: {
      name: "lampoarvolaskuri",
      value: "puupelletit2"
    },
    maara: DataFactory.lampoarvolaskuri.puupelletitMaara,
    maara2: DataFactory.lampoarvolaskuri.puupelletitMaara2
  },
  polttohake: {
    name: 'polttohake',
    header: 'Polttohake',
    type: 'text',
    suffix: "irto-m³",
    suffix2: "irto-m³",
    bind: {
      name: "lampoarvolaskuri",
      value: "polttohake"
    },
    bind2: {
      name: "lampoarvolaskuri",
      value: "polttohake2"
    },
    maara: DataFactory.lampoarvolaskuri.polttohakeMaara,
    maara2: DataFactory.lampoarvolaskuri.polttohakeMaara2
  },
  kivihiili: {
    name: 'kivihiili',
    header: 'Kivihiili',
    type: 'text',
    suffix: "kg",
    suffix2: "kg",
    bind: {
      name: "lampoarvolaskuri",
      value: "kivihiili"
    },
    bind2: {
      name: "lampoarvolaskuri",
      value: "kivihiili2"
    },
    maara: DataFactory.lampoarvolaskuri.kivihiiliMaara,
    maara2: DataFactory.lampoarvolaskuri.kivihiiliMaara2
  },
  palaturve: {
    name: 'palaturve ',
    header: 'Palaturve',
    type: 'text',
    suffix: "kg",
    suffix2: "kg",
    bind: {
      name: "lampoarvolaskuri",
      value: "palaturve"
    },
    bind2: {
      name: "lampoarvolaskuri",
      value: "palaturve2"
    },
```

```
            maara: DataFactory.lampoarvolaskuri.palaturveMaara,
            maara2: DataFactory.lampoarvolaskuri.palaturveMaara2
          },
          puubriketit: {
            name: 'puubriketit ',
            header: 'Puubriketit',
            type: 'text',
            suffix: "kg",
            suffix2: "kg",
            bind: {
              name: "lampoarvolaskuri",
              value: "puubriketit"
            },
            bind2: {
              name: "lampoarvolaskuri",
              value: "puubriketit2"
            },
            maara: DataFactory.lampoarvolaskuri.puubriketitMaara,
            maara2: DataFactory.lampoarvolaskuri.puubriketitMaara2
          }
        }
      };
      var keys = Object.keys(vm.template.models);
      for (var key of keys) {
        vm.template.models[key].onChange =
changeFn.bind(vm.template.models[key]);
      }
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 41

LampopumppuLammitysmuotona Controller (lampopumppuLammitysmuotona.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('lampopumppuLammitysmuotonaCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;
      vm.valueArrays = {
        lampopumpunLaji: [
          "Maalämpöpumppu (MLP)",
          "Ilma-vesi lämpöpumppu (IVLP)",
          "Poistoilmalämpöpumppu (PILP)"
        ]
      };
      vm.models = {};
      vm.template = {
        name: 'lampopumppuLammitysmuotona',
        readable: 'Lämpöpumppu lämmitysmuotona',
        models: {
          lampopumpunLaji: {
            name: 'lampopumpunLaji',
            header: 'Lämpöpumpun laji',
            type: 'select',
            bind: {
              name: "lampopumppuLammitysmuotona",
              value: "lampopumpunLaji"
            },
            options: 'pumppu for pumppu in valueArrays.lampopumpunLaji',
            source: ''
          },
          lampopumpunTuottamaOsuus: {
            name: 'lampopumpunTuottamaOsuus',
            header: 'Lämpöpumpun tuottama osuus tilojen ja lämpimän käyttöveden
lämmitystarpeesta',
            type: 'text',
            suffix: "%",
            bind: {
              name: "lampopumppuLammitysmuotona",
              value: "lampopumpunTuottamaOsuus"
            },
            source: 'Luetaan taulukosta'
          },
          lampopumpunKausisuorityskykykerroinTilat: {
            name: 'lampopumpunKausisuorityskykykerroinTilat',
            header: 'Lämpöpumpun kausisuosituskykykerroin tilojen lämmityksessä',
            type: 'text',
            prefix: "SPF",
            prefixSub: "tilat",
            bind: {
              name: "lampopumppuLammitysmuotona",
              value: "lampopumpunKausisuorityskykykerroinTilat"
            },
            source: ''
          },
          lampopumpunKausisuorityskykykerroinVesi: {
            name: 'lampopumpunKausisuorityskykykerroinVesi',
            header: 'Lämpöpumpun kausisuosituskykykerroin käyttöveden
lämmityksessä',
            type: 'text',
            prefix: "SPF",
            prefixSub: "lkv",
```

```
            bind: {
              name: "lampopumppuLammitysmuotona",
              value: "lampopumpunKausisuorityskykykerroinVesi"
            },
            source: ''
          },
          lampopumpunNimellisteho: {
            name: 'lampopumpunNimellisteho',
            header: 'Lämpöpumpun nimellisteho',
            type: 'text',
            disabledValue:
helper.getValue("lampopumppuLammitysmuotona.lampopumpunNimellistehoDisabled"),
            disabled: "template.models.lampopumpunNimellisteho.disabledValue ===
true",
            prefix: "ϕ",
            prefixSub: "LP",
            bind: {
              name: "lampopumppuLammitysmuotona",
              value: "lampopumpunNimellisteho"
            },
            source: ''
          }
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 42

Lisalammitysjarjestelmia Controller (lisalammitysjarjestelmia.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('lisalammitysjarjestelmiaCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;

      var changeFn = function () {
        $timeout(function () {
          vm.template.models.pumppujenTeho.model =
helper.getValue("lisalammitysjarjestelmia.pumppujenTeho");
        });
      };

