Lokaskýrsla

Raforkunotkun heimila

veflausn til að fylgjast með raforkunotkun

Verkefnið var styrkt af Orkusjóði
Styrkverkefni nr. 15-2013

HÁSKÓLI ÍSLANDS
IÐNAÐARVERKFRÆÐI-, VÉLAVERKFRÆÐI- OG TÖLVUNARFRÆÐIDEILD

Rúnar Unnþórsson
Háskóli Íslands

Table of Contents

Formáli

Þetta er lokaskýrsla í verkefni sem nefnist „Raforkunotkun heimila“ og var styrkt af Orkusjóði (Styrkverkefni nr. 15-2013). Markmið verkefnisins var að koma upp tækjabúnaði og þróa hugbúnað sem gerði smærri neytendum mögulegt að koma upp raforkuvöktun hjá sér og geta fylgst með henni í gegnum vefsíðu. Einnig var markmið að skrá raforkunotkunina i gagnagrunn þar sem neytandinn gæti nálgast gögnin til að vinna frekar.

Hugbúnaðinn má nýta til að fylgjast með fleiru en orkunotkun heimila, til að mynda má nota lausnina til að fylgjast með orkuframleiðslu vistvænna lausna eins og vindtúrbína og sólarrafhlaðna. Lausina má einnig nota til að fylgjast með mælistærðum sem eru ekki orkueiningar eins og stærðum í fiskeld; s.s., hitastigi, kotvísýringsgildum, súrefnisgildum og fjölmörgum öðrum mælistærðum.

Þróaður var hugbúnaður sem notendur geta hlaðið niður og sett upp sjálfir á sínum netþjónum. Í verkefninu var stuðst við netþjón hjá Amazon sem notendur lausnarinnar geta einnig notað.

Lokaskýrslan lýsir hugbúnaðinum sem var þróaður, hvernig á að nota hann og hvernig má sækja gögn úr gagnagrunninum. Skýrslan er því nokkurns konar handbók fyrir mögulega notendur lausnarinnar. Skýrslan er skrifuð á ensku vegna aðkomu útlendinga að verkefninu og einnig til auðvelda aðkomu þeirra sem ekki skilja íslensku.

Introduction

The **Time Series Data Viewer** (TiSDaV) was built for persons who are watching/monitoring measurements over a long time e.g. power consumption parameters of residential houses (average power, maximum power, voltage fluctuations, etc.), power generation parameters of wind turbines (voltage, ampere, wind speed etc.), aquaculture parameters (temperature, CO2, oxygen and many more).

This manual is a documentation regarding all the features of the web app and how it works for everyday use. It also explains how maintenance is done, both on the data and the user interface.

The manual is organized into the following sections: Section 1 - Overivew which covers the design of the TiSDaV and its functionalities.. Section 2 - Usage examples which provides the reader with information on how to use the the web app and few usage examples. Section 3 - Modifications which provides information on how to customize and adjust the the web app.

The source code is currently located at https://github.com/birgirob/TiSDaV

1. Overview

The TiSDaV is a responsive web application built to be compatible with both desktops and mobile devices. As of this writing, the website is located at http://54.171.144.202/ (doesn't have a domain name yet) using Amazon Web Services (AWS).

At its core, the application uses the Laravel PHP framework so it follows the model-view-controller (MVC) architectural pattern. The application has three types of user roles; user, administrator and a measuring device.

1. User - Can view the data on the website.
2. Administrator - Can add/edit the charts and locations on the website.
3. Measuring device - Can add data to the database through the website.