      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'lisalammitysjarjestelmia',
        readable: 'Lisälämmitysjärjestelmiä uusiutuvan omavaraisenergian
tuottamiseen',
        models: {
          aurinkokerainSeparate: {
            header: "Aurinkokeräin käyttöveden lämmityksessä, max 40% tarpeesta
E-laskennassa",
            type: "separator"
          },
          suuntauskerroin: {
            name: 'suuntauskerroin',
            header: 'Suuntauskerroin',
            type: 'text',
            prefix: 'k',
            bind: {
              name: "lisalammitysjarjestelmia",
              value: "suuntauskerroin"
            },
            source: 'D5 s.46-47'
          },
          pintaAla: {
            name: 'pintaAla',
            header: 'Pinta-ala',
            type: 'text',
            prefix: 'A',
            suffix: 'm²',
            onChange: changeFn,
            bind: {
              name: "lisalammitysjarjestelmia",
              value: "pintaAla"
            },
            source: 'Taul6.9'
          },
          energiatuotto: {
            name: 'energiatuotto',
            header: 'Energiantuotto käyttöveteen pinta-alaa kohti',
            type: 'text',
            prefix: 'q',
            suffix: 'kWh/(m² a)',
            bind: {
              name: "lisalammitysjarjestelmia",
              value: "energiatuotto"
            },
```

```
            source: 'Taul6.8'
          },
          pumppujenTeho: {
            name: 'pumppujenTeho',
            header: 'Pumppujen teho, W',
            type: 'text',
            prefix: 'P',
            suffix: 'W',
            bind: {
              name: "lisalammitysjarjestelmia",
              value: "pumppujenTeho"
            },
            source: '6.12'
          },
          pumpunKayttoaika: {
            name: 'pumpunKayttoaika',
            header: 'Pumpun käyttöaika',
            type: 'text',
            prefix: 't',
            suffix: 'h',
            bind: {
              name: "lisalammitysjarjestelmia",
              value: "pumpunKayttoaika"
            },
            source: 's.47'
          },
          sep: {
            header: "",
            type: "separator"
          },
          aurinkojarjestelmaEnergia: {
            name: 'aurinkojarjestelmaEnergia',
            header: 'Aurinkosähköjärjestelmän tuottama sähköenergi käyttöveden
lämmittämiseen. ' +
            'Käytä valmistajan ilmoittamaa lukua.',
            type: 'text',
            prefix: 'W',
            prefixSub: 'pv',
            suffix: 'kWh/a',
            bind: {
              name: "lisalammitysjarjestelmia",
              value: "aurinkojarjestelmaEnergia"
            },
            source: 's.47'
          },
          tuulivoimala: {
            name: 'tuulivoimala',
            header: 'Tuulivoimalan tuottama sähköenergia. Käytä valmistajan
ilmoittamaa lukua.',
            type: 'text',
            prefix: 'W',
            prefixSub: 'tuuli',
            suffix: 'kWh/a',
            bind: {
              name: "lisalammitysjarjestelmia",
              value: "tuulivoimala"
            },
            source: 's.47'
          }
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 43

Main page Controller (main.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
    'use strict';
    controllers.controller('mainCtrl', ["$scope", "$timeout", "$window",
"$rootScope",
        function ($scope, $timeout, $window, $rootScope) {
            var vm = $scope;
            vm.save = helper.save;
            vm.uploadedFile;
            vm.$watch(helper.getFiles, function(files){
                vm.datas = files;
            }, true);
            vm.$watch(helper.getCurFile, function (file) {
                vm.currentFile = file;
                $timeout(function(){$scope.$digest();});
            }, true);
            vm.error;
            vm.Math = $window.Math;
            vm.renameFile = function(file){
                var newName = prompt("Anna uusi nimi");
                if(newName){
                    helper.renameFile(file, newName);
                }
            };
            vm.addFix = function () {
                helper.addFile({
                    name: "Korjaus " + vm.datas.files.length,
                    value: JSON.parse(JSON.stringify(vm.currentFile.value))
                });
            };
            vm.removeFile = function (file) {
                helper.removeFile(file);
            };
            vm.swapFile = function (file) {
                helper.setValues(file);
                $rootScope.$broadcast("calculate");
            };
            vm.$watch("uploadedFile", function (value) {
                if (value) {
                    try {
                        helper.addFile({
                            name: value.name,
                            value: JSON.parse(value.result).value
                        });
                        $timeout(function(){$rootScope.$digest();});
                    }
                    catch (e) {
                        vm.error = "Json on huonolaatuista";
                    }
                }
            });
            //  Math.round = function(val){return val;};
        }]);
});
```

Appendix 44

Angular main controller (mainController.js)