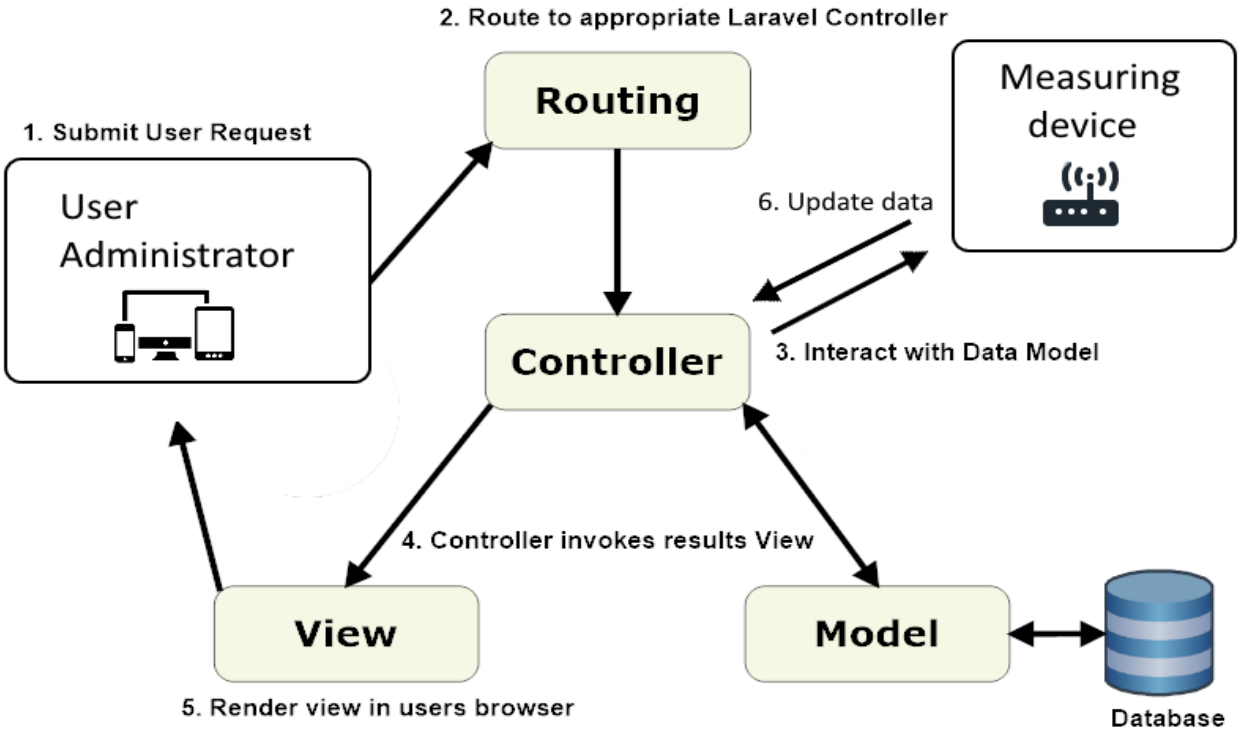

Figure 1: Application overview.

Figure 1 shows the application overview which works as follows:

1. A user or an administrator submit a user request to the server.
2. The route layer finds the right controller.
3. The controller interacts with a model, which fetches data from the database and returns it to the controller.
4. The controller returns a view with the data.
5. The view renders the web page with the data.
6. A measuring device can submit data to a controller, which then uses the model to insert to the database.

Software and tools used:

1. PHP - Scripting language for web development (http://www.php.net)
2. Laravel - PHP framework (http://www.laravel.com)
3. MySQL database (http://www.mysql.com)
4. Composer - PHP package manager (http://www.getcomposer.org)
5. JQuery - Javascript library for easy DOM manipulation, event handling and Ajax (http://www.jquery.com)
6. HighStock - Javascript library for timeline charting (http://www.highcharts.com)
7. Bootstrap - Javascript framework for responsive websites (http://www.getbootstrap.com)
8. Yeti Boostrap Theme (http://www.bootswatch.com/yeti)
9. Gulp - A Javascript task runner (http://www.gulpjs.com)

2. Usage examples

2.1 Viewing time series

The website has a location picker and three sections for each location; status, charts and statistics. When it loads up, it selects the first location by default and shows the status tab for that location.

Status Tab (http://54.171.144.202/#status)

Figure 2 shows the status table for the selected location, where each row has the chart type (as an icon), chart name and the newest value for that chart. It updates every 5 minutes.

Týpa	Heiti	Nýjasta gildi
	Hitastig	12.0 °C
	Rakastig	48.0 %
	Loftgæði	327.0 ppm

Figure 2: The status table.

Chart Tab (http://54.171.144.202/#charts)

Figure 3 shows the chart tab which contains a chart picker and the chart underneath. It is possible to draw multiple series at once to compare values. The chart container updates itself every 5 minutes with new values (if there are any).

It has a zoom picker, a date interval picker, the main chart and a navigator window.

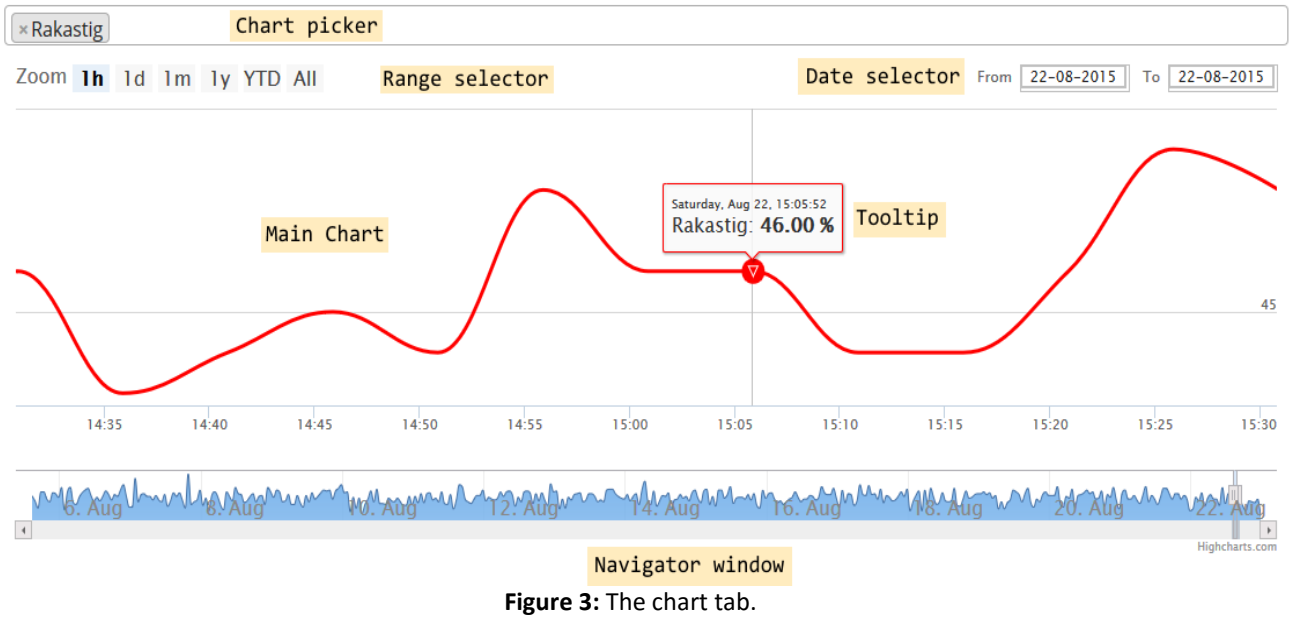

Figure 3: The chart tab.

The range selector allows to quickly set the view with a specific time interval, e.g. [1h] for a one hour interval, [1d] for a 1 day interval etc. Also has a special [YTD] interval which means “year to date” which shows the current year.

The date selector has a calendar popup to pick a date interval. Also supports manual input. The main chart is a time series chart with the values on the Y axis and the time on the X axis. It is possible to move the current view by dragging it with the mouse. It also supports hovering with the mouse on a specific time to view exact data.

The chart is redrawn with new data every 5 minutes (if there is any). The navigator window shows a summarized view of the main chart and shows the chart’s current view. It provides a more visual way of navigating the data set by dragging the scrollbar or dragging the edges of the navigator window.

Statistics Tab (http://54.171.144.202/#statistics)

Figure 4 shows the statistics table which has the minimum, maximum and average values for each chart grouped by months and years. The charts and years rows are expandable.

Tímabil	Min	Max	Avg.
➤Hitastig			
➤2015			
Ágúst	11.5 °C	34.0 °C	22.50 °C
September	11.5 °C	33.0 °C	22.47 °C
➤Rakastig			
➤2015			
➤Loftgæði			
➤2015			

Figure 4: The statistics table.

2.2 Administrative Use

Logging in (http://54.171.144.202/#login) gives administrative privileges for the app. That is done by selecting [Innskráning] and entering in E-mail and password.