```
define(['./module', 'helper', 'json', 'laskenta'], function (controllers, helper,
data, laskenta) {
    'use strict';
    controllers.controller('mainController', ["$scope", "$timeout", "$compile",
function ($scope, $timeout, $compile) {
      var vm = $scope;
    //  Math.round = function(val){return val;};
      vm.controllers = {};
      vm.models = {};
      vm.showTooltip = helper.showTooltip;
      vm.hideTooltip = helper.hideTooltip;
    }]);
});
```

Appendix 45

Perussuureet Controller (perussuureet.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('perussuureetCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;

      var watchHandler = function (value) {
        if (this.model2 === false) {
          // need to wait for change in dataFactory first
          $timeout(function () { this.model = helper.getValue("perussuureet." +
this.name); }.bind(this));
        }
      };

      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'perussuureet',
        readable: 'Perussuureet',
        models: {
          lammitettyNettoala: {
            name: 'lammitettyNettoala',
            header: 'Lämmitetty nettoala',
            type: 'text',
            suffix: "m²",
            prefix: "A",
            prefixSub: "netto",
            bind: {
              name: "perussuureet",
              value: "lammitettyNettoala"
            },
            source: 'havainnointi paikanpäällä'
          },
          rakennuksenIlmatilavuus: {
            name: 'rakennuksenIlmatilavuus',
            header: 'Rakennuksen ilmatilavuus',
            type: 'text',
            suffix: "m³",
            prefix: "V",
            bind: {
              name: "perussuureet",
              value: "rakennuksenIlmatilavuus"
            },
            source: 'ok2000-talossa vaipan alan lukuarvo = tilavuuden lukuarvo'
          },
          sisalampotila: {
            name: 'sisalampotila',
            header: 'Sisälämpötila',
            type: 'switch+text',
            onChange2: watchHandler,
            noValidate2: true,
            disabled: "template.models.sisalampotila.model2 === false",
            prefix: "T",
            prefixSub: "s",
            suffix: "°C",
            bind: {
              name: "perussuureet",
              value: "sisalampotila"
            },
```

```
            bind2: {
              name: "perussuureet",
              value: "sisalampotilaIgnore"
            },
            source: 'D3/2012 taulukko 2 s.18'
          },
          jaahdytysraja: {
            name: 'jaahdytysraja',
            header: 'jäähdytysraja',
            type: 'text',
            disabled: "true",
            prefix: "T",
            prefixSub: "s",
            suffix: "°C",
            bind: {
              name: "perussuureet",
              value: "jaahdytysraja"
            },
            source: 'D3/2012 taulukko 2 s.18'
          },
          rakennuksenIlmanvuoto: {
            name: 'rakennuksenIlmanvuoto',
            header: 'Rakennuksen ilmanvuotoluku',
            type: 'switch+text',
            onChange2: watchHandler,
            noValidate2: true,
            disabled: "template.models.rakennuksenIlmanvuoto.model2 === false",
            prefix: 'q',
            prefixSub: '50',
            suffix: 'm³/ (hm²)',
            bind: {
              name: "perussuureet",
              value: "rakennuksenIlmanvuoto"
            },
            bind2: {
              name: "perussuureet",
              value: "ilmanvuotoIgnore"
            },
            source: 'YM asetus 176/2013, liite 1, taulukko 4 s.9 Tyyppiarvoja D5
taul. 3.5 s.20'
          }
        },

      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 46

Perustiedot Controller (perustiedot.js)

```
define(['./module', 'helper', 'json', 'laskenta'], function (controllers, helper, data, laskenta) {
  'use strict';
  controllers.controller('perustiedotCtrl', ["$scope", "$timeout", "DataFactory",
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;
      var firstYear = 1850;
      var maxResidents = 10;
      var changeStructureType = function () {
        if (vm.template.models.rakennetyyppi.model.value) {
          vm.template.models.lampokapasiteetti.model = vm.template.models.rakennetyyppi.model.value;
        }
      };
      var changeKunta = function () {
        vm.template.models.saavyohyke.model = helper.getValue("perustiedot.saavyohyke");
      };
      vm.valueArrays = {
        cities: data.counties,
        years: helper.arrayInRange(new Date().getFullYear() - firstYear + 1, firstYear).reverse(),
        residentAmount: helper.arrayInRange(maxResidents + 1),
        floorsAmount: helper.arrayInRange(20, 1),
        usageClasses: data.houseTypes,
        structureTypes: data.structureTypes
      };
      vm.models = { asukasmaara: null };
      vm.template = {
        name: "perustiedot",
        readable: 'Perustiedot',
        models: {
          separator2: {
            header: "Tilaajan tiedot",
            type: "separator"
          },
          tilaaja: {
            name: "tilaaja",
            header: "Tilaaja",
            type: "textWide",
            noValidate: true,
            bind: {
              name: "perustiedot",
              value: "tilaaja"
            },
```

```
    source: ""
},
yritys: {
    name: "yritys",
    header: "Yritys",
    type: "textWide",
    noValidate: true,
    bind: {
        name: "perustiedot",
        value: "yritys"
    },
    source: ""
},
separate: {
    header: "Osoitetiedot",
    type: "separator"
},
katuosoite: {
    name: "katuosoite",
    header: "Katuosoite",
    type: "textWide",
    noValidate: true,
    bind: {
        name: "perustiedot",
        value: "katuosoite"
    },
    source: ""
},
postinumero: {
    name: "postinumero",
    header: "Postinumero",
    type: "textWide",
    noValidate: true,
    bind: {
        name: "perustiedot",
        value: "postinumero"
    },
    source: ""
},
postitoimipaikka: {
    name: "postitoimipaikka",
    header: "Postitoimipaikka",
    type: "textWide",
    noValidate: true,
    bind: {
        name: "perustiedot",
        value: "postitoimipaikka"
    },
```

```
    source: ""
},
separate3: {
    header: "Toteutunut energiankulutus",
    type: "separator"
},
seurantavuosi: {
    name: 'seurantavuosi',
    header: 'Seurantavuosi',
    type: 'text',
    bind: {
        name: "toteutunutEnergiankulutus",
        value: "seurantavuosi"
    },
    source: ''
},
sahko: {
    name: 'sahko',
    header: 'Sähkö yhteensä',
    type: 'text',
    suffix: "kWh/a",
    bind: {
        name: "toteutunutEnergiankulutus",
        value: "sahko"
    },
    source: ''
},
fossiiliset: {
    name: 'fossiiliset',
    header: 'Fossiiliset polttoaineet',
    type: 'text',
    suffix: "kWh/a",
    bind: {
        name: "toteutunutEnergiankulutus",
        value: "fossiiliset"
    },
    source: ''
},
uusiutuvat: {
    name: 'uusiutuvat',
    header: 'Uusiutuvat polttoaineet',
    type: 'text',
    suffix: "kWh/a",
    bind: {
        name: "toteutunutEnergiankulutus",
        value: "uusiutuvat"
    },
    source: ''
```