As an administrator it is possible to edit, hide and add charts.

The administrator can also add new locations and new unit types.

Add a chart (http://54.171.144.202/#createChart)

Pick [Nýtt graf] from the [Stjórnborð] dropdown.

Figure 5 shows the chart creation dialog. Type in the chart name and pick location, unit type and chart type in the dialog.

Click [Vista graf] to save the chart.

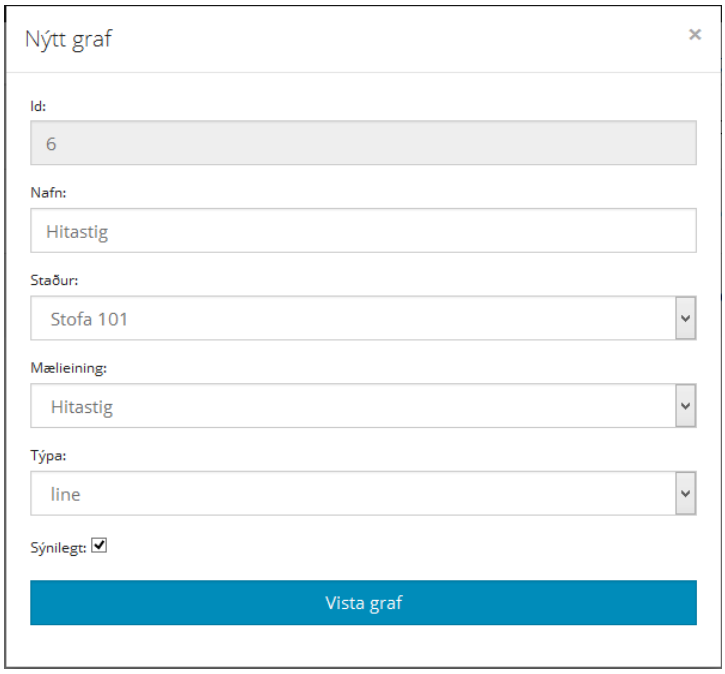

Figure 5: Creating a new chart.

Edit a chart

Figure 6 shows the edit button which appears on hovering the status table with the mouse.

A dialog, identical to the chart creation dialog, will popup with the chart's current information available for editing.

Click [Uppfæra graf] to update the chart.

Figure 6: Edit button.

Hide a chart

Click on the [Edit icon] on the row of the chart to be hidden.

Unselect the [Sýnilegt] checkbox and click [Uppfæra graf].

Add a new location (http://54.171.144.202/#createLocation)

Select [Ný staðsetning] from the [Stjórnborð] dropdown.

A location only has one attribute, its name.

Once added, it's possible to add new charts to it.

Add a new unit type (http://54.171.144.202/#createUnit)

Select [Ný mælieining] from the [Stjórnborð] dropdown.

A unit type has three attributes; its name, symbol and icon (optional).

The icon is an image file type, preferably with size 64x64.

3. Modifications

Information here about how one can add/change functionalities of the TisDaV.

3.1 The user interface

The user interface can be altered in a few different ways:

1. CSS - The two main stylesheets of the website are called “all.css” and “bootstrap.min.css”. They are both referenced in “app.blade.php”
    a. The bootstrap css is a theme called Yeti, which is easily swappable by other themes on http://www.bootswatch.com.
    b. The all.css file is a stylesheet generated by Gulp. It has the styles specific to the website. To change it, it is best to change the “app.scss” file in the “/resources/assets/scss” folder and compile it by running “gulp” using the command line in the project’s root folder.
2. HTML - The website’s html is split into several Blade layout files - two main files and several partials. Blade is Laravel’s layout templating engine.
    a. app.blade.php - The main layout which all other layouts extend from.
    b. home.blade.php - The layout which contains the tab contents.
    c. partials - The layouts in the “/resources/views/partials” folder are all partial layouts which are included by other layouts. For example, “nav.blade.php” contains the navigation header.
3. Javascript - The main javascript file is called “timeseries.js” which contains the bulk of the project’s functionality. To change the interface would be to change the DOM manipulation code. For example, to change how the statistics table loads, look for the function called “loadStatistics” and change the HTML code.