```
            },
            separate2: {
              header: "Perustiedot",
              type: "separator"
            },
            sijaintikunta: {
              name: "sijaintikunta",
              header: "Sijaintikunta",
              type: "select",
              bind: {
                name: "perustiedot",
                value: "sijaintikunta"
              },
              onChange: changeKunta,
              options: "city as city.kunta for city in valueArrays.cities track by city.kunta
",
              source: ""
            },
            rakennusluvanVuosi: {
              name: "rakennusluvanVuosi",
              header: "Rakennusluvan vireilletulovuosi",
              type: "select",
              bind: {
                name: "perustiedot",
                value: "rakennusluvanVuosi"
              },
              options: "year as year for year in valueArrays.years",
              source: "Rakennuksen asiakirjat"
            },
            valmistumisvuosi: {
              name: "valmistumisvuosi",
              header: "Valmistumisvuosi",
              type: "select",
              options: "year as year for year in valueArrays.years",
              bind: {
                name: "perustiedot",
                value: "valmistumisvuosi"
              },
              source: "Rakennuksen asiakirjat"
            },
            asukasmaara: {
              name: "asukasmaara",
              header: "Asukasmäärä",
              type: "select",
              bind: {
                name: "perustiedot",
                value: "asukasmaara"
              },
```

```
            options: "resident as resident for resident in
valueArrays.residentAmount",
            source: "Keskustelut"
          },
          saavyohyke: {
            name: "saavyohyke",
            header: "Säävyöhyke",
            type: "text",
            disabled: "true",
            bind: {
              name: "perustiedot",
              value: "saavyohyke"
            },
            source: "D3/2012 kuva L2.1 s.29"
          },
          kayttotarkoitusluokka: {
            name: "kayttotarkoitusluokka",
            header: "Käyttötarkoitusluokka",
            type: "selectWide",
            bind: {
              name: "perustiedot",
              value: "kayttotarkoitusluokka"
            },
            options: "class as class.tyyppi for class in valueArrays.usageClasses track
by class.tyyppi",
            source: "YM asetus 176/2013 L2 s.18-22"
          },
          kerrokset: {
            name: "kerrokset",
            header: "Kerrosten lukumäärä",
            type: "select",
            bind: {
              name: "perustiedot",
              value: "kerrokset"
            },
            options: "floor as floor for floor in valueArrays.floorsAmount",
            source: "havainnointi paikanpäällä"
          },
          rakennetyyppi: {
            name: "rakennetyyppi",
            header: "Rakennetyyppi",
            type: "select",
            bind: {
              name: "perustiedot",
              value: "rakennetyyppi"
            },
            options: "type as type.name for type in valueArrays.structureTypes track
by type.name",
```

```
            onChange: changeStructureType,
            source: "havainnointi paikanpäällä"
          },
          lampokapasiteetti: {
            name: "lampokapasiteetti",
            header: "Tehollisen lämpökapasiteetin ominaisarvo kalusteineen",
            type: "text",
            prefix: "C",
            prefixSub: "rak omin",
            suffix: "Wh/(m² K)",
            disabled: "template.models.rakennetyyppi.model.name!='Oma Valinta'",
            bind: {
              name: "perustiedot",
              value: "lampokapasiteetti"
            },
            source: "YM 176/2013 L1 taul.8 s.12"
          }
        },
        extraInfo: {
          visible: false,
          show: function () {
            this.visible = true;
          },
          hide: function () {
            this.visible = false;
          }
        }
      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 47

Rakennusosat Controller (rakennusosat.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('rakennusosatCtrl', ['$scope', '$timeout',
'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;
      $timeout(helper.zebraRows.bind(null, document, ".row", "#EEEEEE",
"#CCCCCC", "rowCont"));
      var calculate = function () {
        this.value = (this.model * this.model2);
      };
      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'rakennusosat',
        readable: 'Rakennusosat',
        models: {
          ulkoseinatUlkoilmaan: {
            name: 'ulkoseinatUlkoilmaan',
            header: 'Ulkoseinät ulkoilmaan',
            type: 'text',
            tila: "Ulkolämpötila",
            onChange: calculate,
            bind: {
              name: "rakennusosat",
              value: "ulkoseinatUlkoilmaan"
            },
            bind2: {
              name: "rakennusosat",
              value: "ulkoseinatUlkoilmaan2"
            },
            value: DataFactory.rakennusosat.ulkoseinatUlkoilmaanValue,
            source: 'Piirrustukset'
          },
          ylapohja: {
            name: 'ylapohja',
            header: 'Yläpohja',
            type: 'text',
            tila: "Ulkolämpötila",
            onChange: calculate,
            bind: {
              name: "rakennusosat",
              value: "ylapohja"
            },
            bind2: {
              name: "rakennusosat",
              value: "ylapohja2"
            },
            value: DataFactory.rakennusosat.ylapohjaValue,
            source: 'Piirrustukset'
          },
          alapohjaMaataVasten: {
            name: 'alapohjaMaataVasten',
            header: 'Alapohja, maata vasten',
            type: 'text',
            tila: "Maaperä",
            onChange: calculate,
            bind: {
              name: "rakennusosat",
              value: "alapohjaMaataVasten"
```

```
  },
  bind2: {
    name: "rakennusosat",
    value: "alapohjaMaataVasten2"
  },
  value: DataFactory.rakennusosat.alapohjaMaataVastenValue,
  source: 'Piirrustukset'
},
alapohjaTuulettuva: {
  name: 'alapohjaTuulettuva',
  header: 'Alapohja, tuulettuva tai ulkoilmaa vasten',
  type: 'text',
  tila: "Ulkolämpötila",
  onChange: calculate,
  bind: {
    name: "rakennusosat",
    value: "alapohjaTuulettuva"
  },
  bind2: {
    name: "rakennusosat",
    value: "alapohjaTuulettuva2"
  },
  value: DataFactory.rakennusosat.alapohjaTuulettuvaValue,
  source: ''
},
alapohjaYhteensa: {
  name: 'alapohjaYhteensa',
  header: 'Alapohja, yhteensä',
  type: 'text',
  disabled: true,
  disabled2: true,
  tila: "",
  onChange: calculate,
  bind: {
    name: "rakennusosat",
    value: "alapohjaYhteensa"
  },
  bind2: {
    name: "rakennusosat",
    value: "alapohjaYhteensa2"
  },
  value: DataFactory.rakennusosat.alapohjaYhteensaValue,
  source: ''
},
ikkunat: {
  name: 'ikkunat',
  header: 'Ikkunat',
  type: 'text',
  disabled: "true",
  disabled2: "true",
  tila: "Ulkolämpötila",
  onChange: calculate,
  bind: {
    name: "rakennusosat",
    value: "ikkunat"
  },
  bind2: {
    name: "rakennusosat",
    value: "ikkunat2"
  },
  value: DataFactory.rakennusosat.ikkunatValue,
  source: 'piirrustukset'
},
ovet: {
```

```
            name: 'ovet',
            header: 'Ovet',
            type: 'text',
            tila: "Ulkolämpötila",
            onChange: calculate,
            bind: {
              name: "rakennusosat",
              value: "ovet"
            },
            bind2: {
              name: "rakennusosat",
              value: "ovet2"
            },
            value: DataFactory.rakennusosat.ovetValue,
            source: 'Havainnointi paikanpäällä'
          }
        },
      };
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 48

Tilojen-lammitysjarjestelma Controller (Tilojen-lammitysjarjestelma.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  var fireplaces = 4;
  controllers.controller('tilojenLammitysjarjestelmaCtrl', ["$scope", "$timeout",
"DataFactory",
    function controller($scope, $timeout, DataFactory) {
      var vm = $scope;

      var fireplaceChange = function () {
        if (vm.template.models.tulisijojenKaytto.model.name == "Standardikäyttö")
{
          vm.template.models.tulisijojenKaytto.model2 =
data.fireplaceUse[0].value;
        }
      };

      vm.valueArrays = {
        heatingTypes: data.heatingTypes,
        heatDivisionSystems: data.heatDivisionSystems,
        fireplacesAmount: helper.arrayInRange(fireplaces),
        fireplaceUse: data.fireplaceUse
      };
      vm.models = {};
      vm.template = {
        name: "tilojenLammitysjarjestelma",
        readable: 'Tilojen lämmitysjärjestelmä',
        models: {
          lammitysjarjestelma: {
            name: "tilojenLammitysjarjestelma",
            header: "Tilojen lämmitysjärjestelmä",
            type: "select",
            bind: {
              name: "tilojenLammitysjarjestelma",
              value: "lammitysjarjestelma"
            },
            options: "type as type.nimi for type in valueArrays.heatingTypes
track by type.nimi",
            source: "havainnointi paikanpäällä ja Ym asetus 176/2013 taul.10 ja
11 s.13"
          },
          lammonjakojarjestelma: {
            name: "tilojenLammonjakojarjestelma",
            header: "Tilojen lämmönjakojärjestelmä",
            type: "selectWide",
            bind: {
              name: "tilojenLammitysjarjestelma",
              value: "lammonjakojarjestelma"
            },
            options: "type as type.name for type in
valueArrays.heatDivisionSystems track by type.name",
            source: "Ym asetus 176/2013 taul.9 s.12"
          },
          tulisijat: {
            name: "tulisijat",
            header: "Varaavien tulisijojen määrä",
            type: "select",
            bind: {
              name: "tilojenLammitysjarjestelma",
              value: "tulisijat"
```

```
            },
            options: "type for type in valueArrays.fireplacesAmount",
            source: "Havainnointi paikan päällä"
          },
          tulisijojenKaytto: {
            name: "tulisijojenKaytto",
            name2: "tulisijojenKayttoText",
            header: "Tilojen lämmitykseen luovutettu lämpömäärä tulisijaa
kohden",
            type: "select+text",
            onChange: fireplaceChange,
            suffix2: "kWh/a",
            noValidate: true,
            disabled2: "template.models.tulisijojenKaytto.model.name!='Oma
Valinta'",
            bind: {
              name: "tilojenLammitysjarjestelma",
              value: "tulisijojenKaytto"
            },
            bind2: {
              name: "tilojenLammitysjarjestelma",
              value: "tulisijojenKaytto2"
            },
            options: "use as use.name for use in valueArrays.fireplaceUse track
by use.name",
            source: "YM asetus 176/2013 L1 kohta 2.3 s.15"
          }
        }
      };
      helper.mapDefaultValues(vm, vm.template, true);

    }]);
});
```

Appendix 49

VaraavienTulisijojenPolttoaineet Controller (varaavienTulisijojenPolttoaineet.js)

```
define(['./module', 'helper', 'json'], function (controllers, helper, data) {
  'use strict';
  controllers.controller('varaavienTulisijojenPolttoaineetCtrl', ['$scope',
'$timeout', 'DataFactory',
    function ($scope, $timeout, DataFactory) {
      var vm = $scope;
      $timeout(helper.zebraRows.bind(null, document, ".row", "#EEEEEE",
"#CCCCCC", "rowCont"));
      var changeFn = function () {
        $timeout(function () {
          this.tarvittava = helper.getValue("varaavienTulisijojenPolttoaineet." +
this.name + "Tarvittava");
          this.ostoenergia = helper.getValue("varaavienTulisijojenPolttoaineet."
+ this.name + "Ostoenergia");
          this.tuotto = helper.getValue("varaavienTulisijojenPolttoaineet." +
this.name + "Tuotto");
          this.ostoenergia2 = helper.getValue("varaavienTulisijojenPolttoaineet."
+ this.name + "Ostoenergia2");
        }.bind(this));
      };

      vm.valueArrays = {};
      vm.models = {};
      vm.template = {
        name: 'varaavienTulisijojenPolttoaineet',
        readable: 'Varaavien tulisijojen polttoaineen laskenta',
        models: {
          polttopuu: {
            name: 'polttopuu',
            header: 'Polttopuu yleensä (pilkkeet)',
            type: 'text',
            suffix: "kg",
            suffix2: "kg",
            bind: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "polttopuu"
            },
            bind2: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "polttopuu2"
            },
            tarvittava:
DataFactory.varaavienTulisijojenPolttoaineet.polttopuuTarvittava,
            ostoenergia:
DataFactory.varaavienTulisijojenPolttoaineet.polttopuuOstoenergia,
            tuotto: DataFactory.varaavienTulisijojenPolttoaineet.polttopuuTuotto,
            ostoenergia2:
DataFactory.varaavienTulisijojenPolttoaineet.polttopuuOstoenergia2
          },
          pilkkeetHavu: {
            name: 'pilkkeetHavu',
            header: 'Pilkkeet (havu- ja sekapuu)',
            type: 'text',
            suffix: "pino-m³",
            suffix2: "pino-m³",
            bind: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "pilkkeetHavu"
            },
            bind2: {
```