3.2 The database entries

Currently, the project is configured to connect to a local database. To change that, alter the DB entries in the “.env” file located in the project’s root folder. The database config, “database.php”, is located in the “config” folder and references the .env file.

Figure 7 shows how the database is structured:

1. Charts table - Main table which is connected to every other table (except Users).
    a. name - The name of the chart
    b. refresh_time_seconds - The time between each chart refresh on the website
    c. location_id - The id of the location
    d. type_id - The id of the chart type
    e. unit_id - The id of the chart unit
    f. visible - If the chart should be visible on the website or not
2. Locations table - Contains the locations.
    a. name - The name of the location
    b. created_at / updated_at - Creation and update timestamp (generated by Laravel)
3. Chart_types table - Contains available chart types.
    a. type - The type of the chart. The types are determined by the HighStock library

4. Chart_units table - Contains the chart measurement units.
    a. unit - The name of the unit
    b. symbol - The symbol of the unit
    c. icon_path - The filename of the icon (if it has one)
5. Chart_values table - Contains all of the chart values.
    a. timestamp - The Unix timestamp of the value
    b. value - The value itself. The data type is decimal, supports 8 integer digits and 1 fractional digit.
    c. chart_id - The id of the chart
6. Chart_daily_stats table - Contains monthly stats for all charts.
    a. date - The year and month of the stat
    b. chart_id - The id of the chart
    c. min - The minimum value of the chart for that month
    d. max - The maximum value of the chart for that month
    e. avg - The average value of the chart for that month
7. Users table - Contains the users with administrative privileges (generated by Laravel).
    a. name - The name of the user (not used)
    b. email - The email of the user
    c. password - The password of the user, which is hashed using bcrypt
    d. remember_token - Used with the “remember me” function when signing in
    e. created_at / updated_at - Creation and update timestamps (generated by Laravel)

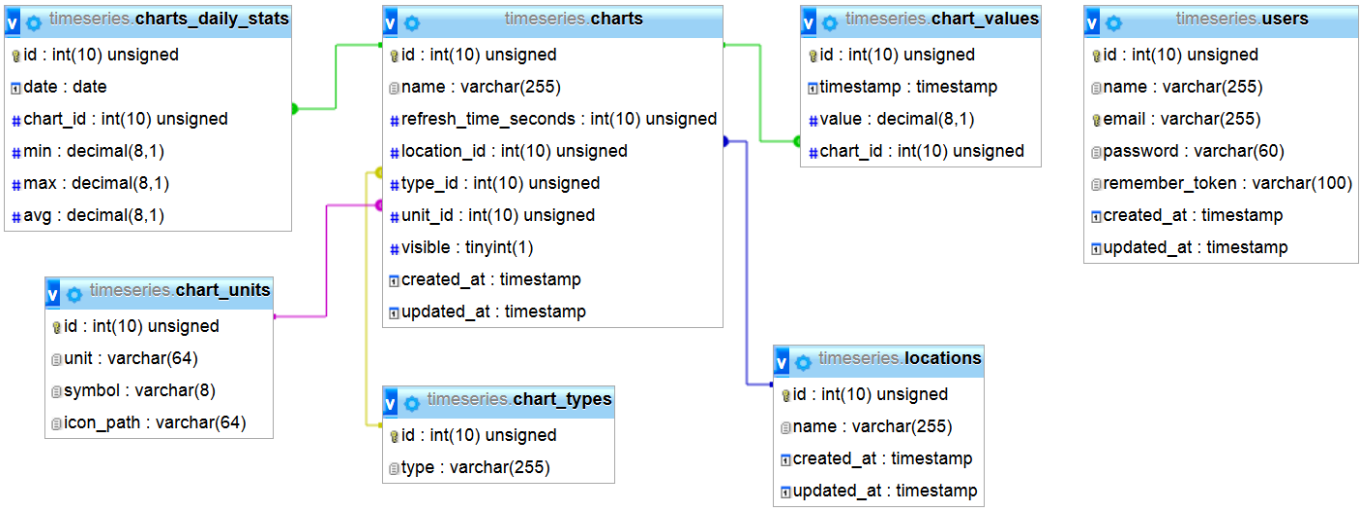

Figure 7: The database structure.