```
              name: "varaavienTulisijojenPolttoaineet",
              value: "pilkkeetHavu2"
            },
            tarvittava:
DataFactory.varaavienTulisijojenPolttoaineet.pilkkeetHavuTarvittava,
            ostoenergia:
DataFactory.varaavienTulisijojenPolttoaineet.pilkkeetHavuOstoenergia,
            tuotto:
DataFactory.varaavienTulisijojenPolttoaineet.pilkkeetHavuTuotto,
            ostoenergia2:
DataFactory.varaavienTulisijojenPolttoaineet.pilkkeetHavuOstoenergia2
          },
          pilkkeetKoivu: {
            name: 'pilkkeetKoivu',
            header: 'Pilkkeet (koivu)',
            type: 'text',
            suffix: "pino-m³",
            suffix2: "pino-m³",
            bind: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "pilkkeetKoivu"
            },
            bind2: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "pilkkeetKoivu2"
            },
            tarvittava:
DataFactory.varaavienTulisijojenPolttoaineet.pilkkeetKoivuTarvittava,
            ostoenergia:
DataFactory.varaavienTulisijojenPolttoaineet.pilkkeetKoivuOstoenergia,
            tuotto:
DataFactory.varaavienTulisijojenPolttoaineet.pilkkeetKoivuTuotto,
            ostoenergia2:
DataFactory.varaavienTulisijojenPolttoaineet.pilkkeetKoivuOstoenergia2
          },
          puupelletit: {
            name: 'puupelletit',
            header: 'Puupelletit',
            type: 'text',
            suffix: "kg",
            suffix2: "kg",
            bind: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "puupelletit"
            },
            bind2: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "puupelletit2"
            },
            tarvittava:
DataFactory.varaavienTulisijojenPolttoaineet.puupelletitTarvittava,
            ostoenergia:
DataFactory.varaavienTulisijojenPolttoaineet.puupelletitOstoenergia,
            tuotto:
DataFactory.varaavienTulisijojenPolttoaineet.puupelletitTuotto,
            ostoenergia2:
DataFactory.varaavienTulisijojenPolttoaineet.puupelletitOstoenergia2
          },
          polttohake: {
            name: 'polttohake',
            header: 'Polttohake',
            type: 'text',
            suffix: "irto-m³",
            suffix2: "irto-m³",
```

```
            bind: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "polttohake"
            },
            bind2: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "polttohake2"
            },
            tarvittava:
DataFactory.varaavienTulisijojenPolttoaineet.polttohakeTarvittava,
            ostoenergia:
DataFactory.varaavienTulisijojenPolttoaineet.polttohakeOstoenergia,
            tuotto:
DataFactory.varaavienTulisijojenPolttoaineet.polttohakeTuotto,
            ostoenergia2:
DataFactory.varaavienTulisijojenPolttoaineet.polttohakeOstoenergia2
          },
          kivihiili: {
            name: 'kivihiili',
            header: 'Kivihiili',
            type: 'text',
            suffix: "kg",
            suffix2: "kg",
            bind: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "kivihiili"
            },
            bind2: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "kivihiili2"
            },
            tarvittava:
DataFactory.varaavienTulisijojenPolttoaineet.kivihiiliTarvittava,
            ostoenergia:
DataFactory.varaavienTulisijojenPolttoaineet.kivihiiliOstoenergia,
            tuotto: DataFactory.varaavienTulisijojenPolttoaineet.kivihiiliTuotto,
            ostoenergia2:
DataFactory.varaavienTulisijojenPolttoaineet.kivihiiliOstoenergia2
          },
          palaturve: {
            name: 'palaturve ',
            header: 'Palaturve',
            type: 'text',
            suffix: "kg",
            suffix2: "kg",
            bind: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "palaturve"
            },
            bind2: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "palaturve2"
            },
            tarvittava:
DataFactory.varaavienTulisijojenPolttoaineet.palaturveTarvittava,
            ostoenergia:
DataFactory.varaavienTulisijojenPolttoaineet.palaturveOstoenergia,
            tuotto: DataFactory.varaavienTulisijojenPolttoaineet.palaturveTuotto,
            ostoenergia2:
DataFactory.varaavienTulisijojenPolttoaineet.palaturveOstoenergia2
          },
          puubriketit: {
            name: 'puubriketit ',
            header: 'Puubriketit',
```

```
            type: 'text',
            suffix: "kg",
            suffix2: "kg",
            bind: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "puubriketit"
            },
            bind2: {
              name: "varaavienTulisijojenPolttoaineet",
              value: "puubriketit2"
            },
            tarvittava:
DataFactory.varaavienTulisijojenPolttoaineet.puubriketitTarvittava,
            ostoenergia:
DataFactory.varaavienTulisijojenPolttoaineet.puubriketitOstoenergia,
            tuotto:
DataFactory.varaavienTulisijojenPolttoaineet.puubriketitTuotto,
            ostoenergia2:
DataFactory.varaavienTulisijojenPolttoaineet.puubriketitOstoenergia2
          }
        }
      };
      var keys = Object.keys(vm.template.models);
      for (var key of keys) {
        vm.template.models[key].onChange =
changeFn.bind(vm.template.models[key]);
      }
      helper.mapDefaultValues(vm, vm.template, true);
    }]);
});
```

Appendix 50

Final calculations Module (laskenta.js)

```
define(['helper', 'json', 'q',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/pintaalaKerrokset.json',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/lammitysenergiaKuukausit
tain.json',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/ikkunatYmparistonVarjost
uskertoimet.json',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/ylapuolinenVarjostus.jso
n',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/sivuVarjostus.json',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/sateilynLapaisy.json',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/kokonaissateilyEnergia.j
son',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/ikkunaSateilyenergia.jso
n',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/lampokuormat.json',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/vaipanJohtumishaviot.jso
n',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/lampokuormienHyodyntamis
aste.json',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/lammitysenergianNettotar
ve.json',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/tilojenLammitys.json',
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/lammitystehonTarve.json’
,
'jsonr!../laskenta/RakennuksenLammitysenergianNettotarve/lammitysjarjestelmaSahko
energianKulutus.json',
    'jsonr!../laskenta/ostonergiankulutus/tilojenLammitys.json',
    'jsonr!../laskenta/ostonergiankulutus/lamminKayttovesi.json',
    'jsonr!../laskenta/ostonergiankulutus/tuloilmanLammitys.json',
    'jsonr!../laskenta/ostonergiankulutus/ilmanvaihtojarjestelma.json',
    'jsonr!../laskenta/ostonergiankulutus/kotitaloussahko.json',
    'jsonr!../laskenta/ostonergiankulutus/varaavatTulisijat.json',
    'jsonr!../laskenta/ostonergiankulutus/summa.json',
    'jsonr!../laskenta/eLuku/sahko.json',
'jsonr!../laskenta/eLuku/kaukolampo.json',
    'jsonr!../laskenta/eLuku/kaukojaahdytys.json',
'jsonr!../laskenta/eLuku/fossiilisetPolttoaineet.json',
    'jsonr!../laskenta/eLuku/uusiutuvatPolttoaineet.json',
'jsonr!../laskenta/eLuku/summa.json'],
    function (helper, data, q, pintaalaKerrokset, lammitysenergiaKuukausittain,
ikkunatYmparistonVarjostuskertoimet,
        ylapuolinenVarjostus, sivuVarjostus, sateilynLapaisy,
kokonaissateilyEnergia, ikkunaSateilyenergia,
        lampokuormat, vaipanJohtumishaviot, lampokuormienHyodyntamisaste,
lammitysenergianNettotarve,
        tilojenLammitys, lammitystehonTarve,
lammitysjarjestelmaSahkoenergianKulutus, tilojenLammitysOsto,
        lamminKayttovesi, tuloilmanLammitys, ilmanvaihtojarjestelmaLa,
kotitaloussahko, varaavatTulisijat,
        ostoenergiaSumma, eLukuSahko, eLukuKaukolampo, eLukuKaukojaahdytys,
eLukuFossiilisetPolttoaineet,
        eLukuUusiutuvatPolttoaineet, eLukuSumma) {
        'use strict';
        var Laskenta = function () {
            var models = {
                pintaalaKerrokset: pintaalaKerrokset,
                lammitysenergiaKuukausittain: lammitysenergiaKuukausittain,
                sateilynLapaisy: sateilynLapaisy,
                ikkunaSateilyenergia: ikkunaSateilyenergia,
                lampokuormat: lampokuormat,
```

```
                vaipanJohtumishaviot: vaipanJohtumishaviot,
                lampokuormienHyodyntamisaste: lampokuormienHyodyntamisaste,
                lammitysenergianNettotarve: lammitysenergianNettotarve,
                tilojenLammitys: tilojenLammitys,
                lammitystehonTarve: lammitystehonTarve,
                lammitysjarjestelmaSahkoenergianKulutus:
lammitysjarjestelmaSahkoenergianKulutus,
                tilojenLammitysOsto: tilojenLammitysOsto,
                lamminKayttovesi: lamminKayttovesi,
                tuloilmanLammitys: tuloilmanLammitys,
                ilmanvaihtojarjestelmaLa: ilmanvaihtojarjestelmaLa,
                kotitaloussahko: kotitaloussahko,
                varaavatTulisijat: varaavatTulisijat,
                ostoenergiaSumma: ostoenergiaSumma,
                eLukuSahko: eLukuSahko,
                eLukuKaukolampo: eLukuKaukolampo,
                eLukuKaukojaahdytys: eLukuKaukojaahdytys,
                eLukuFossiilisetPolttoaineet: eLukuFossiilisetPolttoaineet,
                eLukuUusiutuvatPolttoaineet: eLukuUusiutuvatPolttoaineet,
                eLukuSumma: eLukuSumma
            };
            var calculatedModels = {
                ikkunatYmparistonVarjostuskertoimet:
ikkunatYmparistonVarjostuskertoimet,
                ylapuolinenVarjostus: ylapuolinenVarjostus,
                sivuVarjostus: sivuVarjostus,
                kokonaissateilyEnergia: kokonaissateilyEnergia
            };
            for (var key of Object.keys(calculatedModels)) {
                helper.addToModels(calculatedModels[key], key);
            }
            this.handleJSONs = function () {
                var deferred = q.defer();
                setTimeout(function () {
                    var modelKeys = Object.keys(models);
                    for (var modelKey of modelKeys) {
                        calculatedModels[modelKey] = {};
                        var curModel = models[modelKey];
                        for (var obj of curModel) {
                            calculatedModels[modelKey][obj.name] = {};
                            var objKeys = Object.keys(obj);
                            var i = 0;
                            for (var key of objKeys) {
                                var curValue = obj[key];
                                if (typeof curValue === "object") {
                                    calculatedModels[modelKey][obj.name][key] =
calculate(
                                        curValue,
                                        calculatedModels[modelKey],
                                        i
                                    );
                                }
                                else if (key !== "name") {
                                    calculatedModels[modelKey][obj.name][key] =
curValue;
                                }
                                i++;
                            }
                        }
                        helper.addToModels(calculatedModels[modelKey], modelKey,
true);
                    }
                    deferred.resolve();
                }, 0);
```

```
                return deferred.promise;
            };
            window.calc = this.handleJSONs;
            function calculate(valuesObj, model, ind) {
                var result;
                var getSelfValues = function (str) {
                    var objPath = str.substr(1).split(".");
                    str = model[objPath.shift()];
                    for (var path of objPath) {
                        str = str[path];
                    }
                    return str;
                };
                var calculatedValues = valuesObj.fields.map(function (value) {
                    var ret = value;
                    if (typeof ret === "string") {
                        if (/^#/.test(ret)) {
                            ret = getSelfValues(ret);
                        }
                        else {
                            if
(calculatedModels.hasOwnProperty(ret.match(/(\w+?)\./)[1])) {
                                ret = helper.getValue(ret);
                            }
                            else {
                                ret = helper.getValue(ret);
                            }
                        }
                    }
                    if (typeof ret === "object") {
                        if (ret instanceof Array) {
                            ret = JSON.stringify(ret.map(val =>
                                typeof val === "string" && /^#/.test(val) ?
getSelfValues(val) : val)
                            );
                        }
                        else {
                            ret = JSON.stringify(ret);
                        }
                    }
                    return ret;
                });
                if (valuesObj.formula === "sum") {
                    // Returns the sum of all the calculated values
                    result = calculatedValues.reduce(function (value, previous) {
                        return value + previous;
                    }, 0);
                }
                else if (valuesObj.formula === "avg") {
                    // Returns the average of all the calculated values
                    result = calculatedValues.reduce(function (value, previous) {
                        return value + previous;
                    }, 0) / calculatedValues.length;
                }
                else {
                    var finishedStr = valuesObj.formula.replace(/\$index/g, ind);
                    for (var index = 0; index < calculatedValues.length; index++)
{
                        var repRegex = new RegExp("\\$" + index + "(?!\\d)",
"g");
                        // Replace the nth number on the string with nth value
from the calculatedValues
                        finishedStr = finishedStr.replace(repRegex,
calculatedValues[index]);
```

```
                    }
                    var calcFn = new Function("return " + finishedStr);
                    result = calcFn();
                }
                return result;
            }
        };
        return new Laskenta();

    }
);
```

Appendix 51

Csv-to-json converter program

```
#!/usr/bin/env node
(function () {
    'use strict';
    var commandLineArgs = require('command-line-args');
    var fs = require('fs');
    var parse = require('csv-parse/lib/sync');

    const optionDefinitions = [
        { name: "src", type: String, multiple: true, defaultOption: true },
        { name: "headers", alias: "h", type: String, multiple: true },
        { name: "name", alias: "n", type: String },
        { name: "schema", alias: "s", type: String },
    ];

    const options = commandLineArgs(optionDefinitions);
    var fileData = [];
    var schemaData = [];
    var JSONToSend = {};
    var schema;
    var fileStream;
    var writeStream;
    var schemaStream;

    if (options.schema) {
        try {
            schemaStream = fs.createReadStream(options.schema);
        } catch (e) {
            throw new Error(options.schema + " Not found or is badly formatted");
        }
    }
    else {
        try {
            console.log("No schema file defined, using schema.js");
            schemaStream = fs.createReadStream('./json.schema');
        } catch (e) {
            throw new ReferenceError("schema.js file found");
        }
    }
    if (options.src && options.src[1]) {
        fileStream = fs.createReadStream(options.src[1]);
    }
    else {
        console.log("No csv file defined, looking for main.csv");
        fileStream = fs.createReadStream("./main.csv");
    }
    if (options.src && options.src[0]) {
        writeStream = fs.createWriteStream(options.src[0]);
    }
    else {
        console.log("No output json file defined, writing into output.json");
        writeStream = fs.createWriteStream("./output.json");
    }
    // var schemaStream = fs.createReadStream("./json.schema");
    schemaStream.setEncoding("utf-8");
    schemaStream.on("data", d => schemaData.push(d));
    schemaStream.on("end", function () {
        schema = Function("return "+schemaData.join('').replace(/\n|\r/g,
"").replace(/\"/g, "'"))();
    });
```

```
    var formatBySchema = function (data) {
        var format = [];
        for (let d of data) {
            let obj = {};
            let schemaKeys = Object.keys(schema);
            for (let i = 0; i < d.length; i++) {
                let curKey = schemaKeys.shift();
                let curSchema = schema[curKey];
                if (typeof curSchema === "function") {
                    obj[curKey] = curSchema(d[i]);
                }
                else {
                    obj[curKey] = {};
                    (function handle(schem, obje) {
                        if (typeof schem === "object") {
                            let keys = Object.keys(schem);
                            for (var j = 0, len = keys.length; j < len; j++) {
                                let curKey = keys.shift();
                                let cur = schem[curKey];
                                if (typeof cur === "function") {
                                    obje[curKey] = cur(d[i]);
                                        i++;
                                }
                                else {
                                    obje[curKey] = {};
                                    handle(cur, obje[curKey]);
                                }
                            }
                            i--;
                        }
                    })(curSchema, obj[curKey]);
                }
            }
            format.push(obj);
        }
        return format;
    };

    fileStream.setEncoding("utf-8");
    fileStream.on("data", (d) => fileData.push(d));
    fileStream.on("end", function () {
        var formatted = fileData.join('').replace(/\;{1,}/g, ";");
        var formatted2 = formatBySchema(parse(formatted, { delimiter: ";" }));
        if (options.name) {
            JSONToSend[options.name] = formatted2;
        }
        else {
            JSONToSend = formatted2;
        }
        writeStream.write(JSON.stringify(JSONToSend, null, 2));
        console.log("DONE");
    });
    fileStream.on("error", function (e) { throw new Error(e.message); });
})();
```